

Elettronica **& Maker**

n° 1 - 2018

**Cubo LED 4x4x4
con PIC**

**Misuratore a Ultrasuoni
del Livello dei Liquidi con PIC**

**Login su WEB
con un Pulsante**

**MAKER sì, ma...
Senza Farsi Male!**

... e molto altro!

Bitcoin:
la Rivoluzione del Blockchain

Sommario

Elettronica & Maker



EDITORIALE

r.armani@elettronicaemaker.it

Si Parte!

Ebbene sì, ci siamo arrivati: il primo numero di Elettronica & Maker guadagna gli spazi del Web! Si è trattato, come potrete immaginare, di un lungo lavoro di preparazione, non ancora concluso, ma indubbiamente fruttuoso e di grande soddisfazione. Potrà anche suonare un po' scontato e retorico ma, prima di parlarvi di questa nuova realtà, desidero ringraziare calorosamente, uno ad uno, gli amici autori che hanno collaborato alla realizzazione di questa prima uscita della rivista, per lo straordinario entusiasmo partecipativo e per l'impegno profuso "a piene mani", nonostante la cronica carenza di tempo di tutti. Ma soprattutto per la fiducia accordata a questo progetto e per averlo subito percepito come qualcosa di proprio. E&M si pone molto vicino ai lettori, con la passione comune per l'elettronica e la scienza, si rivolge al tecnico di collaudata esperienza così come a colui che magari preferisce un approccio un po' più semplice ed immediato, oppure a chi si trova in piena fase di apprendimento ed ha molte domande da porre. Soprattutto sotto questo aspetto E&M vuole lasciare un notevole spazio al rapporto autore-lettore, garantendo la comunicazione immediata, in un percorso che, nei fatti, si rivela sempre di crescita comune. Stiamo vivendo un periodo di rinnovato interesse per l'elettronica, anche e soprattutto grazie alle piattaforme quali Arduino, Raspberry PI e alle derivazioni compatibili. Questi prodotti hanno avuto un successo superiore alle più rosee aspettative proprio per la loro stessa

filosofia di approccio; sono veloci, intuitivi e permettono di ottenere risultati in tempi brevi. Quest'ultima caratteristica è dovuta proprio alla vera e propria esplosione "social" che si è verificata intorno ad essi, con lo sviluppo e la diffusione di un imponente numero di applicazioni, fruibili da chiunque, che hanno dato (e danno...) a moltissimi appassionati la possibilità di arrivare velocemente ai risultati desiderati. In queste condizioni, con i propri progetti si parte in fretta e, soprattutto... si parte avvantaggiati!

Buona Lettura.

Roberto Armani

Due Parole...

- 4** **Maker sì, ma...senza farsi male!**
- 14** **Blockchain: la rivoluzione che ha cambiato il mondo!**

Arduino, Raspberry & Co.

- 24** **Progetto ArduStepper**
- 34** **Inseguitori di linea con Arduino**
- 46** **Login su Web con un pulsante**

Maker

- 54** **Telecontrollo IoT per Condizionatori**
- 64** **Alimentatore Custom**

PIC Micro

- 70** **Rilevatore Ultrasonico di Livello**
- 80** **Cubo 4x4x4 con PIC**

Integrati

- 89** **Riavvio Automatico con LTC4012**



Maker sì, ma... ...Senza Farsi Male!

di Mario Rotigni

m.rotigni@elettronicaemaker.it

L'attività di ogni maker o hobbista elettronico porta a contatto con una varietà di attrezzature, lavorazioni, grandezze fisiche, non prive a priori di un certo grado di pericolo. Questo articolo vuole portare a riflettere sull'argomento sicurezza forse non sempre correttamente considerato in ambito hobbistico.

La cultura della sicurezza nelle attività quotidiane, lavorative ma non solo, è molto diffusa nel mondo anglosassone. Da un paio di decenni si sta meglio diffondendo anche nel nostro Paese, purtroppo spinta dalla necessità di ridurre gli infortuni sul lavoro che troppo spesso popolano le cronache. La legislazione ha costruito un articolato quadro di riferimento (vedi Riferimento 1), sostenuto anche da enti preposti alla capillare applicazione (vedi Riferimento 2). Volendo parlare quindi di sicurezza in attività lavorative o, come nel nostro caso, affini, è possibile trarre profitto da tale ambiente, senza bisogno di reinventare nulla, naturalmente tenendo presenti i differenti ambiti ed esigenze.

Si procede anzitutto all'identificazio-

ne dei pericoli presenti nell'ambiente e nelle attività considerate.

A ciascun pericolo è associata una probabilità di incidenti ed una gravità. La gravità tiene conto dei possibili effetti sulle persone. La combinazione di gravità e probabilità definisce il rischio. Il rischio è quindi tanto più elevato quanto maggiore è la gravità delle conseguenze dell'incidente e la probabilità che l'incidente stesso avvenga. La legislazione prevede che in ambiente lavorativo i rischi vengano

valutati e documentati e si intraprendano poi tutte le misure opportune per ridurre al minimo probabilità e conseguenze. Gli interventi possono comprendere formazione, prescrizione di mezzi di protezione individuale (occhiali, caschi, guanti, calzature antinfortunistiche), modifica di impianti ed attrezzature o procedure di lavorazione. Precise responsabilità vengono assegnate a persone che rispondono della corretta implementazione gestione e aggiornamento di



Esempio deteriore di laboratorio dove i limiti di fruibilità e di buon senso sono stati superati



documentazione e procedure.

È interessante notare che la legislazione più recente assegna responsabilità anche al singolo lavoratore, in particolare nella segnalazione di situazioni di pericolo, nella corretta applicazione delle procedure di sicurezza e nello scrupoloso uso dei mezzi di protezione individuale ricevuti in dotazione. In questo articolo condurremo a grandi linee una analisi simile, cercando di identificare i principali rischi associati all'attività hobbistica, in particolare nel campo dell'elettronica. Non si pretende qui di costruire un quadro esaustivo ma di iniziare almeno di introdurre e sviluppare un'attenzione all'argomento sicurezza che dovrebbero diventare parte integrante del bagaglio culturale di ogni maker.

Prima di iniziare, sono opportuni alcuni consigli di carattere generale. Quando ci si avvicina per la prima volta ad una nuova tipologia di lavorazione è meglio non fare da soli ma affidarsi al consiglio e sostegno di persone di persone già esperte. È possibile seguire corsi presso gli istituti professionali o gli istituti tecnici locali oppure frequentare un FabLab o specifici corsi introduttivi in qualche caso proposti da aziende commerciali. Anche le locali associazioni di modellisti potrebbero rivelarsi fon-

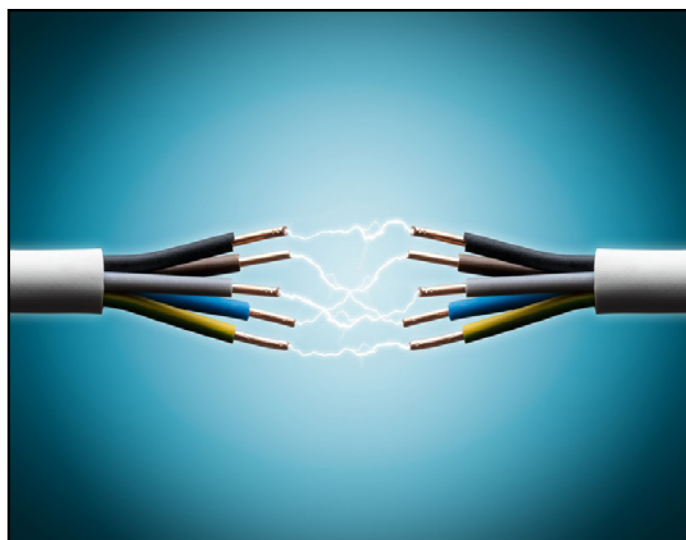
ti preziose. Internet è certamente ricco di risorse anche in questo settore. In particolare cercando in YouTube si possono trovare istruzioni per praticamente qualsiasi tipo di lavorazione possa rendersi necessaria.

Dal punto di vista sicurezza a parere dell'autore è meglio affidarsi direttamente all'interazione con persone fisiche, con maggiori possibilità di avere indicazioni pratiche e risolvere in modo veloce ed affidabile i propri dubbi. Un'altra raccomandazione fondamentale riguarda il mantenimento di un costante livello di attenzione e concentrazione sulle attività in corso. Questo è certamente garantito durante l'esecuzione di attività complesse e gratificanti, quali il debug di un circuito, ma viene facilmente disatteso nell'esecuzione di attività ripetitive monotone. Ad esempio durante la saldatura dei componenti su un circuito stampato sarà facile ritrovarci a pensare allo step successivo o alla lista della spesa, lasciando le mani ad eseguire in quasi completa autonomia il ripetitivo processo di posizionamento del saldatore e della lega saldante. Questa mancanza di consapevolezza può facilmente creare fertile terreno per incidenti. Non dobbiamo dimenticare che il saldatore è un oggetto a temperatura superiore ai 250° nelle immediate vi-

cinanze del nostro corpo. Molti altri esempi simili si potrebbero fare circa quello che viene indicato come “pilota automatico”, attitudine assolutamente comune agli esseri umani ma non per questo meno pericolosa. Le note che seguono sono inoltre rivolte ad un pubblico adulto e danno per scontato che utensili e materiali citati non siano lasciati a disposizione di bambini senza supervisione. Questo articolo non può certamente essere esaustivo, troppo vasta essendo la casistica possibile. Tuttavia è auspicio dell'autore che il metodo seguito e le osservazioni riportate possono richiamare l'attenzione di tutti noi sull'esecuzione in sicurezza di tutte le attività connesse al nostro hobby (e non solo), contribuendo alla creazione di una mentalità che ci accompagni e si evolva con il nostro hobby.

RISCHIO ELETTRICO

Il rischio elettrico è certamente il più diffuso elemento con cui ci troveremo a trattare, non fosse altro per il fatto che stiamo realizzando circuiti elettrici ed elettronici, lavorando su apparecchiature aperte, prive del contenitore che impedirebbe il contatto con parti in tensione ad apparecchiatura finita. Qualcuno ha scritto che la corrente elettrica non si vede e quando si sente è troppo tardi. Considerazione forse banale ma



Scariche elettriche accidentali
rappresentano un notevole fattore di rischio, non sempre correttamente considerato

purtroppo estremamente fondata, tanto che dovremmo interiorizzarla e farne una regola di comportamento.

Anche se probabilmente lavoreremo soprattutto con basse tensioni di alimentazione, magari a batteria, il primo consiglio è non dare confidenza ai circuiti elettrici in tensione di non prendere l'abitudine di toccare conduttori esposti. Troppo facile può diventare il toccare senza pensarci un circuito a tensione di rete, con il quale probabilmente prima oppure finiremo per avere che fare. La grandezza elettrica direttamente responsabile di danni sui tessuti viventi è in realtà la corrente elettrica e non la tensione di per sé.

La circolazione di corrente causa direttamente danni ai tessuti per ustione e tetanizzazione (contrazioni involontarie)



ed interferisce inoltre con funzioni vitali quali respirazione battito cardiaco. Questi effetti dipendono dal percorso seguito dalla corrente attraverso il corpo e possono risultare fatali. In particolare è indispensabile evitare in qualsiasi situazione che il torace sia interessato da circolazione di correnti, esponendo il cuore a rischio di fibrillazione. La corrente circolante è naturalmente determinata dalla legge di ohm (tensione diviso impedenza) rimettendo quindi in gioco anche la tensione come sorgente di rischio.

Le precauzioni principali da prendere riguardano anzitutto l'impianto elettrico di casa, realizzato da un professionista a regola d'arte e munito di dispositivo salvavita (relais differenziale). Questo dispositivo è molto sensibile e interrompe l'alimentazione elettrica nel caso in cui rilevi correnti disperse, ad esempio attraverso il contatto accidentale con una persona. Si raccomanda di lavorare sui circuiti su un piano di lavoro isolato (tavolo di legno o con superficie materiale plastico) possibilmente indossando calzature con suola isolante. Per evitare contatti accidentali si raccomanda di posizionare i puntali degli strumenti di misura saldamente fissati nel punto di interesse a circuito spento (ad esempio usando pizze coccodrillo). Qualora questo non sia possibile o pratico si raccomanda di fissare in modo solido

uno dei terminali di misura (tipicamente massa), toccando con il secondo terminale di misura i nodi del circuito di interesse tenendo l'altra mano dietro la schiena (idealmente). Si evita in questo modo di costituire un percorso conduttivo con il proprio corpo. Il puntale di misura deve avere l'isolamento integro ed essere impugnato con l'isolamento tra sé ed il circuito. È interessante notare come le raccomandazioni fornite circa la postazione di lavoro siano opposte alle prescrizioni volte ad evitare Danni ai semiconduttori da scariche elettrostatiche (ESD). Superfici ed operatore connessi a massa sono la prima linea di difesa da fenomeni ESD ma risultano incompatibili con le precauzioni suggerite. La soluzione consiste nel separare fisicamente la zona montaggio, priva di apparecchiature aperte ed in tensione, dalla zona collaudi dove invece valgono le prescrizioni viste.

RISCHIO INCENDI

L'uso di energia elettrica comporta la generazione e gestione di calore. Il rischio incendio può innanzitutto essere determinato da un'eccessiva concentrazione di calore in presenza di materiali combustibili. La progettazione termica della propria applicazione deve essere quindi sempre attentamente considerata (per una introduzione sistematica, anche se non elementare, vedi Riferimento 6).



Durante la fase sperimentale di un progetto, il rischio incendi non va mai sottovalutato...

Rischi di incendio in apparecchiature elettroniche moderne possono derivare dall'uso improprio di batterie litio (vedi ad esempio Riferimento 5) e da condensatori al tantalio impropriamente connessi o sollecitati. Tali condensatori possono prendere fuoco se sottoposti a tensione di alimentazione superiore a quella nominale e richiedono quindi opportune cautele nel dimensionamento e nella protezione durante il funzionamento.

Un'altra possibile rischio è rappresentato dalle cosiddette ciabatte di alimentazione. Non si deve dimenticare che prolunghe e ciabatte sono realizzate per una massima corrente erogata. Collegando prese multiple magari con ulteriori ciabatte o adattatori è facile

superare involontariamente la portata nominale con susseguente surriscaldamento e rischio. È opportuno tenere a portata di mano un piccolo estintore che essendo possibilmente chiamato ad intervenire su impianti elettrici deve essere a polvere. Familiarizziamo con le istruzioni riportate sull'estintore e ricordiamoci che per mantenere la sua massima efficienza l'apparecchio ha bisogno di una revisione periodica.

RISCHIO MECCANICO

La prima raccomandazione che si può dare per quanto riguarda tutte le lavorazioni meccaniche è l'uso di utensili adeguati al lavoro da eseguire. L'uso di una punta o di un attrezzo non dimensionato correttamente può risultare nella ge-



In meccanica le protezioni sono importanti, comprese quelle per le vie respiratorie quando si lavorano materiali leggeri

nerazione di schegge taglienti lanciate nell'ambiente in modo incontrollato. Anche la corretta affilatura di punte e lame deve essere curata al fine di evitare rotture e grippaggi.

L'inadeguato fissaggio del pezzo in lavorazione è un'altra principale sorgente di rischi. Si deve sempre lavorare su un pezzo ben fermo, evitando ogni possibilità di movimenti incontrollati. Occorre fare attenzione ad evitare di lavorare con organi meccanici in movimento indossando cravatte, braccialetti, collane, qualsiasi cosa possa essere lasciata degli organi stessi (sì, vale anche

per il piccolo Mini-trapano). Lavorando all'asportazione di materiale gli occhi andranno sempre protetti con occhiali e gli attrezzi da taglio non andranno mai, in nessuna situazione, rivolti verso il proprio corpo. Per la cronaca delle raccomandazioni vale anche nelle nostre cucine ed attività quotidiane. Angoli e spigoli vivi vanno smussati immediatamente con qualche opportuno colpo di lima, ad evitare di costituire oggetti taglienti come rasoi. Il consiglio dato dell'introduzione a proposito di tutor, si applica particolarmente bene a tutte le lavorazioni meccaniche non banali.

RISCHIO CHIMICO

Il professore di chimica dell'Istituto tecnico usava dire che la maggior parte dei prodotti chimici sono in qualche modo pericolosi o perché acidi e quindi corrosivi o perché basici e quindi in grado comunque di creare ustioni (I tessuti organici non amano agenti aggressive di alcuna specie). L'uso di ogni sostanza deve quindi essere accompagnato dall'acquisizione delle corrette procedure di manipolazione. Pensando al laboratorio dello sperimentatore elettronico medio, probabilmente il composto

chimico più diffuso è il percloruro ferrico usato per l'incisione di circuiti stampati. Il rischio connesso a questo prodotto sembra soprattutto associato alla sua capacità di macchiare abiti. In realtà si tratta di un prodotto irritante, pericoloso in particolare per gli occhi. Non va inoltre dimenticato come sia in grado di corrodere quasi qualunque metallo e non solamente il rame dei circuiti stampati. Il prodotto va quindi correttamente maneggiato e conservato per evitare indesiderati effetti collaterali. Alcuni consigliano di aggiungere acidi alla soluzione di corrosione per accelerare il tempo di lavorazione. A prescindere dall'efficacia del suggerimento l'autore sconsiglia di portarsi in casa prodotti chimici tra i più pericolosi in commercio. Un ulteriore rischio chimico del laboratorio elettronico può essere considerato il fumo di saldatura. Evitiamo di inalarlo e lavoriamo in ambienti ventilati. Supponendo di non avere che fare con attività continuative, questo dovrebbe essere sufficiente ad evitare rischi per la salute. Una interessante ricaduta della normativa di legge è l'obbligo di indicare per tutte le sostanze chimiche informazioni circa la loro pericolosità,

corretto uso, corretto smaltimento e precauzione in caso di contatti diretti. Sta a noi "perdere" i pochi minuti necessari a leggere le etichette e familiarizzare con le informazioni contenute. La tendenza moderna promossa ed imposta dalla legislatura, vede la fornitura di una vera e propria scheda informativa ad integrazione ed espansione dei dati sintetici contenuti nell'etichetta. Possiamo prendere visione della scheda relativa al percloruro ferrico, ad esempio, all'indirizzo Web indicato nel paragrafo approfondimento (Riferimento 7).

UN NEMICO SILENZIOSO, IL RADON

Argomento in realtà non limitato a maker e sperimentatori, ma poco conosciuto, considerare la presenza di Radon può essere opportuno specialmente qualora



La sperimentazione chimica mette particolarmente a rischio occhi, epidermide e, in taluni casi, anche le vie respiratorie



si operi in seminterrati o al piano terra di edifici. Un ospite silenzioso frequenta le nostre case, con incidenze a priori degne di nota in varie zone d'Italia, il Radon. Il Radon è un gas naturale prodotto del decadimento radioattivo successivo a partire dal capostipite uranio 238, diffuso nella crosta terrestre. Rimane per la maggior parte intrappolato in rocce e terreni, piccole quantità possono filtrare e diffondersi nell'aria e nelle acque. Se inalato viene rapidamente espulso con scarsi effetti sui polmoni. La potenziale pericolosità deriva invece dei prodotti di decadimento radioattivo del Radon che ha periodo di dimezzamento di circa quattro giorni (radiazione alfa), prodotti che possono depositarsi nei tessuti bronchiali, con rischio di sviluppo di tumori polmonari. La probabilità del danno è proporzionale alla concentra-

zione, alla durata dell'esposizione e favorita dall'assenza di ventilazione. Un buon ricambio d'aria risulta efficace nel ridurre il rischio. La misura della concentrazione di Radon risulta complicata dalla necessità di acquisire dati sul lungo periodo (mesi) e da richiedere l'uso di strumentazione dedicata. Essa può essere richiesta agli enti preposti (ARPA), per maggiori informazioni vedi Riferimento 8.

CONCLUSIONI

Abbiamo introdotto un aspetto poco considerato del nostro hobby ed abbiamo identificato i rischi più comuni ad esso connesso. Questo articolo può essere considerato un cantiere aperto alla collaborazione di tutti i lettori, unico modo per coprire un argomento di tale ampiezza.

Per Approfondire:

Riferimento 1: decreto Legislativo 81/08

Riferimento 2: <https://www.inail.it/cs/internet/attivita/prevenzione-e-sicurezza/conoscere-il-rischio>

Riferimento 3: <https://www.inail.it/cs/internet/attivita/prevenzione-e-sicurezza/conoscere-il-rischio/rischio-elettrico.html>

Riferimento 4: CONFERENZA DELLE REGIONI E DELLE PROVINCE AUTONOME, 10/011/CR10c/C7, GUIDA OPERATIVA PER LA SICUREZZA DEGLI IMPIANTI - IMPIANTI ELETTRICI

Riferimento 5: <http://www.alcava.it/SCHEDA%20TECNICHE/Batterie%20e%20Celle%20al%20Litio%20-%20Precauzioni%20di%20Sicurezza.pdf>

Riferimento 6: Design consapevole degli aspetti termici, <http://www.nepss.net/presentations/npss080423.pdf>

Riferimento 7: Esempio di Scheda Sicurezza del Percloruro Ferrico:

https://www.elcoteam.com/media/productfile/570Acido_Cloruro_Ferrico_in_soluzione_acquosa.PDF

Riferimento 8, Radon: <http://www.arpalombardia.it/Pages/Radioattivita/Radon/Livelli%20di%20riferimento.aspx?firstlevel=Radon>



Sono nato a Bergamo nel lontano 1958, 5 mesi dopo il lancio dello Sputnik, quindi già in Era Spaziale. Sono appassionato di elettronica dall'età di 12 anni circa. Il contagio è avvenuto, guarda caso, attraverso riviste regalatemi da un cugino più "anziano". Subito dopo il diploma in Elettrotecnica ho avuto la fortuna di trasformare la mia passione per l'elettronica in

lavoro e l'avventura è ancora in corso 39 anni dopo... Mi sono occupato di progettazione e collaudo di componenti, schede, apparecchiature di produzione, strumentazione e sistemi. Non vi tedio con i dettagli, se siete curiosi potete trovarli nella mia pagina LinkedIn <https://it.linkedin.com/pub/mario-rotignì/28/b74/9b3> Oltre naturalmente a molto studio autonomo, all'ITIS, al Politecnico di Milano ed alle molte occasioni di aggiornamento offerte dalle aziende per cui ho lavorato, sono state determinanti per la mia carriera esempio ed aiuto ricevuti da parte di molte persone (che colgo qui l'occasione per ringraziare). Consigli, spiegazioni, opportunità ricevute, attività insieme, tutto ha "allargato la mente", come si suol dire, aiutandomi ad avanzare in un settore nel frattempo letteralmente esploso e dilagato dai ristretti ma rigorosi confini dell'industria e della ricerca alle mani di tutti noi. Parte di questa rete di contatti, relazioni, sorgenti di informazioni sono anche le riviste di settore, amatoriali e, più tardi, professionali. Ricordo con piacere l'attesa dell'uscita della rivista preferita e la curiosità della mia adolescenza per quello che ci avrei trovato dentro, completamente imprevedibile in molti casi. Oggi mi fa molto piacere partecipare al tentativo di ricreare quelle stesse emozioni in altri, ragazzi e ragazze di ogni età, cosciente dei miei limiti ma desideroso di far trascorrere qualche bel momento e contribuire a consolidare e sostenere la passione che abbiamo in comune.



BLOCKCHAIN:

il sistema che ha cambiato il mondo

di Ivan Scordato

i.scordato@elettronicaemaker.it

Inutile negarlo, specialmente in quest'ultimo anno sono sicuro che abbiate sentito nominare almeno una volta la parola Blockchain. Chi segue il mondo delle criptovalute sa di trovarsi in un momento della storia importante, in quanto si sta assistendo ad una vera e propria rivoluzione. In questo articolo vedremo quali sono le principali applicazioni della Blockchain, spaziando tra criptovalute e IoT.

La parola Blockchain, tradotta significa “catena di blocchi”, e sostanzialmente è un sistema, introdotto dalla criptovaluta Bitcoin, che permette di sviluppare applicazioni “Cryptocurrency-like” funzionando come un vero e proprio registro pubblico condiviso che tiene traccia di tutte le operazioni effettuate che sono state confermate dai singoli nodi, ma per come vedremo è possibile integrare questo sistema anche in altri progetti, come l'IoT e l'accesso ai propri dati personali negli ospedali.

Il funzionamento della Blockchain non è garantito da un ente centrale, ma ogni singola transazione viene convalidata dall'interazione di tutti i nodi. Infatti si tratta di un vero e proprio registro pubblico che viene aggiornato automaticamente dai nodi che fanno parte della rete.



Le operazioni che vengono eseguite all'interno della Blockchain, devono venire confermate dai nodi attraverso un sistema di crittografia, che grazie alle chiavi private che vengono utilizzate per firmare le transazioni, riesce a confermare che una determinata operazione sia stata eseguita in un momento preciso da un determinato individuo.

Nella Blockchain vengono incluse soltanto le transazioni che sono state confermate. La definizione originale fu scritta da Satoshi Nakamoto, l'inventore di Bitcoin, la criptovaluta che ha dato il via alla creazione di molte altre varianti.

Una criptovaluta (o crittovaluta o criptomoneta) è una valuta paritaria, decentralizzata digitale la cui implementazione si basa sui principi della crittografia per convalidare le transazioni e la generazione di moneta in sé. Come ogni valuta digitale, consente di effettuare pagamenti online in maniera sicura.

Le implementazioni spesso usano uno schema proof-of-work come salvaguardia alla contraffazione digitale. Esse utilizzano tecnologie di tipo peer-to-peer (p2p) su reti i cui nodi sono computer di utenti disseminati in tutto il globo. Non c'è attualmente alcuna autorità centrale che le controlla. Le transazioni e il rilascio avvengono collettivamente in rete, pertanto non c'è una gestione di tipo "centralizzato".



Queste proprietà uniche nel suo genere, non possono essere esplicate dai sistemi di pagamento tradizionale.

Wikipedia

BITCOIN

Sono più che sicuro che nemmeno questa parola vi sia nuova.

Bitcoin è una cryptovaluta che è stata idealizzata nel 2008 e creata ufficialmente nel 2009 da qualcuno che nei suoi post utilizzava lo pseudonimo di Satoshi Nakamoto.

Lo scopo principale di Bitcoin è quello di ottenere una moneta digitale che può essere facilmente scambiata tramite la Blockchain, senza essere sotto il controllo di nessun ente centrale.

Nessuno sa chi sia Satoshi Nakamoto, anche se ormai da diversi anni sono state formulate diverse ipotesi, quasi subito tutte smentite. La sua vera identità rimane quindi ignota, così come vuole la filosofia del progetto, ma quello che è certo, è che oltre ad essere un

genio è anche ricco. Il progetto venne caricato su SourceForge nel novembre del 2008.

Bitcoin sfrutta la crittografia per gestire la generazione di nuova moneta e l'attribuzione di proprietà dei bitcoin, inoltre la blockchain consente il possesso ed il trasferimento anonimo della valuta.

COME OTTENERE DEI BITCOIN

Se pensi di volere ottenere dei bitcoin ma non hai idea di come fare, prima di chiedere come fare dovresti chiedere a te stesso perché vuoi ottenerli. Solo una volta fatta chiarezza su questo punto potrai decidere come ottenerli.

Questo appunto perché sicuramente ci sarà chi vuole compiere questo passo per specularci sopra, cercando di ottenere il massimo profitto nel minor tempo possibile, ma c'è anche chi è semplicemente innamorato della filosofia del progetto. Le modalità principali per ottenere dei bitcoin sono 3:



- Acquistarli da una persona fisica
- Acquistarli tramite un sito di Exchange
- Ottenerli tramite il Mining o Cloud Mining

Vediamo quale è la differenza tra i canali proposti.

ACQUISTO DA UNA PERSONA FISICA

Nel momento in cui hai capito quali sono i rischi delle cryptovalute e hai voglia di cominciare a capire esattamente come funziona il sistema o semplicemente devi fare un pagamento in bitcoin, il metodo più veloce, anche se non più economico, è quello di acquistare una quantità di bitcoin da una persona fisica che conosciamo.

Se non conoscete nessuno che venda bitcoin non preoccupatevi, basta cercare online sul sito localbitcoin qualcuno che si trovi dalle vostre parti e che offra questo servizio, meglio però che prima controlliate se abbia una buona reputazione e feedback positivi.

SITO WEB DI EXCHANGE

Se non ci fidiamo molto a fare uno scambio con persone che non conosciamo, qui entrano in gioco i siti web di Exchange, che permettono, dopo essersi registrati ed avere confermato la propria identità, di scambiare le comuni valute con delle cryptovalute e viceversa. Tra-

mite questo sistema è possibile decidere quando comprare e quando vendere i propri bitcoin, potendo così ottenere un profitto anche considerevole qualora si dovessero fanno bene i conti.

MINING E POOL MINING

Il terzo metodo per ottenere dei bitcoin, è quello di andare a cercarli in miniera. Ovviamente si parla in senso figurato, quello di cui stiamo parlando è il Mining e il Cloud Mining.

Il Mining è il processo attraverso il quale le persone che sostengono il network della blockchain attraverso la condivisione della potenza di calcolo dei loro computer, vengono ricompensate con una determinata quantità di bitcoin.

La potenza di calcolo messa a disposizione dai miners è necessaria all'autenticazione delle transazioni all'interno





Imponente cluster di apparati per il cloud mining. Il rapporto tra i costi di acquisto e soprattutto di gestione energetica dell'hardware, in relazione al profitto dell'attività di mining, è critico e di gestione tutt'altro che banale.

della Blockchain attraverso operazioni complesse. Inizialmente, per contribuire alla blockchain, bastava utilizzare normalissimi computer senza particolari specifiche hardware, in quanto la difficoltà di calcolo non era molto elevata.

A lungo andare, con l'espandersi della blockchain e l'esponenziale aumento della difficoltà di calcolo, l'utilizzo delle singole CPU divenne inefficiente.

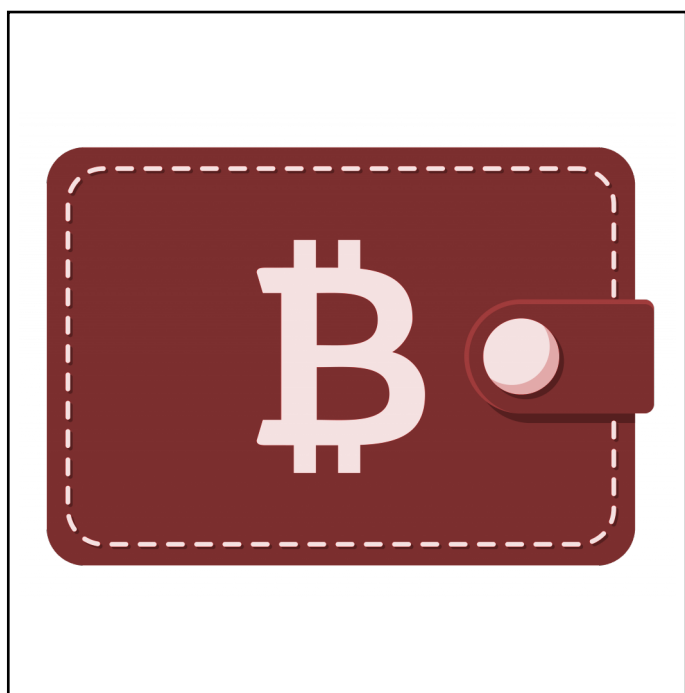
Per appagare la fame di potenza della blockchain fu necessario spostarsi dal

mining attraverso CPU, al mining tramite l'impiego di GPU e FPGA, che vennero in seguito sostituite con dei banchi di chip ASIC. Nascono così dei computer realizzati ad hoc per minare bitcoin, aumentando sempre di più la potenza di calcolo e cercando di diminuire i consumi della corrente elettrica.

Se è vero che all'inizio della storia di Bitcoin per trovare e risolvere un blocco non era necessario impiegare dei computer con un'elevata potenza di calcolo, con il passare del tempo divenne

necessario non soltanto utilizzare dei computer assemblati ad hoc con banki di ASIC, ma divenne indispensabile spostarsi sul Pool mining. Attraverso il pool mining, tante persone sparse per il mondo, collegano i propri computer ad un unico server per collaborare nel mining di bitcoin, in modo che la potenza di calcolo di ogni client viene sommata a quella degli altri per avere più possibilità di riuscire a creare un blocco prima degli altri.

In questo modo tutte le macchine collegate alla pool, lavorano all'unisono dividendosi il lavoro, aumentando le probabilità di risolvere un blocco per primi ottenendo come ricompensa una determinata quantità di bitcoin, che verrà distribuita automaticamente tra i miners in base alla quota di partecipazione alla pool.



CLOUD MINING

Acquistare e mantenere i computer necessari per il mining, specialmente qui in Italia, ha un costo molto elevato. Se non si è disposti ad effettuare questo investimento, è sempre possibile optare per il Cloud mining, un servizio offerto da molte società, la maggior parte delle quali cinesi, che hanno a disposizione grandi quantità di potenza di calcolo che frazionano in piccole parti che offrono in vendita.

In questo modo ognuno è libero di acquistare una determinata quantità di Ghs da utilizzare per minare bitcoin sulla quale dovrà pagare una piccola quota per il costo di manutenzione e della corrente elettrica.

La potenza di calcolo che viene acquistata, può essere gestita dal proprio account ed eventualmente reimmessa nel mercato.

UTILIZZARE LA CRYPTOVALUTA

Per come avviene con le normali valute che bisogna depositarle da qualche parte, anche i Bitcoin hanno bisogno di un posto dove essere conservati per essere utilizzati. Il primo passo per potere utilizzare la cryptovaluta, è infatti quello di generare il proprio Wallet, che consiste in un vero e proprio portafoglio digitale nel quale riporre i propri bitcoin.

Per avere una maggiore sicurezza e



controllo sui propri bitcoin, è consigliabile utilizzare il proprio wallet sul computer personale.

Se si vuole avere la possibilità di poter accedere al proprio wallet con facilità dovunque ci si trovi, è possibile fare ricorso ad uno dei tanti servizi online che permettono di accedere al nostro wallet digitale ovunque ci si trovi.

In alternativa, in commercio esistono degli hardware wallet tramite i quali è possibile gestire e portare sempre con sé il proprio wallet e, a detta del produttore, nella massima sicurezza.

Una volta caricati dei Bitcoin sul nostro portafoglio preferito, non dobbiamo fare altro che decidere cosa farne.

Possiamo utilizzarli per scambiare beni e servizi, per fare degli investimenti o delle donazioni. Anche se sono in costante crescita, i negozi che accettano pagamenti in bitcoin in Italia non sono molti, ma nel caso in cui vi siate messi in testa l'idea di voler pagare un determinato prodotto in bitcoin, ma sfortunatamente il negozio in cui si trova il prodotto non accetta questa tipologia di pagamento, esistono dei servizi che permettono di pagare in valute convenzionali tramite la conversione istantanea della quantità necessaria di bitcoin. Naturalmente usufruendo di questo servizio bisogna considerare la presenza, seppur minima, di una commissione.

BLOCKCHAIN NELL'IOT

Da alcune statistiche di Cisco risulta che entro il 2020 i dispositivi che saranno collegati online saranno più di 50 miliardi.

Questo significa che tutti questi dispositivi che invieranno e riceveranno dati sulla rete, porteranno dei problemi all'intera rete Internet, infatti basti pensare alla grande mole di dati che deve essere processata.

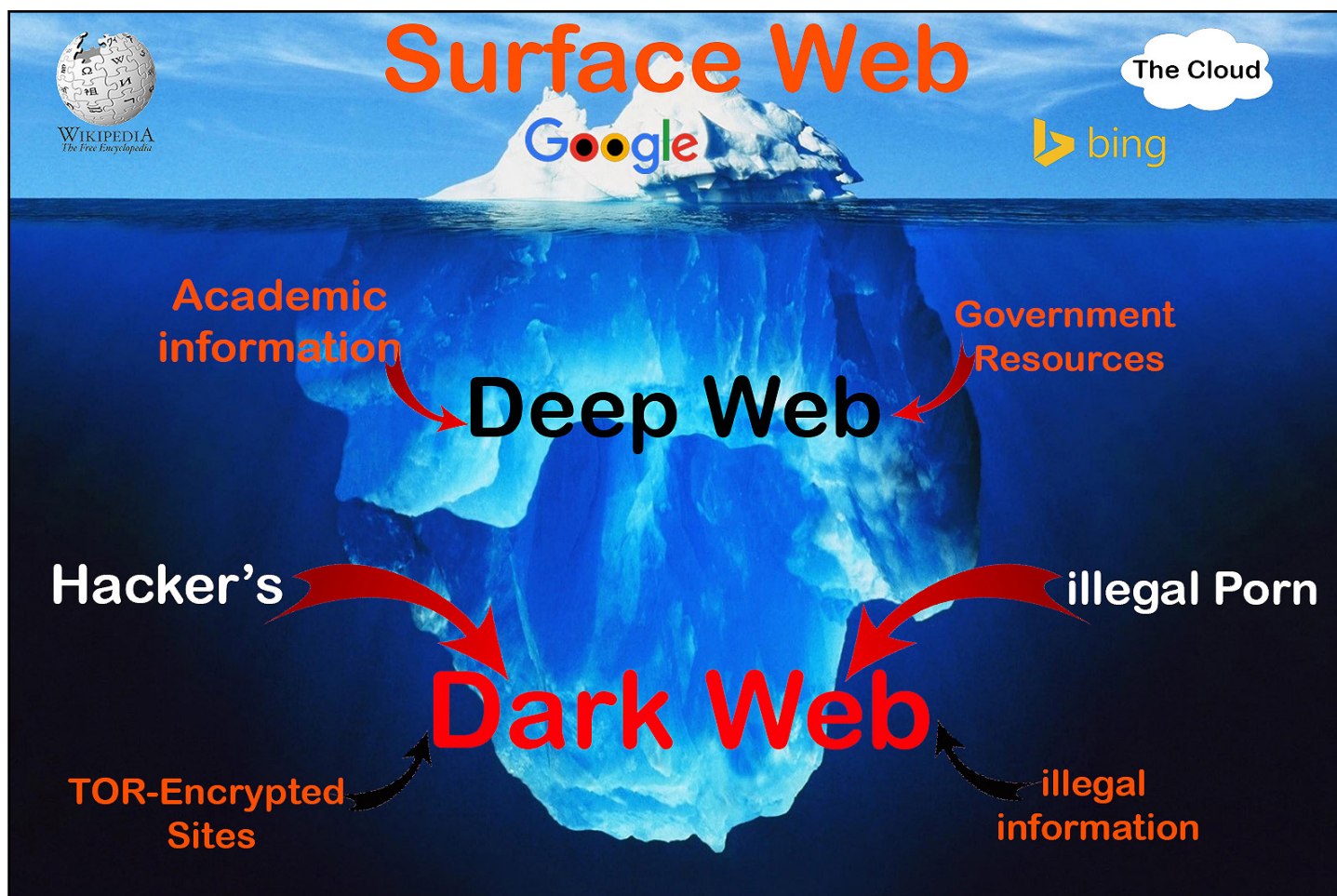
Anche la maggior parte dei dispositivi IoT attualmente si basano sulla comunicazione client/server, e quindi significa che probabilmente, entro il 2020 potrebbero cominciare ad avere dei malfunzionamenti dovuti al ritardo nell'invio e nella ricezione dei comandi di controllo.

Diverse compagnie, tra cui IBM e Samsung, stanno lavorando sulla realizzazione di un sistema IoT che basi il proprio funzionamento sulla tecnologia Blockchain, con lo scopo di creare una rete decentralizzata in modo da poter eliminare l'HUB centrale necessario per permettere la comunicazione dei dispositivi, creando un vero e proprio registro pubblico delle operazioni eseguite.

BLOCKCHAIN NEL CAMPO MEDICO

Anche in campo medico sembra che si stia muovendo qualcosa.

Infatti è recente l'interessante propo-



Si stima che la parte chiara (emersa) del web sia soltanto una percentuale modesta di tutto quanto invece si trova in rete. La porzione sommersa non risulterà di immediata accessibilità, a meno di non utilizzare strumenti dedicati e, vista la rappresentazione grafica, non senza margini di rischio.

sta da parte di Tierion di implementare la tecnologia della Blockchain in campo medico, realizzando un database contenente i dati dei pazienti, la storia clinica ed eventuali dati sullo stato di salute acquisiti attraverso dispositivi di monitoraggio dei parametri vitali.

In questo modo, in caso di problemi, ogni ospedale avrebbe direttamente accesso alla storia clinica di ogni paziente, venendo subito a conoscen-

za del gruppo sanguigno, di eventuali allergie, patologie e tutto quello che è importante sapere specialmente quando è necessario un pronto intervento.

CRYPTOVALUTE UTILIZZATE PER FINI ILLEGALI

Tutto quanto, nelle mani sbagliate può diventare un'arma pericolosa.

Posso avere un coltello ed utilizzarlo per preparare un'ottima cena, oppure



per andare a fare una rapina.

La creazione delle cryptovalute è stata vista come una grande opportunità anche da coloro che hanno un'indole criminale.

Ad esempio hanno trovato largo impiego nel Deep Web, grazie al fatto che prendendo i giusti accorgimenti nello scambiarle, è difficile, se non impossibile, risalire all'identità del proprietario, permettendo così l'acquisto di molti beni, la maggior parte dei quali illegali. Spesso si è sentito parlare dei temutissimi Ransomware, virus informatici che rendono inutilizzabili i file del computer

infetto, e l'unico modo per recuperarli è quello di pagare una somma in cryptomonete, generalmente in bitcoin.

Negli ultimi giorni su Facebook gira la foto di un foglio che è stato distribuito in molti negozi di Bergamo, dove viene richiesto il pagamento di un vero e proprio pizzo direttamente in bitcoin.

CONCLUSIONE

Come avrete sicuramente notato, l'utilizzo della Blockchain e delle cryptovalute ha riscosso molto successo in moltissimi campi diversi tra loro, aprendo le porte ad una nuova era digitale.

Molte persone all'inizio della storia di Bitcoin pensavano che si trattasse soltanto di un gioco, non tenendo realmente conto della filosofia e della struttura del progetto, che come abbiamo visto, è stata di ispirazione per la proposta di nuovi dispositivi collegati ad una rete decentralizzata e di database medici accessibili da qualunque parte del mondo.

In questo articolo non ho parlato volutamente del fatto se investire o meno nelle cryptovalute, in quanto è una scelta che andrebbe analizzata a parte, ma mi sono limitato a cercare di fornire delle infor-

La sua attività imprenditoriale soddisfa i requisiti necessari per il pagamento obbligatorio di 50€ mensili da corrispondere per il mese corrente e successivamente la prima settimana di ciascun mese successivo.

Il pagamento obbligatorio deve essere versato in bitcoin acquistabili sul sito www.coinbase.com oppure www.localbitcoins.com con carta di credito, conto corrente o altre forme di pagamento disponibili.

Per acquistare i bitcoin è necessario registrare un account su uno dei 2 siti sopracitati e procedere alle verifiche necessarie di identità.

L'importo richiesto di soli 50€ mensili è volutamente basso e molte imprese hanno già aderito a questo programma obbligatorio fin da subito per convenienza.

Denunciare la presente lettera ostacolerebbe soltanto i pagamenti da versare da parte delle forze dell'ordine e il mancato pagamento mensile dell'importo richiesto comporta a danni economici dell'immagine della sua attività imprenditoriale sia online che offline, nonché danni fisici ai beni della sua attività imprenditoriale.

Le forze dell'ordine dispongono di risorse insufficienti per proteggere la sua attività imprenditoriale; quindi la perdita economica è molto superiore ai 50€ mensili richiesti e si troverebbe con le mani legate senza più la possibilità di uscirne fuori. Pagare con regolarità fin dal primo mese è la cosa giusta da fare e vi è anche la possibilità di pagare un anno anticipato a 500€ anziché 600€.

Per ciascun intervento di riscossione, che può avvenire anche a distanza di mesi, verranno addebitati 500€ per l'intervento degli addetti ai lavori che vanno sommati agli eventuali arretrati dei mesi scorsi e la riparazione dei danni arrecati dall'intervento. Una spesa di oltre 1000€ che cresce esponenzialmente nel tempo, quando sarebbe sufficiente corrispondere 50€ al mese dall'inizio.

L'acquisto di bitcoin mediante www.coinbase.com oppure www.localbitcoins.com è molto semplice, sicuro e veloce.
Se necessario, farsi aiutare da un utente che sia in grado di effettuare pagamenti online mediante carta di credito oppure conto corrente.

Una volta incassati i bitcoin sul suo account, bisogna versare l'importo mensile sul seguente indirizzo bitcoin e fare molta attenzione a minuscole, minuscole e numeri:

1ASerEbQNbcZ4PnjTWX2Ws9J56Kf7fXM5k

Parte del suo contributo sarà devoluto in beneficenza.
Cordiali saluti

mazioni riguardo il loro funzionamento, focalizzandomi sul Bitcoin che ha dato il via a tutto. Per qualsiasi informazione in più riguardo ai Bitcoin,

sono sicuro che possiate trovare interessante consultare la wiki ufficiale, nella quale troverete informazioni e riferimenti dettagliati su tutto.



Sono Ivan Scordato, attualmente studente all'università di Palermo di Ingegneria Cibernetica. Appassionato di elettronica, informatica e del fai da te sin da piccolo, non perdo occasione per lavorare su qualche progetto interessante ed imparare cose nuove. Nel

tempo ho pubblicato diversi articoli sul sito EMC Elettronica e nella rivista Fare Elettronica, scrivendo in particolare sulle board di sviluppo Arduino e Raspberry Pi, in quanto amo particolarmente queste due schede tramite le quali, grazie anche ad una community molto attiva, è possibile realizzare tutto quello di cui si necessita. Spero che i miei articoli possano continuare ad essere interessanti e di stimolo per tutti coloro che vogliono avvicinarsi al mondo dei Makers. Il mio obiettivo principale: Riuscire a fare quello che nessun altro è ancora riuscito a fare. "La mente è come un paracadute. Funziona solo se si apre." (Albert E.)



Progetto

ArduStepper

di Giovanni Carrera

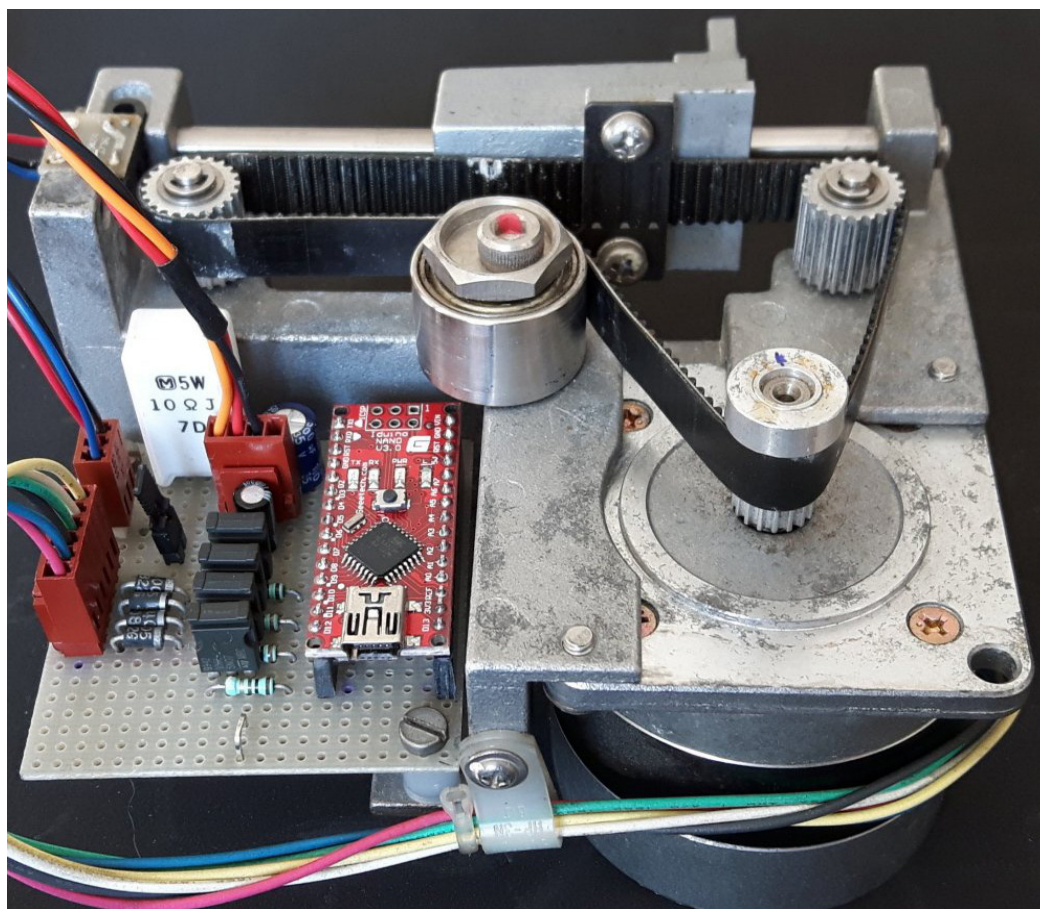
g.carrera@elettronicaemake.it

Scopo di questo progetto è quello di sperimentare sui motori passo-passo unipolari con Arduino controllando un attuatore lineare di recupero.

In una fiera del radioamatore ho acquistato un compatto e robusto attuatore lineare composto da un motore passo-passo che, con una cinghia dentata, muove un cursore su barra tonda e con sensore ottico di finecorsa ad una estremità.

A questo punto ho pensato di realizzare una scheda di pilotaggio per motori unipolari basata su un Arduino Nano. Naturalmente si può costruire la parte meccanica utilizzando una vite senza fine o una cremagliera oppure cinghiette e componenti recuperati da una vecchia stampante o da uno scanner.

Questo progetto non ha elementi molto innovativi ma penso che possa essere utile per capire il funzionamento di questi interessanti motori.



I MOTORI PASSO-PASSO

Una prima classificazione divide questi motori in:

- **Unipolari:**

la corrente scorre negli avvolgimenti statorici sempre nello stesso verso, sono facilmente identificabili perché, possono avere 5, 6 o anche 8 fili. Il circuito di pilotaggio è relativamente semplice: bastano 4 transistori che operano in modo switch. Quelli a 6 o 8 fili si possono trasformare in bipolare mettendo in serie i due avvolgimenti, ma la tensione di alimentazione si raddoppia.

- **Bipolari:**

la corrente cambia verso, quindi occorre un circuito di pilotaggio realizza-

to con 4 transistori switch, disposti a ponte, per ognuna delle due coppie di avvolgimenti. Sono riconoscibili perché, solitamente, hanno solo 4 fili. In entrambi i casi esistono circuiti integrati, più o meno sofisticati, che realizzano il pilotaggio ma non sono adatti per motori medio-grandi per problemi di dissipazione termica.

Questi motori sono

costituiti da un rotore a magnete permanente costituito da due ruote dentate magnetizzate con orientamento opposto e con denti sfasati in modo che la testa del dente di una coincida con la base dell'altra. La figura 1 mostra l'interno di un piccolo motore unipolare, non quello usato in questo progetto, in questo caso le cavità delle ruote dentate sono ricoperte di resina. Lo statore è costituito da diversi avvolgimenti, solitamente otto, che se percorsi da corrente, fanno ruotare il rotore fino a trovare una posizione di equilibrio. Per chiarire meglio il funzionamento si veda il modello semplificato della figura 2. Il rotore, magneticamente polarizzato, lo

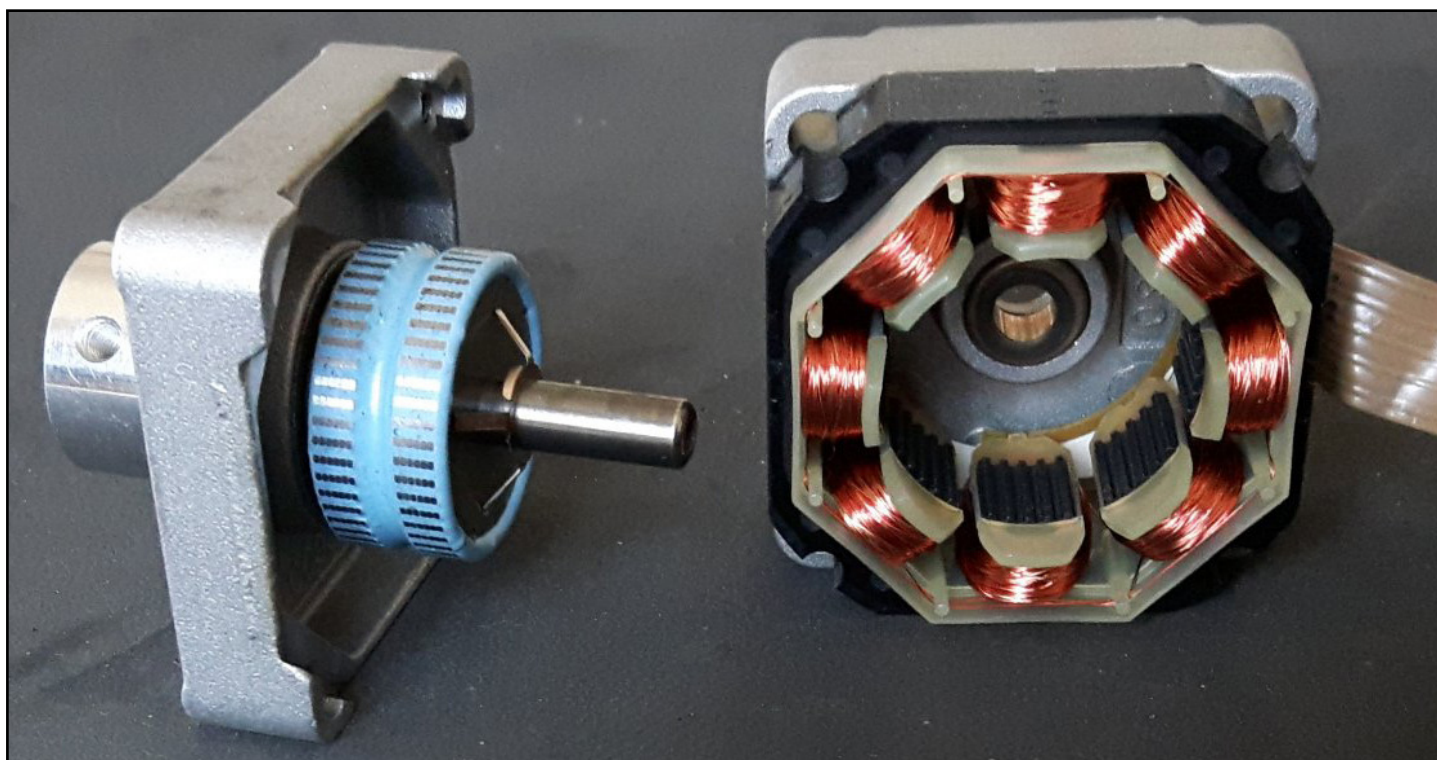


Figura 1. - Interno di un motore passo-passo.

possiamo rappresentare con la lancetta di una bussola e lo statore con quattro avvolgimenti separati (motore unipolare) o in serie (motore bipolare). La figura 2 si riferisce al motore unipolare azionato in modalità “Full Drive Stepping” o “Two phase-on” che è quella più usata.

Questo motore semplificato fa un giro completo in soli quattro passi, nella realtà i passi per giro sono molto di più, quello da me usato ha ben 400 passi per giro, creati dalle fitte dentature del rotore e dello statore.

Il modo di funzionamento “Two pha-

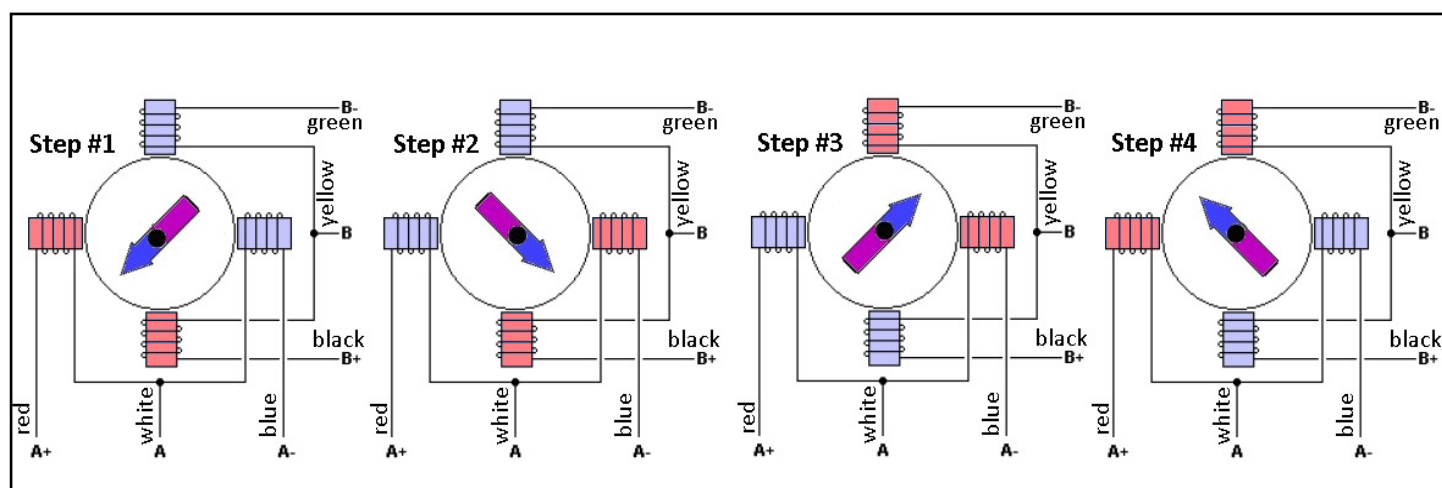


Figura 2 - Sequenza degli step in un motore semplificato.

se-on” è realizzato alimentando contemporaneamente due estremità di fasi differenti. Il rotore troverà equilibrio nella posizione intermedia tra le due espansioni polari e manterrà la posizione finché si manterranno eccitati i due avvolgimenti. Per far ruotare il rotore di uno step, occorre eccitare il solenoide attiguo e spegnere quello precedente, come illustra la tabella 1

La sequenza si ripete a rotazione: 1,2,3,4,1,..

Con la sequenza illustrata il verso di rotazione è antiorario, per cambiare verso si usa la sequenza di Tabella 2.

Il modo “Two phase-on” ha un consumo doppio rispetto a quello che alimenta un solo solenoide alla volta (Wavemode), ma presenta una coppia che è 1.41 volte maggiore (radice quadrata di 2), perché è ottenuta dalla somma vettoriale di due forze ortogonali.

Se si volesse una maggiore risoluzione, occorre utilizzare una libreria che possa pilotare a $\frac{1}{2}$ passo il motore, in rete ne esistono diverse e si trovano anche

Step	A+ lead	A- lead	B+ lead	B- lead
1	1	0	1	0
2	0	1	1	0
3	0	1	0	1
4	1	0	0	1

Tabella 1

Step	A+ lead	A- lead	B+ lead	B- lead
1	1	0	1	0
2	1	0	0	1
3	0	1	0	1
4	0	1	1	0

Tabella 2

patch per modificare quella originale. In questo caso i passi sono otto e si alternano le alimentazioni di due fasi con quelle di una sola (Wave mode). Ne deriva un’irregolarità della coppia e della corrente. La tabella seguente mostra la sequenza half step.

Step	A+ lead	A- lead	B+ lead	B- lead
1	1	0	1	0
2	0	0	1	0
3	0	1	1	0
4	0	1	0	0
5	0	1	0	1
6	0	0	0	1
7	1	0	0	1
8	1	0	0	0

Esistono dei controller, adatti ai motori bipolari, che adottano la tecnica microstepping che consiste nell’operare sempre con due fasi alimentate ma variando opportunamente le correnti nelle fasi, solitamente in PWM, in modo da creare delle condizioni di equilibrio in posizioni intermedie. Con questa tecnica si ottiene un $\frac{1}{2}$ passo più equilibrato e si raggiunge comunemente $\frac{1}{8}$ di



passo e, per alcuni controller, perfino 1/128 di passo. Le correnti, nei vari step del giro, variano con legge sinusoidale e il motore diventa simile a un motore sincrono. Con il controllo microstep si hanno anche minori vibrazioni.

Spesso le dimensioni dei motori sono indicate con un numero NEMA (National Electrical Manufacturers Association) che è pari a 10 volte il diametro in pollici del motore. Si è cercato di standardizzare anche le altre dimensioni, come il diametro dell'asse e il sistema di fissaggio.

IL MOTORE

Il motore passo-passo che era montato nell'attuatore lineare è un VEXTA PH265M-33-C3, un motore a due fasi, unipolare a sei fili, alimentato a 24 V, 400 step/giro ($0,9^\circ$), con corrente di 0,21 A e con resistenza (misurata) di ogni avvol-

gimento pari a circa 109Ω . Esso potrebbe essere un NEMA 22, ma non so se i giapponesi seguano questo standard. Le seguenti figure mostrano l'aspetto del motore, i fili di collegamento e la curva velocità/coppia.

Dalla curva della terza figura si vede che la coppia massima ($37,2 \text{ N}\cdot\text{cm}$) si ha per una frequenza degli impulsi pari a circa 200 Hz, quindi userò un ritardo di 5 millisecondi per la generazione degli impulsi.

In questo caso la velocità dell'asse in rivoluzioni/giro sarà pari a 30 RPM, infatti:

$$30 \cdot 400/60 = 200 \text{ Hz}$$

LA PARTE HARDWARE

Lo schema è visibile in figura 4, tutti i componenti sono montati su una piccola scheda millefori alloggiata vicino al motore.

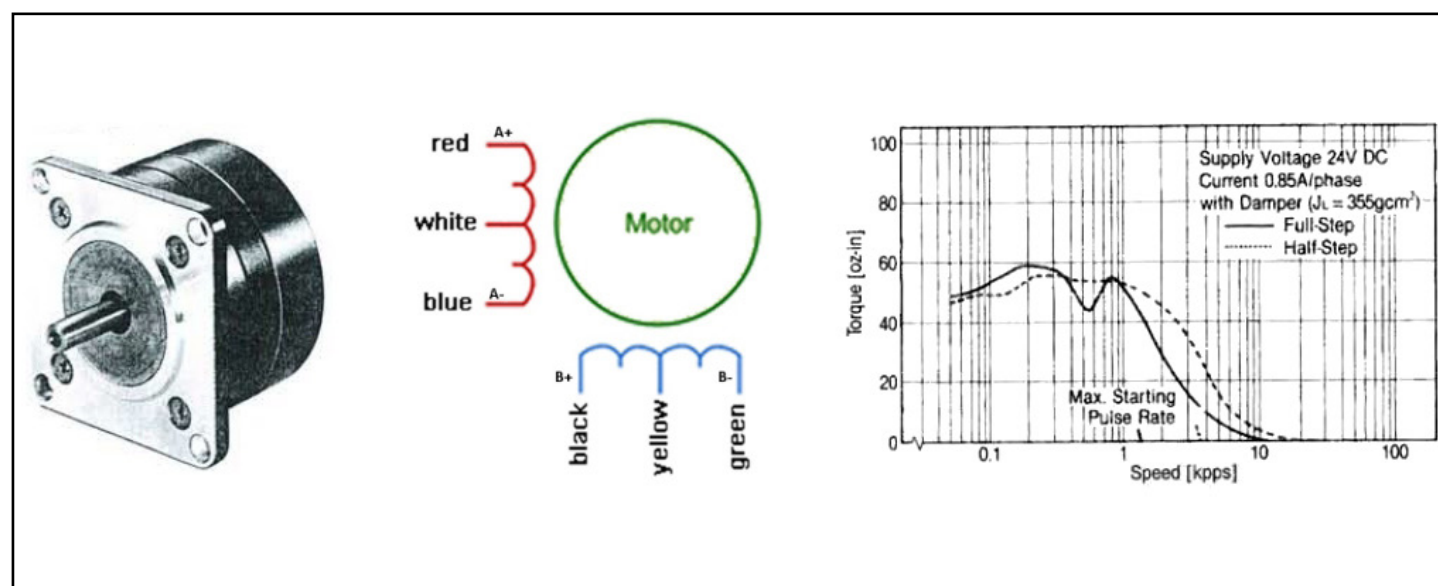


Figura 3 - Il motore utilizzato nel progetto.

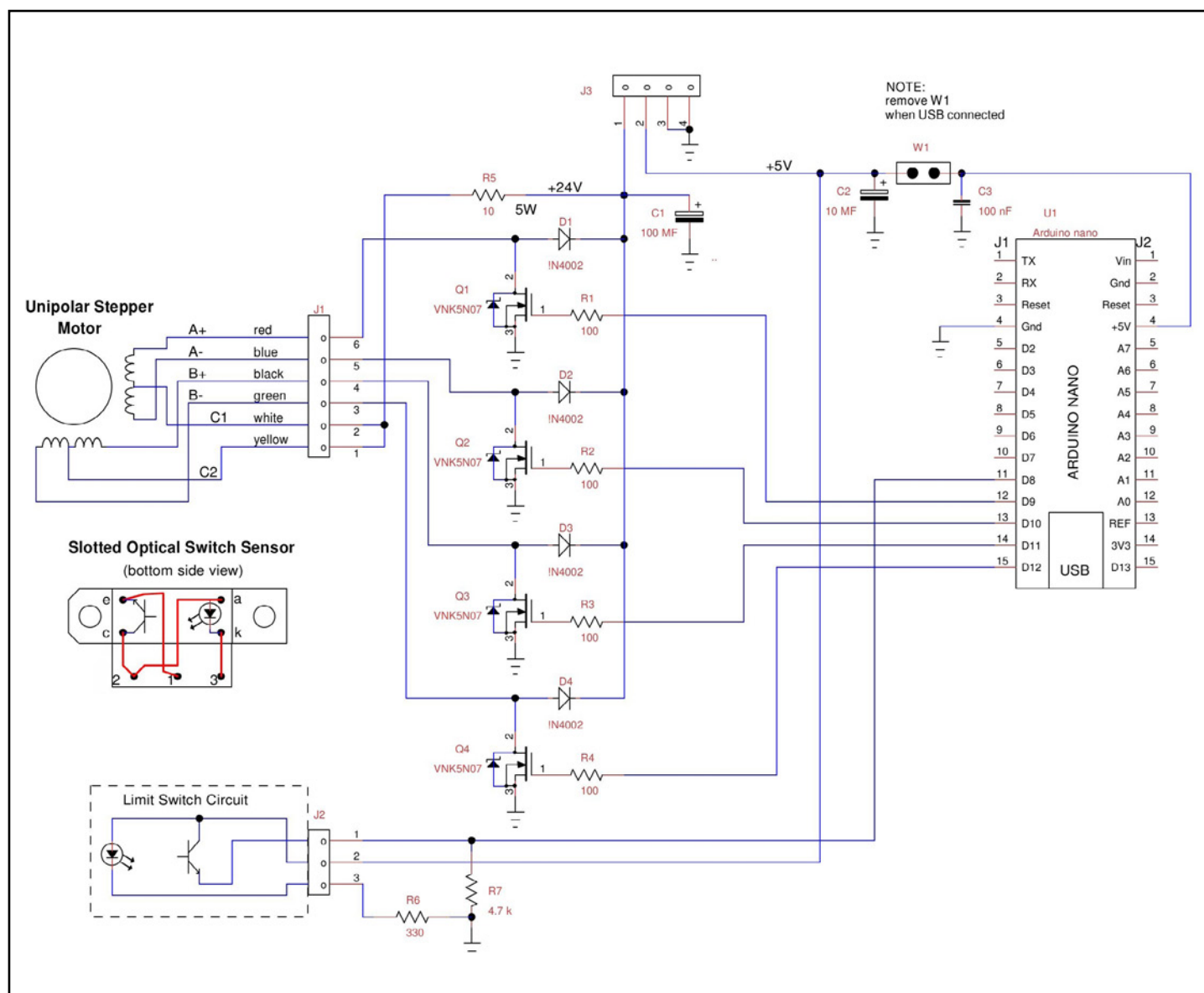


Figura 4 - Schema elettrico di ArduStepper.

Ho preferito usare Arduino Nano perché ha tutti i pin con passo 0,1" (2,54 mm), ha l'interfaccia USB ed è molto compatto. Per pilotare il motore ho usato 4 Mosfet a canale N che avevo recuperato, vanno bene i logic level IRLZ44 e per motori più piccoli si può usare l'integrato ULN2003 che incorpora anche i diodi di protezione. Naturalmente si possono usare anche 4 transistor darlington npn come i TIP120, ma occorre usare dei

dissipatori per via dell'alta tensione di saturazione dei transistori a giunzione bipolare.

I resistori sui gate hanno la funzione di limitare i picchi di corrente creati dalla non trascurabile capacità d'ingresso. Lo switch di finecorsa è meglio che sia ottico, che è più preciso di quello elettromeccanico. Il connettore J3 serve per collegare il sistema a un alimentatore a +24V 0,5 A e a +5 V. Io ne ho usato uno



di recupero da una vecchia stampante Canon.

Il ponticello w1 va rimosso quando Arduino è collegato al pc.

In figura 5 si vede l'attuatore lineare completo di scheda elettronica, la meccanica è notevolmente robusta.

Sul cursore è fissata una sottile linguetta che serve per oscurare la fotocellula (montata su piccolo circuito stampato a destra) quando è arrivato a fine corsa.

Il sistema meccanico in mio possesso ha una corsa utile di circa 634 passi, corrispondenti a uno spostamento massimo di 54,25 mm, con una risoluzione di circa 0,086 mm.

LIBRERIA ARDUINO STEPPER

La libreria di Arduino usa proprio il modo di funzionamento "Two phase-on", infatti la sequenza dei segnali di controllo per motori unipolari (4 fili + 2 comuni) è quella di Tabella 4.

In tabella ho messo anche i pin di Arduino che ho usato per il pilotaggio e i colori dei fili di collegamento del mio motore.

	D9	D10	D11	D12	Pin
	red	blue	black	green	Color
Step	C0	C1	C2	C3	
1	1	0	1	0	
2	0	1	1	0	
3	0	1	0	1	
4	1	0	0	1	

Tabella 4

I parametri da passare alla funzione di settaggio sono i seguenti:

```
Stepper myStepper(stepsPerRevolution, 9,10,11,12);
```

Dove la costante `stepsPerRevolution` rappresenta il numero di passi per giro, 400 nel nostro caso.

Così configurato, la direzione positiva degli step è quella verso il sensore di finecorsa.

La funzione:

```
myStepper.step(1);
```

Fa avanzare di un passo il motore.

Se l'argomento è maggiore di uno, occorre immettere la funzione che imposta la velocità di rotazione in giri al minuto (rpm):

```
myStepper.setSpeed(30); // motor speed in rpm
```

Per invertire il senso di marcia si mette un argomento negativo.

LISTATO DI ESEMPIO

Di seguito un semplice programma di prova. Inizialmente il motore si muove nella direzione del sensore di finecorsa e si arresta al suo raggiungimento con la funzione `RefPos()`. Poi si porta verso circa metà corsa e si sposta di +/- 200 passi e così via.

```

/*
  Linear positioning with unipolar stepper motor
  with limit switch for initial positioning
  G. Carrera, 25 luglio 2017
  */

#include <Stepper.h>

const int stepsPerRevolution = 400;  // steps/revolution of my motor

// initialize the stepper library on pins 9 through 12:
Stepper myStepper(stepsPerRevolution, 9,10,11,12);
int LimitSensor = 8;

void setup() {
  pinMode(LimitSensor, INPUT);
  myStepper.setSpeed(30); // motor speed in rpm
}

void RefPos() {
  // return to reference position (on the slotted sensor)
  while (true) {
    myStepper.step(1); // one step forward
    delay(5);
    if (digitalRead(LimitSensor) == LOW) break;
  }
}

void loop() {
  RefPos();
  delay(1000);
  myStepper.step(-316); // approx mean position
  delay(1000);
  myStepper.step(-200);
  delay(1000);
  myStepper.step(400);
  delay(1000);
}

```

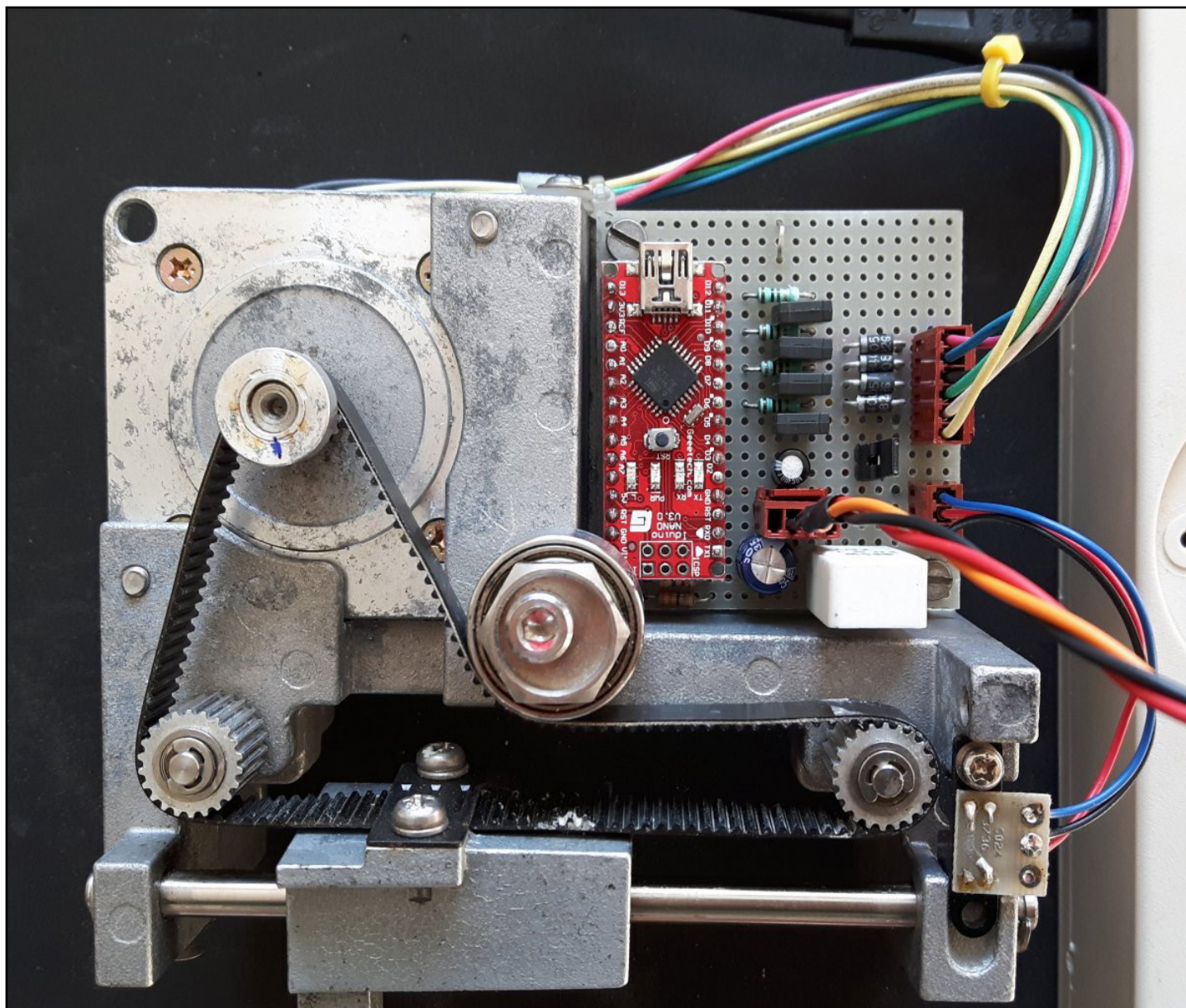


Figura 5- L'attuatore lineare completo di pilotaggio.

Bibliografia

1. "PH265M-33",
datasheet dal catalogo VEXTA;
2. "I motori passo-passo"
Vincenzo Villa, www.vincenzov.net, versione 2.1h - Giugno 2001;
3. "Motore passo-passo"
http://it.wikipedia.org/wiki/Motore_passo-passo



Sono nato a Genova nel 1947, ingegnere elettronico e professore universitario. Appassionato di elettronica sin da ragazzo. Ho praticamente assistito all'evoluzione dei computer a iniziare da quelli con memoria a nuclei magnetici da 32kB e con i disk pack da 80MB fino ai moderni PC con SSD. Dalla fine degli anni '70 ho progettato moltissimi sistemi basati su cpu 6502, 8085, e infine sui PIC e Arduino. Nei primi tempi ho progettato e anche auto-costruito molti sistemi di sviluppo e di

programmazione dei microcontrollori. Ho iniziato a sviluppare il firmware sui 6502 con compilatori Assembler, Forth e Basic per poi passare al Pascal per i PIC e al C con Arduino e derivati. Su PC utilizzo prevalentemente i linguaggi Matlab/ Octave (per elaborazione e analisi dati), Simulink (per la simulazione), ed Excel. Uso il CAD per la simulazione dei circuiti e per la realizzazione degli schemi. I prototipi li realizzo su schede millefori con sottili fili di connessione con isolamento resistente al calore come il Kynar o il PTFE. Negli anni mi sono fatto un laboratorio elettronico molto ben attrezzato che mi permette di eseguire misure accurate sui miei prototipi. Per la ricerca universitaria e per i corsi che tenevo, ho acquisito notevoli conoscenze su strumentazione di bordo e di laboratorio, sensori e attuatori, sistemi di acquisizione e automazione, analisi dati, simulazione di impianti, radiotecnica e GPS. Ho anche progettato moltissime apparecchiature analogiche e digitali per specifici impieghi nel campo della ricerca, come sistemi di monitoraggio strutturale e controllo di servo-attuatori idraulici. Ho anche messo a punto, a partire dal 2002, alcuni sistemi per rilevare i moti nave mediante una rete di ricevitori GPS carrier phase tracking, con accuratezza centimetrica. Dal 2015, ho incominciato a pubblicare, in lingua inglese per avere maggiore diffusione, sul mio blog "ArduPicLab" molte idee e progetti originali, in parte pubblicati anche sui siti americani Hackster.io e Hackaday.io. Da quell'anno ho pubblicato anche nove progetti su riviste italiane, sei su "Fare Elettronica" e tre su "Elettronica Open Source".



Inseguitori di Linea con **Arduino**

di Emanuele Paiano

e.paiano@elettronicaemaker.it

In questo articolo studieremo il funzionamento dei sensori ad infrarossi implementati nel modulo KY-033 (o inseguitore di linea), che permette ad Arduino di distinguere le superfici chiare da quelle scure. Vedremo un'applicazione pratica in cui un semplice robot insegue una traccia nera stampata su una superficie bianca (come può essere un cartoncino o un pannello di legno), oltre che gestire curve, incroci e svolte ad angolo retto.

L'impiego dei sensori, in elettronica e robotica, è piuttosto vasto: si va dal semplice termostato per gli impianti di riscaldamento, fino all'impiego di sonar e sensori infrarossi per il rilevamento di ostacoli e superfici. Grazie ai sensori, è possibile misurare le proprietà fisiche dell'ambiente, permettendo loro di conoscerne lo stato ed eventualmente modificarlo tramite gli attuatori.

IL SENSORE DI TRACCIA: CARATTERISTICHE E TEST

Il modulo KY-033 (noto anche come line tracking sensor) è un sensore che permettere di rilevare il passaggio da una superficie chiara ad una scura. È composto da un comparatore di segnali, un potenziometro per regolarne la sensibilità, e due fotodiodi: il primo fotodiodo è un

trasmettitore di raggi infrarossi, mentre il secondo è un ricevitore. Il principio di funzionamento è basato sulla tendenza degli infrarossi ad essere assorbiti dalle superfici scure: tale funzionamento consiste nell'inviare un segnale dal trasmettitore e attenderne l'eventuale ricezione sul ricevitore, in questo modo possiamo distinguere (a grandi linee) il colore bianco della superficie, da quello nero.

Se la ricezione da parte del fotodiodo ricevitore è andata a buon fine, allora vuol dire che l'infrarosso è stato riflesso su una superficie chiara (figura 1), diversamente, dopo un brevissimo intervallo di tempo, si verifica un timeout e ciò indica che il segnale non è stato ricevuto: in tal caso siamo di fronte ad una superficie scura (figura 2).

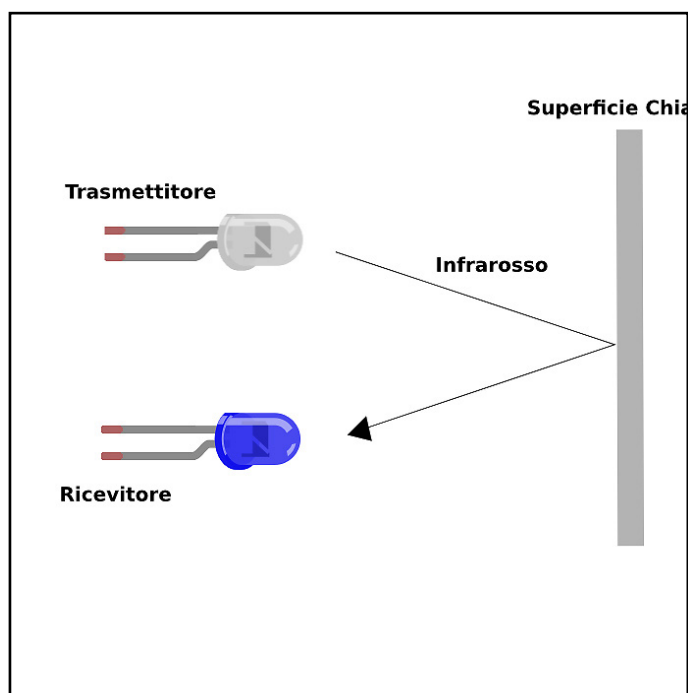


Figura 1: ricezione infrarosso riflesso su una superficie chiara

I fotodiodi non sono gestiti direttamente da Arduino, ma da un comparatore (LM393) integrato nel modulo, il quale confronta il voltaggio generato dal fotodiodo ricevitore con un voltaggio di riferimento.

Il risultato viene inviato sui pin del microcontrollore sotto forma di segnali TTL. In base a tale confronto, l'uscita DO del sensore (Data Output, è l'uscita che andrà collegata ad Arduino) sarà LOW oppure HIGH, ma può anche essere letta come pin analogico, ottenendo un intervallo di possibili valori compresi tra 0 e 1023.

I valori di uscita (analogici e digitali) sono riassunti nella seguente tabella: Per vedere il sensore all'opera, lo possiamo collegare ad un Arduino (nel

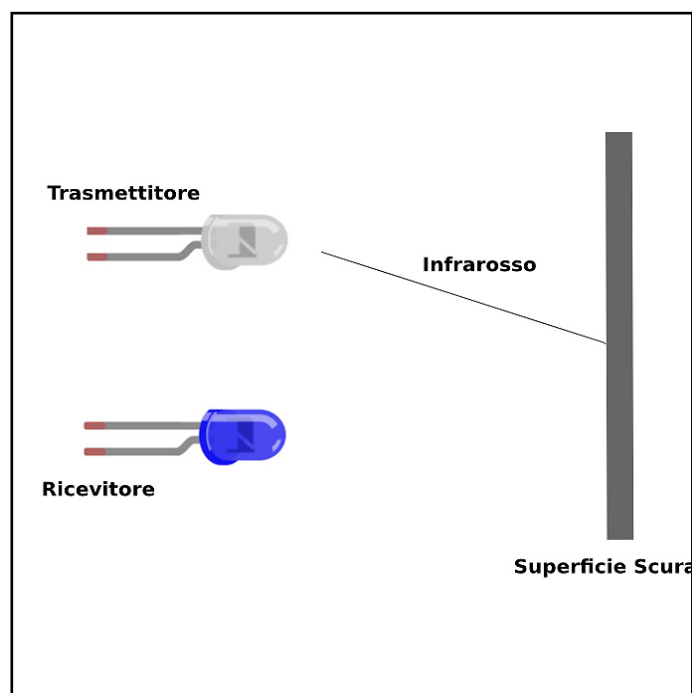


Figura 2: nessuna ricezione, infrarosso assorbito da una superficie scura



nostro caso abbiamo usato la versione Nano, ma è lo stesso sulla UNO) per fare delle letture su diverse superfici. Come illustrato in figura 3, i pin vcc e gnd del

nella figura): in questo modo possiamo leggere contemporaneamente, sia valori analogici che digitali.

Il codice da caricare nella board è riportato

nella pagina seguente. Tale codice è intuitivo: nel metodo loop(), il pin do del sensore viene letto eseguendo il metodo DigitalRead() sul pin 2 di Arduino mentre, tramite

il sensore vanno collegati rispettivamente a 5V e GND di Arduino, mentre il pin do lo colleghiamo all'ingresso digitale 2. Facciamo anche un ponte dal pin2 all'ingresso analogico A0 (filo giallo

con AnalogRead()), leggiamo il valore analogico dal pin A0 (con una pausa di 1 secondo). Infine stampiamo il risultato sulla porta seriale. Il monitor dell'IDE di Arduino va impostato ad una velocità di

SUPERFICIE	TIPO DI LETTURA	USCITA
NERO	DIGITALE	HIGH
NERO	ANALOGICO	MAGGIORE DI 500
BIANCO	DIGITALE	LOW
BIANCO	ANALOGICO	MINORE DI 500

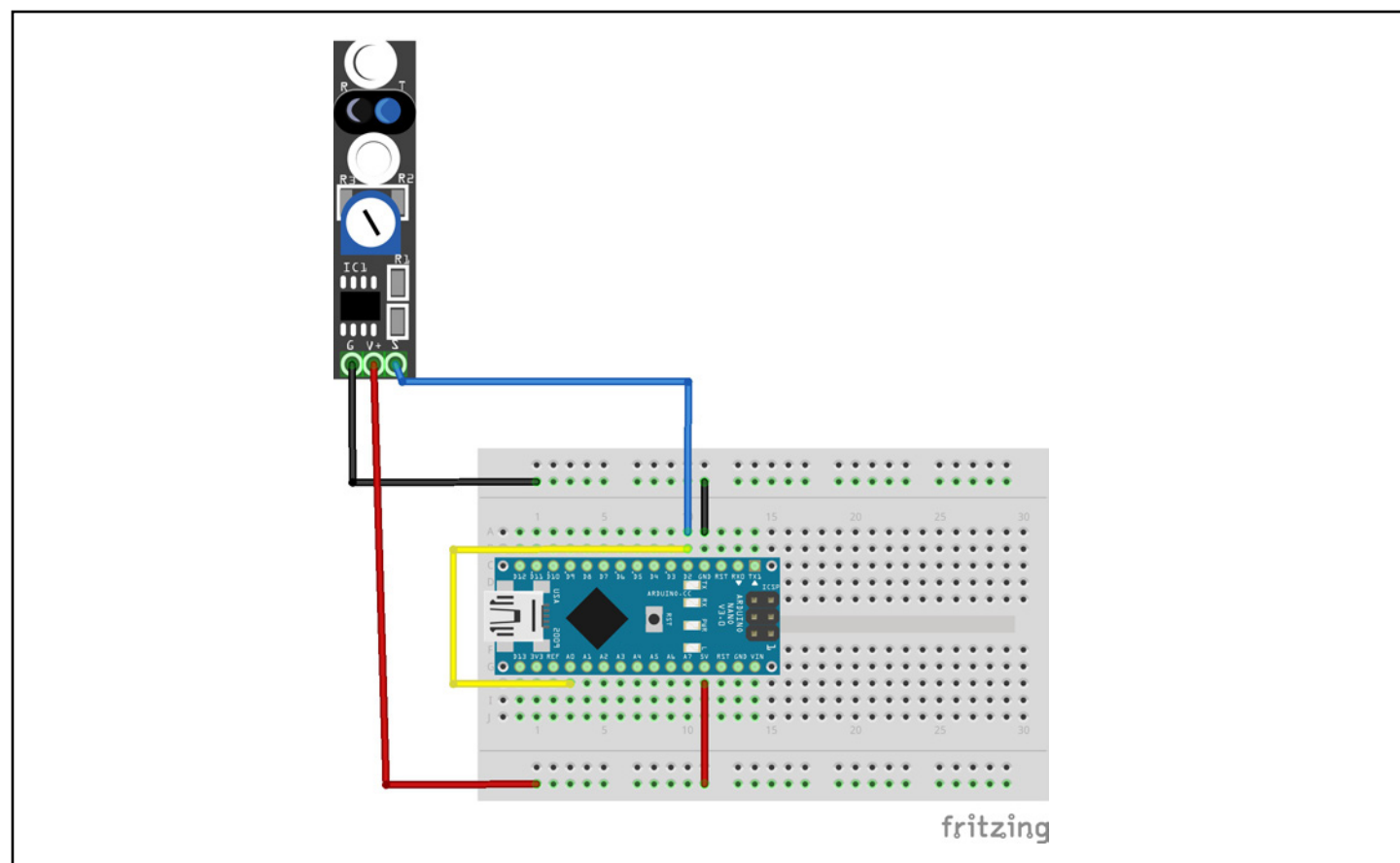


Figura 3: Schema di collegamento per i vari test

```

/* Line Track Sensors */
const int trackDigitalPin=2;
const int trackAnalogPin=A0;
void setup() {
  /* Inizializzo la comunicazione seriale a 19200 bit/sec */
  Serial.begin(19200);
  /* Svuoto il buffer della seriale */
  Serial.flush();
  pinMode(trackDigitalPin, INPUT);
  Serial.println("*****");
  Serial.println(" Arduino KY-033 IR Test *");
  Serial.println("*****");
}
void loop()
{
  delay(1000);
  Serial.print("Lettura Analogica IR: ");
  Serial.println(analogRead(trackAnalogPin));
  Serial.print("Lettura Digitale IR: ");
  Serial.println(digitalRead(trackDigitalPin));
}

```

19200bps.

Grazie a questo schema, nella Figura 4 possiamo vedere le diverse misurazioni in tempo reale), posizionando il sensore su vari tipi di superfici, effettuare dei test, confrontando i risultati con la tabella di valori vista prima.

E' facile notare che il comportamento dei sensori, varia anche in base al tipo di materiale: per esempio sul vetro nero o su un materiale riflettente, si osserva lo stesso risultato che si ottiene sulla carta bianca (così come anche sulle piastrelle del pavimento), quindi la vera differenza nero/bianco si ha su superfici chiare opache, come la carta o il legno.

UN ROBOT CHE INSEGUE UNA PISTA

Passiamo ora a vedere i concetti basilari sul funzionamento del robot, sulla struttura della traccia (stampata su dei fogli in formato A4) e sul posizionamento dei sensori. Per l'esperimento, è stato realizzato il robot 2WD con guida differenziale (quindi dispone di due ruote, un motore CC per ogni ruota), facente uso del modulo L298N collegato ad Arduino Uno. La programmazione del modulo motori e la realizzazione del robot, non verranno discusse in questa sede, ma si darà per scontata l'implementazione delle seguenti funzioni:

- forward(): entrambi i motori in avanti;
- stopLeft(): motore sinistro fermo;
- stopRight(): motore destro fermo;
- reverseRight(): motore destro indietro;
- reverseLeft(): motore sinistro indietro.

Consideriamo la funzione

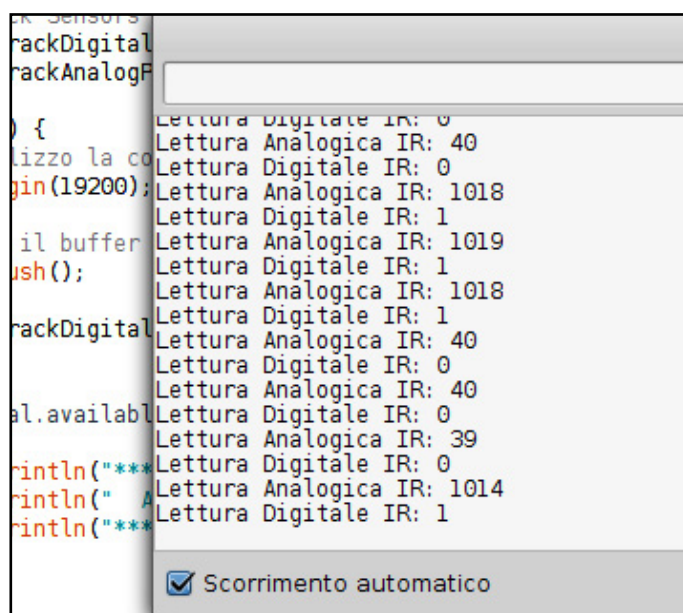


Figura 4: Valori letti tramite il monitor



```
boolean trackOnLeftSensor()  
{  
    return(digitalRead(trackOn-  
LeftPin)==HIGH);  
}
```

la quale ritorna true quando viene rilevata una traccia sotto il sensore sinistro, false altrimenti. In maniera analoga definiamo la funzione che rileva la traccia sul sensore destro:

```
boolean trackOnRightSensor()  
{  
    return (digitalRead(trackOnRi-  
ghtPin)==HIGH);  
}
```

Tali funzioni saranno richiamate all'interno del metodo loop() del codice di Arduino.

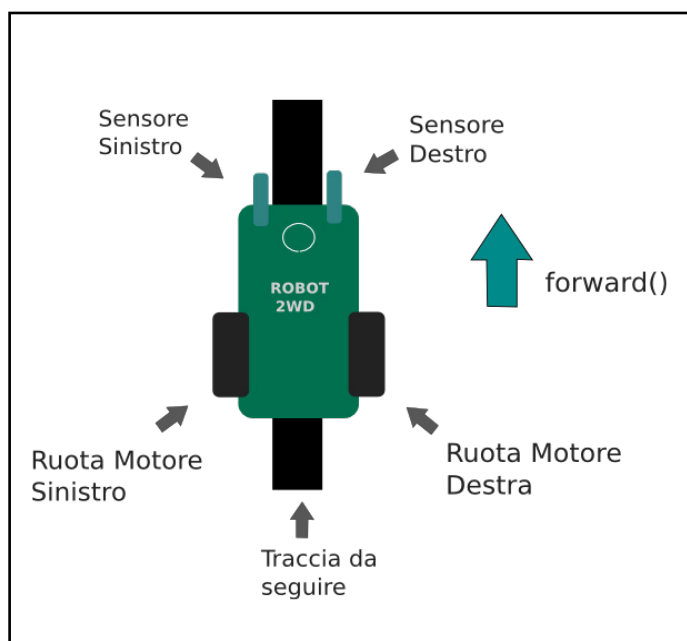


Figura 5: nessuna traccia rilevata, direzione corretta

FUNZIONAMENTO

Studiando il movimento che il robot dovrà fare, notiamo che con due soli sensori a disposizione, abbiamo due possibilità:

- Seguire una striscia bianca su una superficie nera;
- Seguire una striscia nera su una superficie bianca;

Noi faremo riferimento alla seconda opzione, in quanto è più rapida ed economica. Da notare che sono entrambe implementabili allo stesso modo, basta solo invertire la logica dei sensori (scambiare HIGH con LOW nelle funzioni di controllo). L'algoritmo consiste nella continua lettura dei sensori, e nel far variare lo stato dei rispettivi motori non appena viene rilevata la traccia su almeno uno dei due. I sensori saranno posizionati fuori dalla traccia, ai lati del robot, lasciando così la pista nel mezzo (figura 5). In questa situazione, applicheremo la seguente regola: «finchè entrambi i sensori rilevano una superficie chiara, prosegui in avanti». Nel nostro codice, ciò si traduce nell'istruzione

```
if (!trackOnLeftSensor() && !trackOn-  
RightSensor()) forward();
```

ed il robot andrà in avanti, finchè i sensori rilevano una superficie chiara. Se uno dei due sensori finisce sulla traccia, allora vuol dire che quest'ultima sta cam-

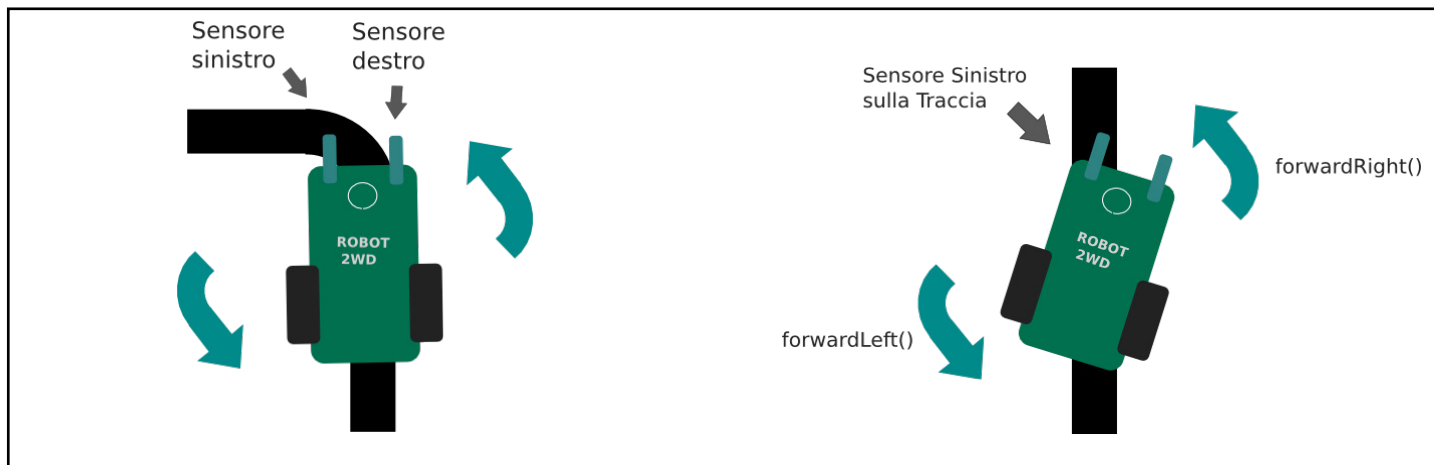


Figura 6: traccia a sinistra, rotazione in senso antiorario

biando direzione, quindi bisogna adattare il robot alla direzione della traccia, effettuando una rotazione finché non si ottiene lo stato precedente. La gestione avviene in questo modo:

- se la traccia è stata rilevata dal sensore sinistro (`trackOnLeftSensor()` ha restituito il valore `true`), vengono richiamate le funzioni `stopLeft()` e `reverseLeft()`, in modo da invertire la rotazione al motore sinistro, lasciando invariata la direzione di quello destro (figura 6). Come risultato, si ottiene una rotazione del robot, seguita da `forward()` su entrambi i motori quando i sensori rilevano nuovamente la superficie bianca. Tale azione è gestita nel codice di Arduino dalle istruzioni

```
if(trackOnLeftSensor() && !trackOn-
RightSensor())
{
  stopLeft();
  reverseLeft();
}
```

- se invece la traccia è stata rilevata dal sensore destro (`trackOnRightSensor()` ha restituito il valore `true`), vengono richiamate le funzioni `stopRight()` e `reverseRight()`, in modo che il motore destro vada all'indietro (il motore sinistro continua a girare in avanti), per poi rieseguire `forward()` su entrambi i motori quando i sensori tornano sulla superficie bianca (figura 6). Anche in questo caso, dobbiamo aggiungere le rispettive istruzioni nel codice:

```
if(!trackOnLeftSensor() && trackOn-
RightSensor())
{
  stopRight();
  reverseRight();
}
```

Utilizzando questo approccio, il nostro robot è in grado di gestire anche le svolte ad angolo retto e alcune curve più difficili, con un andamento di tipo «zig zag». Un miglioramento senza zig zag, può essere ottenuto con altri sensori,



posizionati nel mezzo e programmati appositamente per aumentare la precisione. Il risultato ottenuto, sarà simile a quello visibile in questo video.

A questo punto, potremmo chiederci:

- c'è un punto in cui il robot deve fermarsi?
- come gestire un incrocio?

Possiamo pensare di far fermare il robot quando entrambi i sensori rilevano la traccia, richiamando il metodo `stopMotors()`:

```
if(trackOnLeftSensor() && trackOnRightSensor()) stopMotors();
```

Ovviamente ciò non sarebbe efficiente, in quanto entrambi i sensori potrebbero

finire (per qualche secondo) sulla traccia di una lunga curva, che verrà erroneamente scambiata per incrocio. Sa-

rebbe opportuno leggere i valori per un certo intervallo di tempo, prima di avviare la funzione di stop.

L'idea è quella di includere una variabile contatore (nel codice l'abbiamo chiamata `linesCount`) che si incrementa ogni volta che entrambi i sensori rilevano la traccia, per poi reimpostarsi a zero ogni volta che si rileva la superficie bianca. Se la variabile raggiunge un certo valore, possiamo dire al robot di fermarsi, in quanto siamo certi che non si tratti di una curva o un semplice incrocio da ignorare, ma di una «piazzola di sosta» (figura 9):

```
if(trackOnLeftSensor() && trackOnRightSensor())  
{  
    linesCount++;  
    if (linesCount>6000)
```

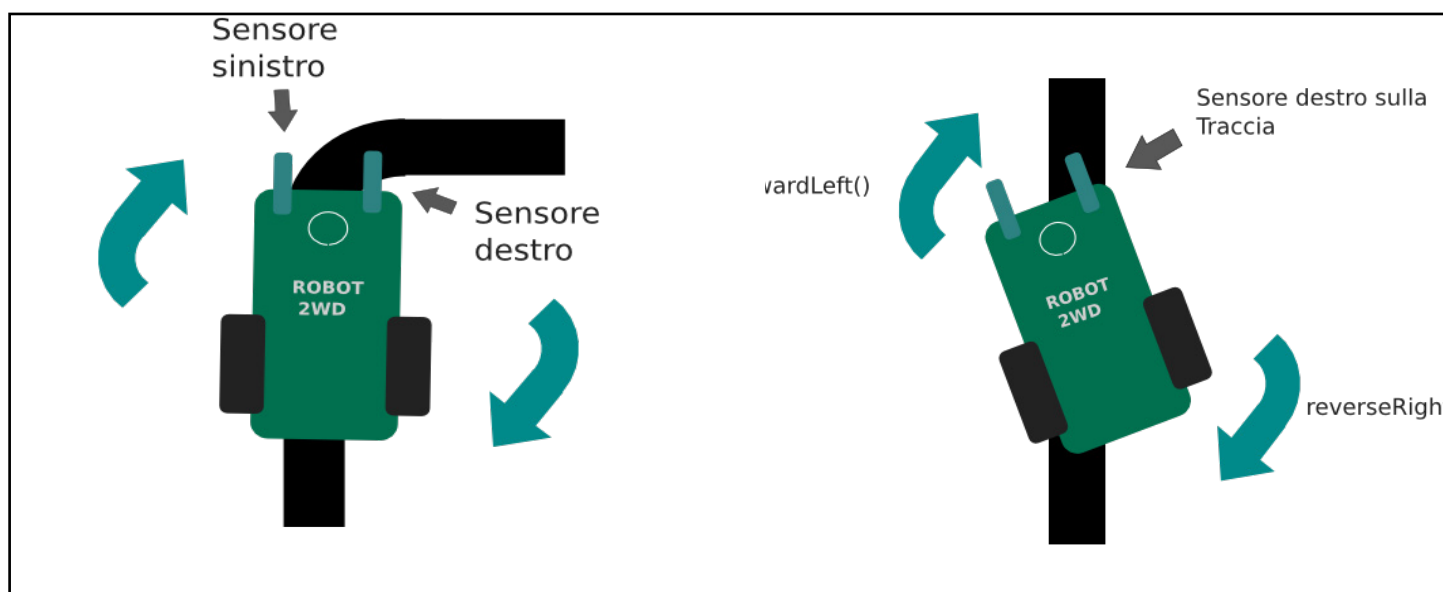


Figura 7: traccia a destra, rotazione in senso orario

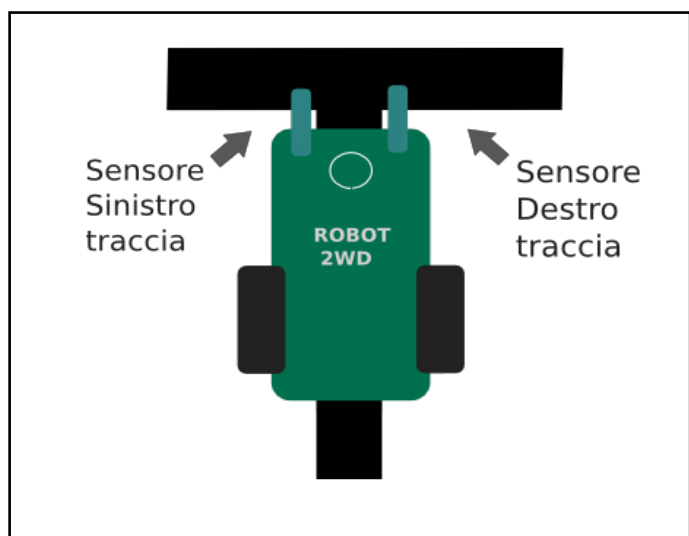


Figura 8: incrocio, entrambi i sensori rilevano la traccia

```
{
  stopMotors();
}else forward();
}
```

in questo modo, quando si ha una traccia a destra e una a sinistra, gli incroci saranno ignorati (il robot proseguirà dritto in alcuni casi, mentre sterzerà se arriverà troppo inclinato e uno dei due sensori rileverà la traccia per primo), ma sicuramente si fermerà ai box neri.

POSIZIONAMENTO DEI SENSORI, COLLEGAMENTO E SKETCH

È preferibile montare i sensori nella parte anteriore del telaio e possibilmente un pò distanti dalla traccia (poco più di mezzo cm). È bene tener presente che, maggiore è tale distanza, maggiore sarà l'effetto «zig zag».

Nel nostro caso, la distanza di un sensore dall'altro è di 8 cm (in quanto la traccia è di circa 5 cm). Non ci sono del-

le misure rigide da rispettare, ma quelle descritte qui sono frutto di varie sperimentazioni, quindi sicuramente funzionanti. In figura 10 è illustrato lo schema (frontale) del nostro robot e le relative misure.

Alcuni sensori devono essere posizionati ad un'altezza massima di 3 cm dalla superficie bianca, mentre altri possono arrivare anche a 5: potete fissarli al telaio utilizzando dei distanziali reperibili in una ferramenta. Ricordiamo che i motori sono gestiti da un modulo L298N collegato ad un Arduino, il tutto alimentato da una batteria al litio da 7,4V.

Il collegamento dei sensori si effettua come mostrato in figura 11. Negli allegati potete trovare lo sketch di Arduino, comprese anche le funzioni e variabili per la gestione del modulo, mentre riportiamo per intero il codice per la sola gestione dei sensori:

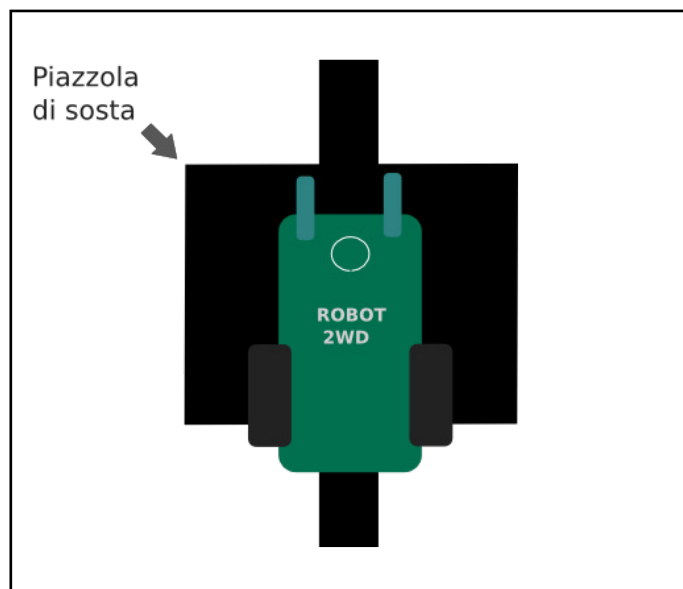


Figura 9: piazzola, rilevata traccia continua

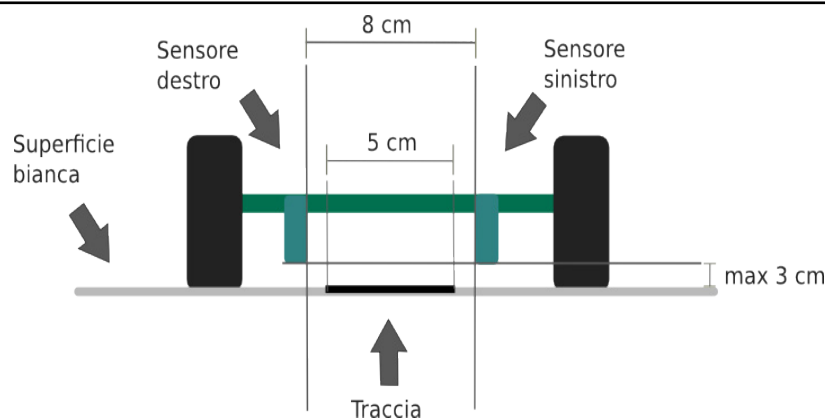


Figura 10: Misure delle traccia e posizionamento dei sensori

```

/* Line Track Sensors */
const int trackLeftPin=A1;
const int trackRightPin=A0;
/* Linee nere su entrambi i sensori */
float linesCount=0;
/* Omesse variabili Pin L298N */
void setup() {
  /* Inizializzo la comunicazione seriale usb a 19200 bit/sec */
  Serial.begin(19200);
  /* Svuoto il buffer della seriale */
  Serial.flush();
  pinMode(trackLeftPin, INPUT);
  pinMode(trackRightPin, INPUT);
  /* Omesse inizializzazione dei pin L298N */
  /* Imposto velocità iniziale dei motorini */
  analogWrite(enA, 90);
  analogWrite(enB, 90);
  forward();
}
void loop() {
  /* Verifica se la traccia è solo sul sensore destro */
  if(!trackOnLeftSensor() && trackOnRightSensor()) {
    stopRight(); reverseRight();
  }
  /* Verifica se la traccia è solo sul sensore sinistro */
  if(trackOnLeftSensor() && !trackOnRightSensor())
  {
    stopLeft();
    reverseLeft();
  }
  /* Verifica se la traccia è su entrambi i sensori */

```

```

if(trackOnLeftSensor() && trackOnRightSensor())
{
  linesCount++;
  if (linesCount>6000)
  {
    stopMotors();
  }else forward();
}
/* Verifica se entrambi i sensori NON sono sulla traccia */
if(!trackOnLeftSensor() && !trackOnRightSensor())
{
  linesCount=0;
  forward();
}
}
/* c'è linea nera sotto il sensore sinistro */
boolean trackOnLeftSensor()
{
  return (digitalRead(trackLeftPin)==HIGH);
}
/* c'è linea nera sotto il sensore destro */
boolean trackOnRightSensor()
{
  return (digitalRead(trackRightPin)==HIGH);
}
/* c'è linea nera sotto entrambi i sensori */
boolean trackOnBothSensor()
{
  return trackOnLeftSensor() && trackOnRightSensor();
}
/* Omesse funzioni L298N */

```

STRUTTURA E REALIZZAZIONE DELLA TRACCIA

È possibile costruire una pista mettendo insieme dei fogli in formato A4 con pezzi di traccia stampati sopra, attaccandoli con colla o nastro adesivo. Negli allegati è possibile trovare immagini in formato png e jpeg, già pronte per essere stampate, ma si possono trovare anche file .svg da modificare a piacere con Inkscape. I pezzi di traccia sono:

- rettilineo: una semplice linea retta;
- svolta a destra: una svolta a 90 gradi verso destra;

- svolta a sinistra: svolta a 90 gradi verso sinistra;
- curva destra: consente di far girare a destra il robot mediante una curva;
- curva sinistra: come prima, ma verso sinistra;
- piazzola: area di sosta, permette al robot di fermarsi;
- incrocio: consente di realizzare svolte multiple oppure piste a forma di «otto» (come le piste polistil per bambini);
- bivio: una svolta a destra o a sinistra, può essere utilizzato per altri casi non descritti qui;

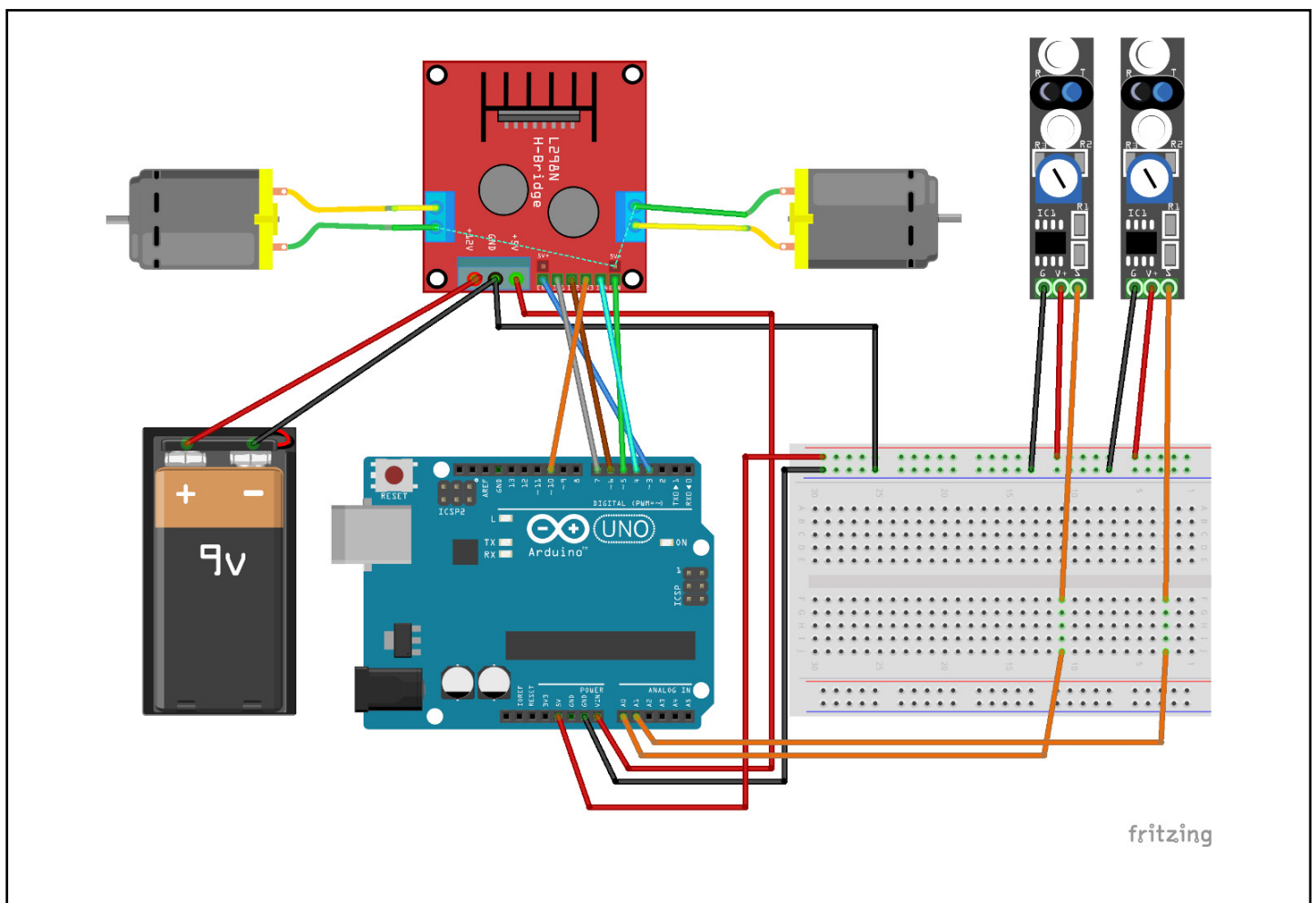


Figura 11: Collegamento dei sensori al robot

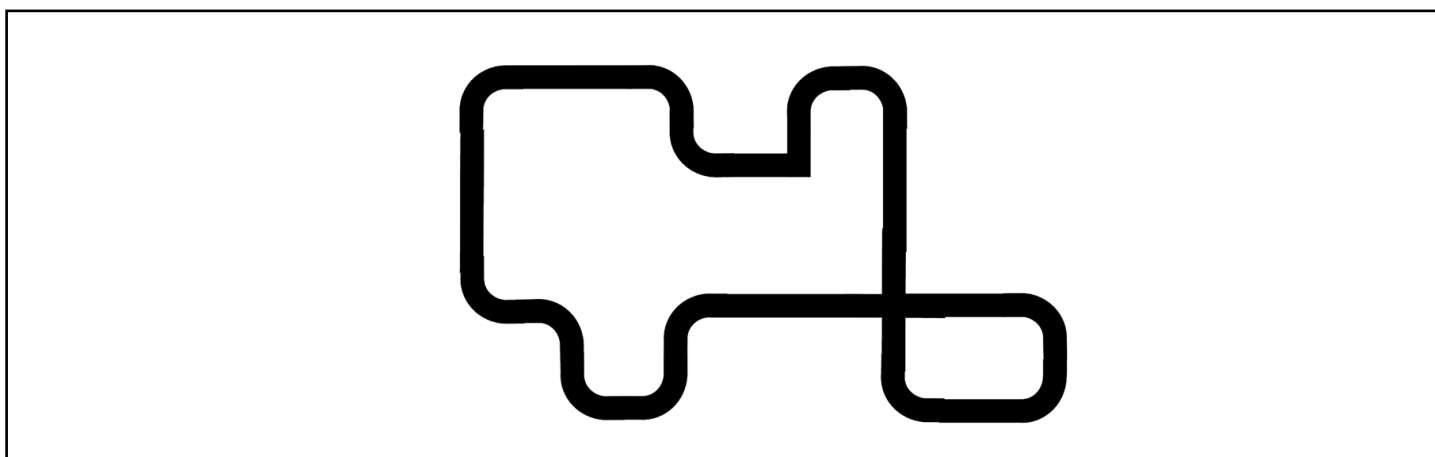


Figura 12: Esempio di pista su cui testare il robot

- curva destra alta: curva destra con rettilineo;
- curva sinistra alta: curva sinistra con rettilineo;

Con questi pezzi si possono ottenere tante combinazioni di pista, tra cui la pista mostrata in figura 12: A causa della svolta ad angolo retto e delle improvvise curve, tale pista è gestibile se il robot si muove con una velocità limitata.

Nel nostro caso, alimentando il tutto con una batteria lipo a 7.4 V, abbiamo impostato il pwm dei motori a 90 e l'andamento appare piuttosto stabile: spesso l'incrocio viene ignorato, in quanto il robot prosegue dritto (grazie al controllo della variabile `linesCount`), e difficilmente esce fuori dalla traccia.

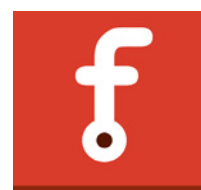
CONCLUSIONI

Quanto realizzato in questo progetto, può essere implementato come una funzionalità aggiuntiva nei robot visti nel numero speciale Arduino Projects.

Infine ci ritroveremo con un robot inseguitore di linea: i più esperti, potranno aggiungere dei sensori che, leggendo delle strisce nere aggiuntive ai lati della traccia, possono interpretarle come segnali in prossimità di un bivio o un incrocio e «lasciando» al robot, la possibilità di decidere in anticipo cosa fare. I limiti di ciò che si può fare, dipendono solo dalla fantasia del costruttore. [Clicca qui sotto per il download degli Sketch](#)



e qui per i circuiti Fritzing





Appassionato di tecnologia dall'età di 14 anni e laureato in Informatica, i miei principali campi di interesse sono i sistemi operativi unix-like (Linux/

BSD e la shell Bash), alcuni framework per lo sviluppo (linguaggi Java, Python e Php), Arduino, Raspberry e dispositivi IoT. I ruoli che ho ricoperto in passato riguardano l'assistenza tecnica, la consulenza informatica nonché l'amministrazione di reti e server Linux. Tra gli studi da autodidatta si trovano l'elettronica e la robotica amatoriale, con sperimentazione nella realizzazione di prototipi basati su Arduino e Raspberry Pi. Per informazioni: <https://emanuelepaiano.github.io>



LOGIN su WEB

con un Pulsante

di Antonello Della Pia

a.dellapia@elettronicaemake.it

PRIMA PARTE

Un dispositivo delle dimensioni di una chiavetta USB, che tramite un unico pulsante ed un firmware personalizzabile, ricorda ed inserisce per noi le credenziali di accesso (nome utente e password) ai siti e servizi Internet preferiti.

Oggigiorno Internet ed il Web ci mettono a disposizione una quantità di informazioni e servizi ai quali è ormai impossibile rinunciare. Un aspetto critico emerso quasi subito e rivelatosi di importanza sempre maggiore è la riservatezza e la protezione dei dati personali degli utilizzatori della rete. Ci si è accorti ben presto che sia un banale scambio di messaggi personali quanto una transazione economica internazionale, se intercettati dalle persone sbagliate, potevano provocare danni anche gravi ai soggetti interessati.

Si è reso necessario quindi studiare ed applicare molteplici livelli di sicurezza e di protezione alla rete ed il primo di questi, che permette all'utente finale di accedere in sicurezza alla stragrande maggioranza dei servizi, è l'autenticazione tramite nome utente e password.

Una password sicura tuttavia rischia di essere complessa e quindi difficile da ricordare, se rispetta le linee guida che ormai tutti dovremmo conoscere. Alzi la mano invece a questo punto, chi non ha mai usato i nomi dei figli, dell'animale domestico, la data di nascita come password o peggio di tutto, la stessa password per tutti i servizi! Il dispositivo che propongo può secondo me contribuire in modo significativo a liberarci dall'incubo e dalla schiavitù di avere password efficaci perché le ricorda al posto nostro!

DALL'IDEA ALLA REALIZZAZIONE

Navigando in rete possiamo constatare come il problema delle password sia stato già affrontato in vari modi e con proposte più o meno complesse a livello software, hardware, misto o con servizi dedicati. Soluzioni che non hanno trovato però, a mio parere, la diffusione che ci si poteva attendere ed ognuna, come spesso accade, con pro e contro. Come autocostruttore, era mia intenzione realizzare qualcosa di semplice, personalizzabile, con un valore anche didattico, comprensibile e replicabile da

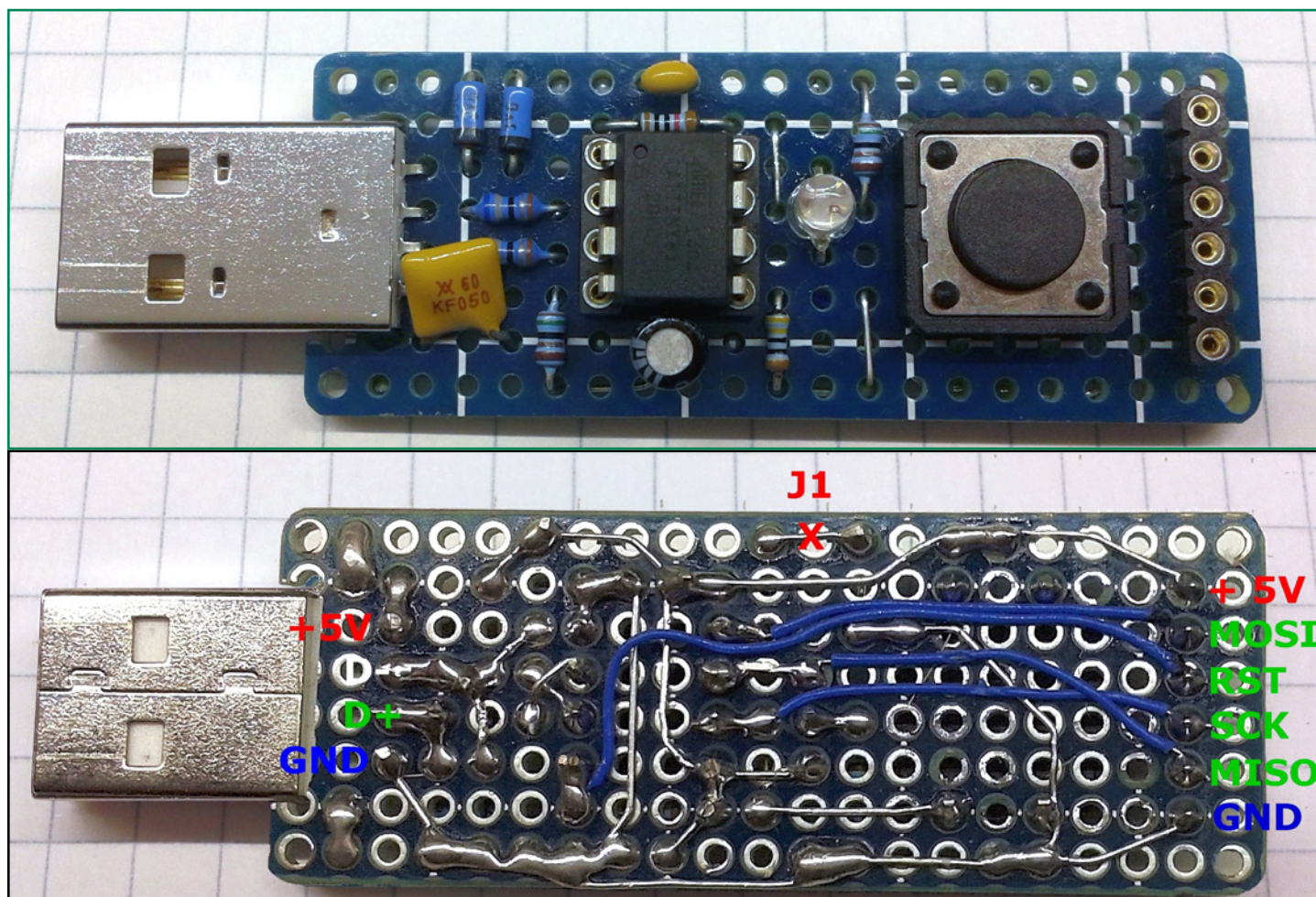


Figura 1: Il prototipo terminato su millefori

Figura 2: lato saldature. In evidenza i collegamenti esterni



altri appassionati.

L'ispirazione è venuta da un esempio di tastiera virtuale USB riportato sul noto sito Adafruit (1).

Per chi non lo conoscesse, potrei descriverlo come "il paradiso del Maker", potendovi trovare un assortimento incredibile di componenti, moduli, accessori, board, quasi sempre corredato da esempi di applicazione, librerie e tutorial utilissimi.

Nel nostro caso, per la piccola board Trinket, basata sul microcontroller ATtiny85, viene fornito un esempio (completo di libreria liberamente scaricabile ed utilizzabile) di periferica USB HID (Human Interface Device) capace di inviare ad un PC una sequenza di tasti prestabilita, da codice, emulando una comune tastiera (2). Da qui a pensare di inviare al computer la sequenza composta da nome utente, password, tasti Tab e Invio e compilare quindi automaticamente un form di login, il passo è stato breve!

Per rendere l'articolo più scorrevole, ho pensato di rivolgermi in prima battuta agli appassionati più esperti e smaliziati, ai "duri" cioè, che chiedono solo uno schema, un elenco componenti, il firmware e sanno già cosa farne. Successivamente, in un riquadro apposito, cercherò di approfondire gli aspetti meno scontati, così da dare anche una valenza didattica al progetto e mettere

in condizioni di realizzarlo anche chi si fosse limitato finora al solo uso "classico" dell'ambiente Arduino.

LA PARTE HARDWARE

Trinket si potrebbe definire come una micro scheda Arduino compatibile. La sua caratteristica più interessante è la possibilità di dialogare con un PC tramite porta USB. Il protocollo USB si avvale in questo caso dell'implementazione cosiddetta firmware-only V-USB, un progetto ingegnoso e poco conosciuto, che permette di aggiungere l'ormai universale interfaccia USB anche ai piccoli microcontrollori AVR di Atmel che ne sono sprovvisti a livello hardware,.

Consiglio una visita al sito relativo (3) per ulteriori informazioni, dettagli e interessanti esempi di utilizzo di V-USB. Un'altra caratteristica peculiare di Trinket è il cosiddetto bootloader, cioè quella parte di firmware che, similmente ad Arduino, permette di caricare lo sketch direttamente dall'ambiente di programmazione (Arduino IDE), via USB. Noi non lo utilizzeremo in quanto soluzione proprietaria e caricheremo il nostro codice con un comune programmatore USBasp, come vedremo in seguito.

Riassumendo, non utilizzeremo per il progetto la scheda Trinket completa, ma il singolo microcontroller Attiny85, pochi componenti di contorno e la libreria TrinketKeyboard che grazie a V-USB

ci permetterà di simulare una tastiera virtuale USB HID riconosciuta anche da Windows 10 senza bisogno di driver.

In questa maniera, a fronte di un percorso un po' più complesso, avremo la possibilità di personalizzare al massimo la realizzazione ed acquisire conoscenze utili anche per altri progetti che utilizzino microcontrollori in modalità stand-alone. Il prototipo è stato realizzato, come mia abitudine, su basetta millefori, con forma e dimensioni che richiamano quelle di una chiavetta USB. Tra i pochi componenti, tutti a foro passante, spiccano il pulsante che rappresenta l'unica interfaccia utente ed un LED bicolore che indica lo stato del dispositivo. I resistori, per contenere le dimensioni, appartengono ad una serie speciale, più picco-

la del solito, ma reperibile abbastanza facilmente. Come buona pratica è stato previsto un fusibile di tipo ripristinabile da 500 mA a protezione delle porte USB del computer da eventuali guasti o cortocircuiti non impossibili specialmente in fase di test. A questo proposito è consigliabile, a realizzazione ultimata, prevedere l'isolamento della basetta, per esempio con guaina termorestringente trasparente oppure realizzando un contenitore ad hoc in materiale isolante. Un connettore personalizzato a sei contatti (connettore ISP ovvero In-system programming oppure ICSP In-Circuit Serial Programming) ricavato da una strip permette il collegamento del programmatore USBasp. Ho scelto questa opzione perché esteticamente migliore e meno

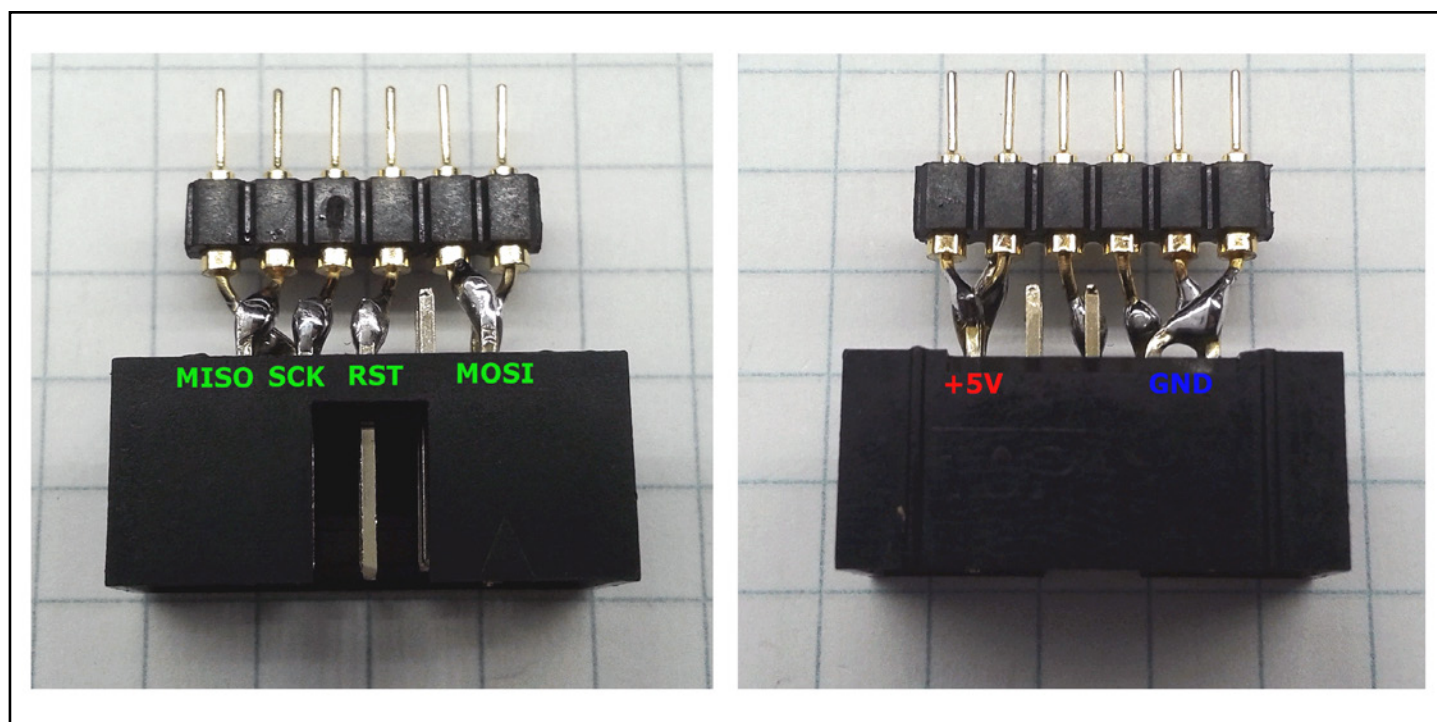


Figura 3: Adattatore per programmatore USBasp



invasiva rispetto al connettore standard IDC a dieci o sei contatti di serie sui programmatori di questo tipo.

Questa soluzione "custom" richiede un adattatore che si può ottenere saldando direttamente sei contatti maschio strip ad un connettore a dieci contatti maschio IDC da circuito stampato, rispettando ovviamente l'ordine dei collegamenti. Questo è possibile perché dei dieci contatti, solo sei sono realmente utilizzati (MISO, MOSI, RESET, SCK, GND, +5V). Le immagini dovrebbero fornire informazioni sufficienti alla corretta realizzazione dell'adattatore. Il programmatore USBasp (acquistabile per pochi euro in rete) riveste importanza fondamentale nel progetto, in quanto oltre alla funzione classica di program-

mazione iniziale del microcontrollore, servirà a ricaricare lo sketch aggiornato in caso di variazione oppure aggiunta di password. Può sembrare complicato, ma vedremo come predisporre e configurare al meglio l'ambiente necessario.

LO SCHEMA ELETTRICO

Lo schema elettrico è molto semplice. I due diodi Zener ed i tre resistori a valle del connettore USB tipo A maschio rappresentano la configurazione hardware tipica richiesta dalla soluzione firmware-only V-USB. I due diodi ed i resistori da $68\ \Omega$ limitano la tensione dei segnali in uscita dal microcontrollore (alimentato a 5 Volt) ad un valore prossimo ai 3,3 Volt standard delle linee dati USB. Il resistore da $1,5\ K\Omega$ detto di pull-up per-

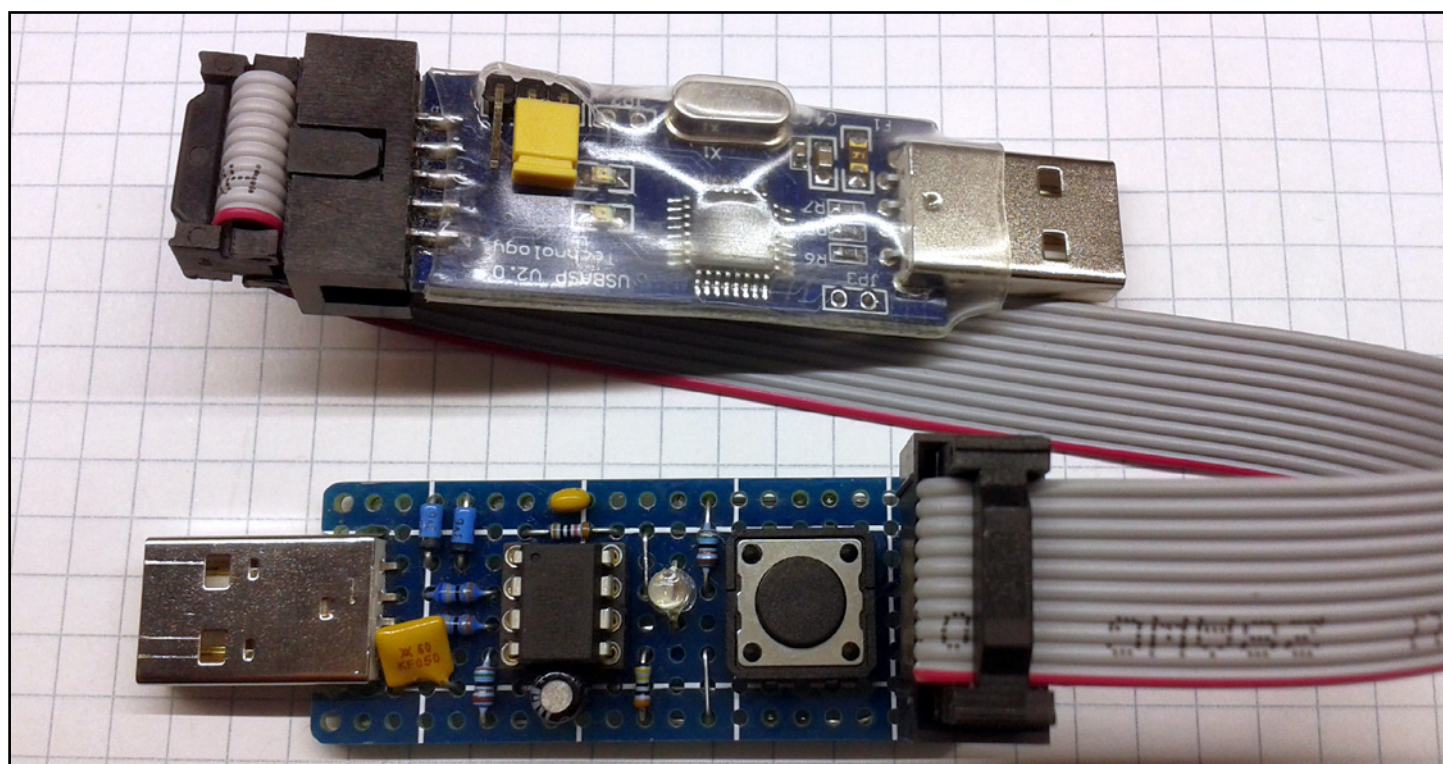


Figura 4: Il circuito collegato al programmatore USBasp



Elenco componenti

R1	330 Ω 0,4 W 1% es. serie Vishay MRS16
R2	10 KΩ 0,4 W 1%
R3	1,5 KΩ 0,4 W 1%
R4	68 Ω 0,4 W 1%
R5	68 Ω 0,4 W 1%
R6	1,5 KΩ 0,4 W 1%
C1	100 nF 50 V ceramico
C2	10 μF 10 V elettrolitico
U1	ATtiny85-20PU con zoccolo 8 pin
D1	LED 3 mm bicolore rosso/blu 3 terminali catodo comune
D2, D3	Diodo Zener 3,6V 0,5W es. BZX55C3V6
F1	Fusibile ripristinabile 500 mA
S1	Pulsante normalmente aperto
X1	Connettore USB-A maschio per PCB
JP1	Contatti strip femmina 6 poli
Altro	Basetta millefori
Altro	Contatti strip maschio 6 poli
Altro	Connettore IDC 10 poli maschio per PCB
Altro	Programmatore USBasp con cavo 10 poli

classici, mentre R1 evita il corto circuito verso massa del terminale MOSI con J1 chiuso, cosa che impedirebbe la programmazione.

J1 rappresenta un ponticello da aprire solo se il PC ha come configurazione predefinita lingua e tastiera inglese (U.S.), da lasciare chiuso in ogni altro

caso. Ne vedremo il motivo analizzando il software.

IL SOFTWARE

Passiamo ora ad analizzare il codice del progetto.

Per la scrittura e la compilazione del programma è stato utilizzato il classico ambiente di sviluppo integrato (IDE) di Arduino, versione 1.6.12. La scheda da installare e selezionare è "Adafruit Trinket 16MHz", programmatore "USBasp". L'impostazione dei fuse per l' ATtiny85 è: L:0xF1, H:0xD5, E:0xFE.

Lo sketch è abbastanza lungo anche se non particolarmente complesso, ma completamente commentato, per cui rimando al codice stesso (allegato all'articolo) per una spiegazione dettagliata. Focalizzeremo qui invece l'attenzione sulle sezioni più originali e specifiche del progetto.

Si nota per prima cosa che lo sketch è diviso in due file, Login_con_un_pulsante.ino, file principale, e form_data.ino. L'intenzione è di separare per così dire il "motore del programma" dai dati, semplificando il compito all'utente che debba modificare o aggiornare successivamente i dati inseriti.

- **Continua sul Prossimo Numero** -



L'elettricità mi ha sempre attirato, da giovane dilapidavo la paghetta in batterie, lampadine e cavi elettrici, a undici anni ho chiesto in regalo il primo multimetro (un glorioso ICE analogico). Anche dopo il diploma di Perito Industriale Elettrotecnico e nonostante un successivo percorso professionale piuttosto eterogeneo, la passione per l'elettronica è rimasta una costante,

arrivando nel tempo a comprendere l'interesse per audio e diffusori, misure elettroacustiche, circuiti di alimentazione e amplificatori, circuiti analogici e digitali a componenti discreti, amplificatori operazionali, tubi termoionici, uso di strumentazione e misure di laboratorio, fino ad arrivare, di recente, al "morbo" della breadboard, dei microcontrollori e relativa programmazione. Insomma, tutto il classico repertorio del saldatore-dipendente! Strada facendo si è inoltre inserita prepotentemente la passione per l'informatica (sfociata poi in sedici anni di attività professionale), che ha rappresentato la quadratura del cerchio, grazie alle risorse quasi infinite improvvisamente disponibili per un appassionato come me. Basti pensare alla reperibilità di componenti e informazioni, alla possibilità di studiare ed applicare diversi linguaggi di programmazione, di disporre di un laboratorio di misura anche virtuale a costi accessibili, alla potenza e comodità dei simulatori, alla attuale realtà dell'Internet of Things. Credo quindi, attualizzando il concetto, di potermi definire un "maker" di 55 anni, senza pretese di genialità, ma con interessi e competenze in diversi ambiti, specialmente pratiche, acquisite essenzialmente da autodidatta. Quando non mi occupo di elettronica, mi piace leggere (soprattutto narrativa, preferibilmente di autori italiani), ascoltare musica (quasi esclusivamente inglese, di artisti fuori dal circuito commerciale), andare in bici, camminare.



TELECONTROLLO IoT per Condizionatori

di Girolamo D'Orio

g.dorio@elettronicaemaker.it

I moderni condizionatori d'aria hanno raggiunto un notevole grado di perfezionamento tecnico nel controllo climatico degli ambienti, ma sono ancora piuttosto carenti per quanto riguarda la possibilità di un efficace controllo remoto. In questo articolo presento un dispositivo IOT universale che permette l'accensione e spegnimento, tramite telecomando, di un dispositivo del genere.

Questa realizzazione nasce dall'esigenza di un mio caro amico Geometra, che per esigenze di lavoro si trova spesso all'esterno per fare dei rilievi. Quindi non ha orari precisi di rientro in ufficio, ma gradirebbe trovare un clima confortevole al suo ritorno.

Il suo ufficio è composto da almeno tre stanze, delle quali ognuna ha una postazione ben precisa. Un giorno lavora in una, un giorno in un'altra...

Ogni stanza ha un proprio Split, quindi la sua richiesta doveva rispondere a queste particolari esigenze:

1. Voglio spendere poco...
2. Con un dispositivo li vorrei pilotare tutti e tre.
3. Possibilmente senza passare tanti fili da una parte all'altra dell'ufficio.
4. Pratico da utilizzare.

5. Deve essere pilotato dal mio Smartphone, ovunque mi trovi.
6. Voglio sapere, se possibile la temperatura ambientale, in quanto voglio decidere se sia il caso di accendere il condizionatore o meno.

Bene, gli ho risposto.... Hai un telecomando da sacrificare?

La risposta è stata affermativa in quanto ne aveva due che non funzionavano più, a causa di cadute dalla scrivania.

Riparato il telecomando, il quale aveva una saldatura distaccata sulle lamine di appoggio delle batterie stilo, mi sono messo a copiare il codice tramite Arduino. Codice che, tra l'altro, si è rivelato piuttosto complesso rispetto a quello di un tradizionale telecomando tv. La codi-

fica del telecomando dei condizionatori è molto lunga, perché anche se si preme il solo tasto accensione, il dispositivo deve inviare ogni volta tutti i settaggi di impostazioni come temperatura, modalità estate/inverno/velocità della ventola.

Riflettendo con calma, mi sono chiesto: "ma se ho il telecomando che già fa tutto questo lavoro, perché non lo interfaccio ad un modulo Wi-Fi come l'ESP8266-01?" Lo rendo alimentabile tramite USB, così il mio amico può decidere se alimentarlo con il carica-batterie dello Smartphone o con un Power-Bank.

Così se lo può spostare da una scrivania all'altra senza creare troppi problemi. Basta indirizzarlo verso il condizionatore ed il gioco è fatto!

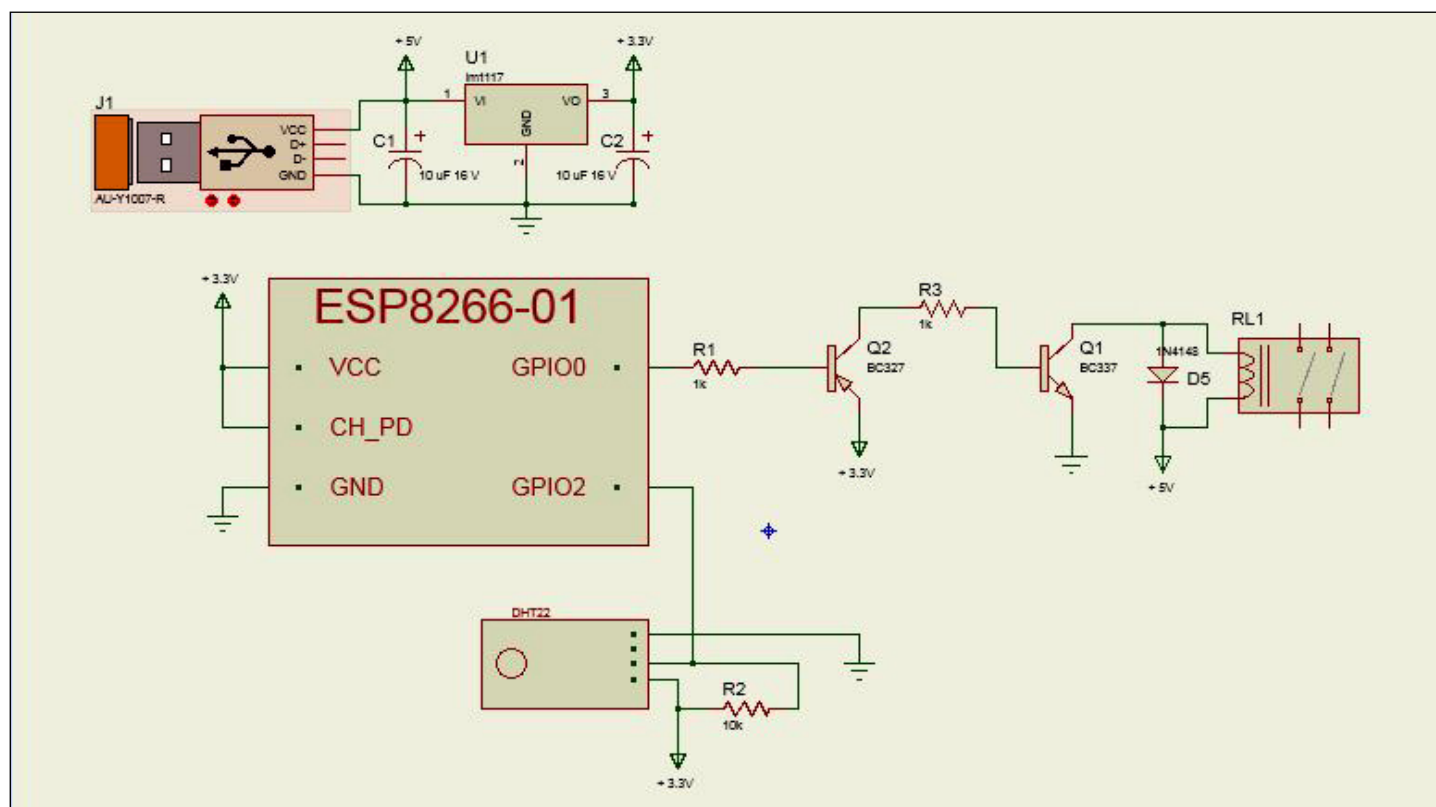


Figura 1: Schema elettrico del circuito

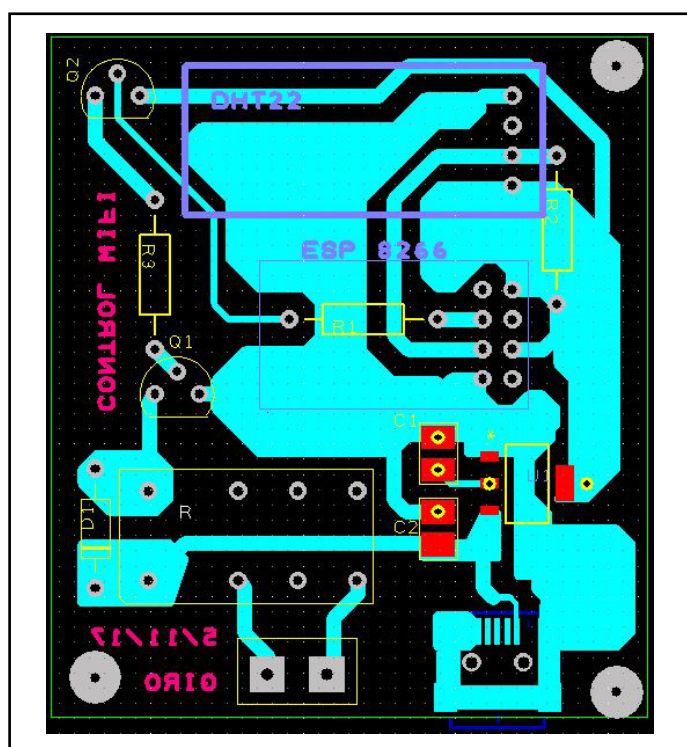


Figura 2: Layout del circuito stampato

SCHEMA E DESCRIZIONE HARDWARE

Come vedete è veramente semplice, alla portata di tutti. L'alimentazione è fornita tramite USB (5Vcc) quindi per alimentare correttamente il modulo ESP8266-01 a 3,3V mi sono avvalso del regolatore lineare Lm1117 che essendo un Low Drop-Out lavora bene anche se la sua tensione di ingresso è poco superiore a quella di uscita. Le due capacità da 10µF sono in SMD, purtroppo le ho reperite così perché non sono riuscito a trovare di meglio e con poca spesa due elettrolitici a LOW ESR, in quanto sono espressamente richiesti consultando il datasheet del regolatore usato.

Nonostante le piccole dimensioni, si saldano tranquillamente bene sul lato piste con un normale saldatore. Il sensore

DHT22 con la sua resistenza di pull-up sul pin Data è collegata sul GPIO2 di ESP8266-01 non a caso. Il sensore una volta alimentato ha un invio di livello logico basso di segnale, se collegato al GPIO1 manderebbe il modulo in configurazione di programmazione. Operazione necessaria se si vuole flashare il micro a bordo del modulo, come faremo in seguito.

Perché la scelta del relè? Proprio per rendere facile la realizzazione in tutte le occasioni, non sapendo che tipo di telecomando potete avere. Avrei potuto estrapolare come funziona la matrice dei pulsanti che



Figura 3: Cattura del segnale di start-stop sul telecomando

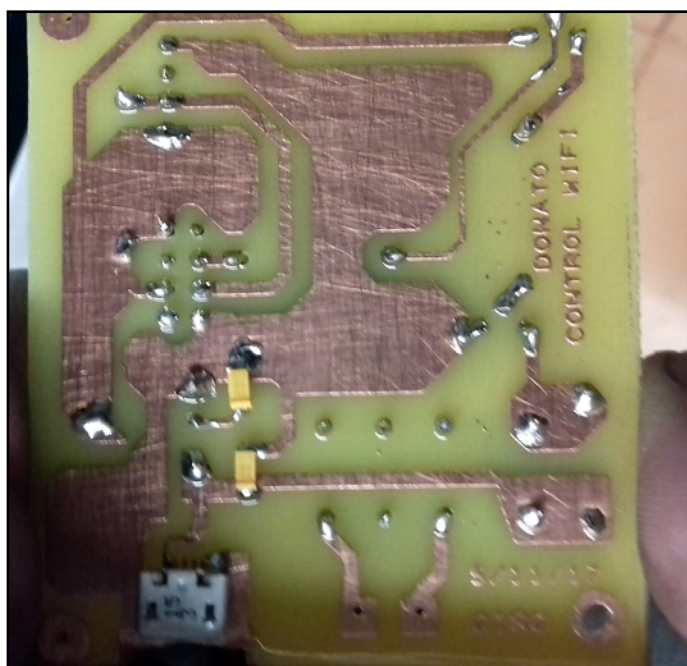


Figura 4: L'interfaccia completa, lato saldature

dialoga con il microcontrollore a bordo del telecomando e usare transistor PNP o NPN a seconda del caso, ma avrei reso la vita difficile ad alcuni tra i lettori che intendranno replicare il progetto. Invece, con il semplice contatto dello scambio NA (normalmente aperto) del relè, sono andato a collegarmi fisicamente sullo stampato del telecomando, in parallelo al pulsante di accensione. Vi chiederete perché due transistor, un PNP che comanda un NPN per poi comandare il relè?

Per il solito discorso sopracitato, utilizzando un transistor NPN ESP8266 collegato tramite la resistenza di base al GPIO il modulo ESP8266 non mi funzionava correttamente. Ho pensato quindi che ritornasse un valore logico Basso che mi mandava il modulo in programmazione.

Utilizzando questa configurazione, invece, funziona perfettamente.

CONSIGLI PER IL MONTAGGIO

Per prima cosa occorre operare sul telecomando, prestando molta attenzione a non danneggiarlo. Se le piste disegnate del “contatto” del pulsante ON/OFF del telecomando sono in grafite come nel mio caso attenzione, non ci potete saldare direttamente, anzi rischiate di danneggiarle irreparabilmente. Il consiglio che vi propongo è di individuare le piste coperte dal solder mask in un posto dove potete saldare agilmente.

Una volta individuate, per rimuovere l'isolamento praticate una leggera abrasione, anche con l'uso di un piccolo cacciavite a taglio, finché il rame sottostante non sarà ben visibile.

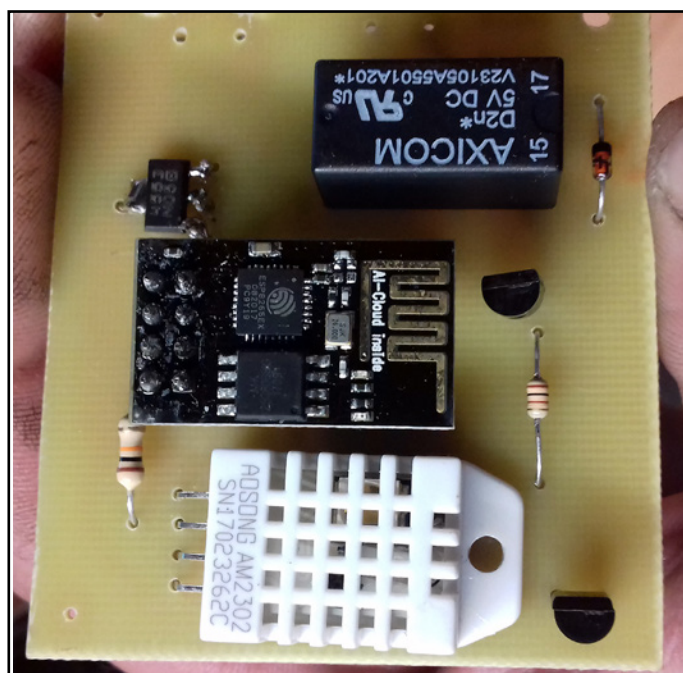


Figura 5: Il modulo Wi-Fi alloggiato sulla scheda



A quel punto potrete saldarci sopra.

In Figura 3 una foto del particolare.

Ricordatevi di saldare le due capacità da 10 μ F sul lato piste, prestando attenzione alla giusta polarità, dato che sono elettrolitici.

Anche la presa USB è a saldatura superficiale quindi, per evitare un circuito a doppia faccia, occorre saldarla sotto. Per quanto riguarda il regolatore Lm1117 in package SOT23 (e non in TO220), mi sono dovuto arrangiare per saldarlo sopra per permettere una migliore dissipazione.

Se lo acquisterete in contenitore TO-220 non avrete problemi, tanto il pin-out è il medesimo.

DESCRIZIONE DEL SORGENTE

Il sorgente da caricare nel modulo ESP8266 è stato compilato con l'IDE di Arduino. Dato che preparare ESP8266 e l'IDE di Arduino non è una cosa spiegabile in 5 minuti, per non dilungarmi troppo in questo articolo, cercate una guida sul web a vostro piacere o leggete una mia guida al riguardo sul sito di E&M.

Il sorgente è veramente semplice, grazie all'uso delle due librerie: Una per il sensore DHT22 di temperatura e umidità e l'altra per il modulo ESP8266. Eseguita alla lettera le istruzioni della guida, ricordatevi di cambiare il tipo di scheda sull'IDE scegliendo, appunto, Generic ESP8266 Module, altrimenti il compilatore non compilerà neanche il semplice sorgente che

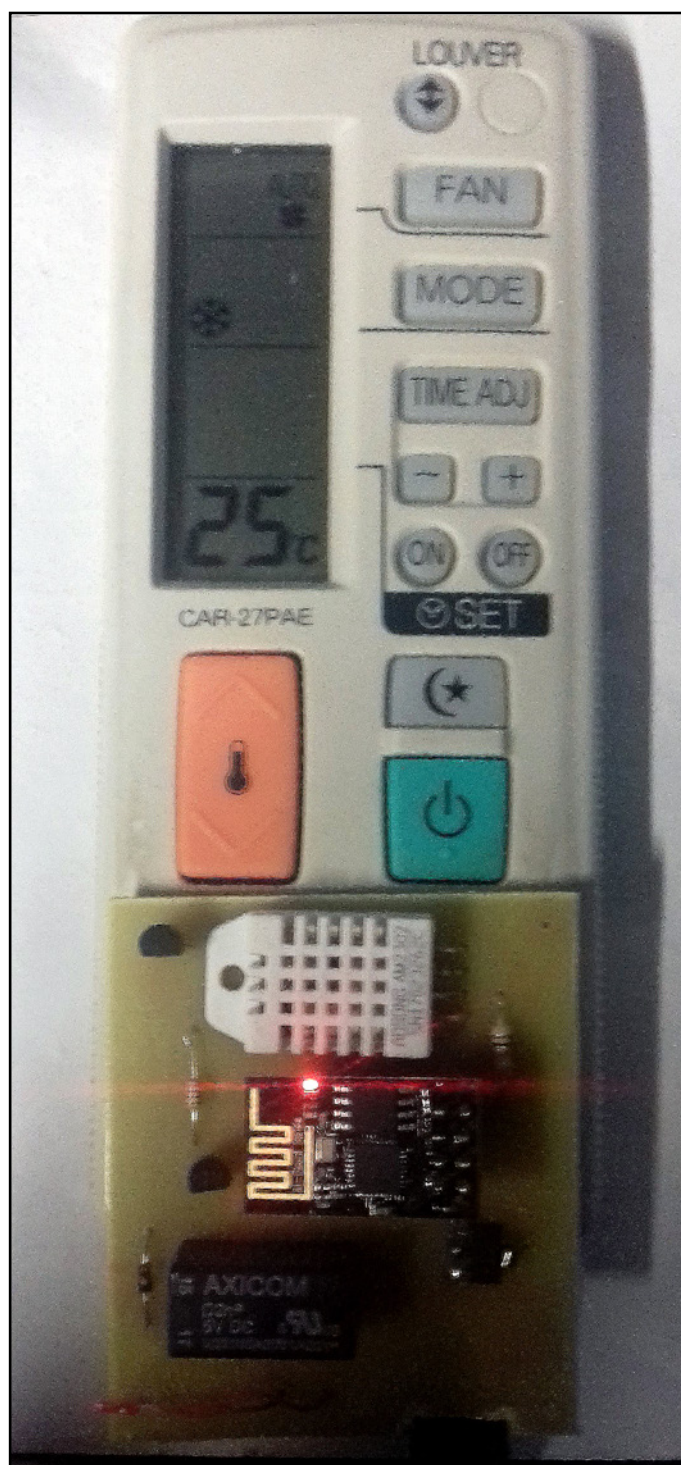


Figura 6: Il dispositivo montato sul telecomando

segue, né tantomeno caricarlo. Soprattutto perché i fuse ecc. sono totalmente diversi dalla comune scheda di Arduino UNO.

```

#include <ESP8266WiFi.h>
#include <DHT.h>
#define DHTTYPE DHT22
#define DHTPIN 2

DHT dht(DHTPIN, DHTTYPE, 30);

const char* ssid = "---XXXXXXXXXX---"; //inserire qui la user della vostra wifi
const char* password = "---XXXXXXXXXX---"; //inserire qui la user della vostra wifi

int ledPin = 0; // GPIO 0//pin che pilota i transistor---->relè
WiFiServer server(80);
int stato;
int tempmax=21;
int tempmin=18;
int value = LOW;

void setup() {
  // Serial.begin(115200);
  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, HIGH);
  delay(10);
  dht.begin();
  delay(10);

  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
  }

  server.begin();
}

void loop() {

  WiFiClient client = server.available();
  if (!client) {
    return;
  }

  Serial.println("new client");
  while(!client.available()){
    delay(1);
  }

  String request = client.readStringUntil('\r');
  Serial.println(request);
  client.flush();
  delay(4000);

  float h = dht.readHumidity();

  float t = dht.readTemperature();

```



```
int value = LOW;
if (request.indexOf("/LED=ON") != -1) {
    digitalWrite(ledPin, LOW);
    delay(700);
    digitalWrite(ledPin, HIGH);
}

client.println("HTTP/1.1 200 OK");
client.println("Content-Type: text/html");
client.println("");
client.println("<!DOCTYPE HTML>");
client.println("<html>");//inizio a scrivere pagina html
client.println("<head>");//
client.println("<title>GESTIONE CONDIZIONATORE by GIRO</title>");//
client.println("<head>");//
client.println("<body>");//lo sfondo
client.println("<body bgcolor='\"\"#CCFFFF\"\">");//colore sfondo
client.println("<center>");
client.println("<font size='\"\"+4\"\">"); client.println(""); // do not forget this one
client.println(""); // do not forget this one
client.println(""); // do not forget this one

client.println("<br><br>");
client.println("<font color='\"\"#000000\"\">");
client.println("Cliccare <a href='\"\"/LED=ON\"\"> ON </a> per ac

cendere condizionatore <br>");

client.println("<center>");

if (isnan(h) || isnan(t) ) {
    client.println("<center>");
    client.println("ERRORE sensore,<a href='\"\"/^\"\"> riaggiornare la pagina </a>");
    return;
    client.println("<center>");
}

client.println("<center>");
client.print("Temperatura: ");
client.print(t);
client.println(" C ");
client.println("<center>");
client.println(""); // do not forget this one

client.print("Umidita: ");
client.print(h);
client.println(" %\t");
client.println("<center>");
client.println("...<a href='\"\"/^\"\">aggiorna DATI </a>...");
client.println("</font>");
client.println("<center>");
client.println("<body>");
client.println("</html>");
delay(100);
}
```

Come avrete notato dal sorgente, una volta richiamato l'ESP8266 digitando l'indirizzo IP assegnato dal Router, il modulo risponde scrivendo una pagina HTML, proprio come se noi la scrivessimo in linguaggio HTML. Tale pagina darà la possibilità di cliccare su ON per accendere il condizionatore e vi comunicherà, in tempo reale, la temperatura e l'umidità misurate dal sensore DHT22. Inoltre, cliccando su aggiorna DATI la pagina subisce un "refresh", aggiornandosi con le nuove misure di temperatura e umidità. Non ho aggiunto nessun controllo di stato del dispositivo in quanto, collegandomi solo in parallelo al pulsante, non posso sapere se il condizionatore è acceso. Se il telecomando viene utilizzato manualmente e supponiamo che sia stato acceso, con questi due collegamenti non posso sapere in che stato si può trovare il condizionatore. Anche "memorizzando" internamente tramite software i cambi di stato

eseguiti da ESP8266, non posso capire se qualcuno ha premuto manualmente il tasto di accensione o spegnimento.

Qui viene però in soccorso il sensore di temperatura e umidità. È piuttosto facile da intuire, poiché se vengono rilevate temperature molto vicine a quelle impostate il condizionatore è acceso, in caso contrario sarà sicuramente spento.

Dato che molti utenti non hanno un indirizzo statico, ma soltanto dinamico, come possiamo richiamare il nostro ESP8266 quando siamo fuori senza grosse difficoltà?

Il mio amico Geometra sopracitato tiene sempre un PC acceso e usa Team Viewer come applicazione per il controllo remoto. E' un applicazione gratuita, se non viene usata per scopi professionali, mentre il mio amico, essendo un professionista, l'ha acquistata.

E' installabile anche su Smartphone, è facile ed intuibile da utilizzare, non avrebbe problemi ad usarla.



Cliccare ON per accendere condizionatore
Temperatura: 13.10 C
Umidità: 88.10 %
...aggiorna DATI...

Figura 7: Pagina HTML di comunicazione dello stato ambientale

In pratica questa applicazione fa da "ponte" tra il vostro Smartphone e il vostro PC. Tale Software vi permette di gestire, con un controllo totale, il vostro PC e di trasferire eventuali file.

Il Programma vi for-



nisce un ID da richiamare e potete scegliere se utilizzare una password rolling-code o fissa a vostra scelta. La rolling-code è la più sicura, ma più scomoda, dato che ad ogni avvio del programma cambia valore, con il rischio di dimenticarla. Inserendo il programma in esecuzione automatica, per non dimenticarsi di aprirlo tutte le volte, basta un breve blackout di corrente elettrica e il PC si riavvia, ma la password rolling-code sarà diversa e non potrete più accedere. Il mio consiglio è di usare una password fissa a vostra scelta, ma la decisione finale è soltanto vostra. Ecco dove potete scaricare Team Viewer <https://www.teamviewer.com>. Una volta entrati nel PC basta semplicemente aprire un qualsiasi browser per la navigazione in Internet e digitare nella barra degli indirizzi l'indirizzo IP che ha assegnato il router al modulo ESP8266.

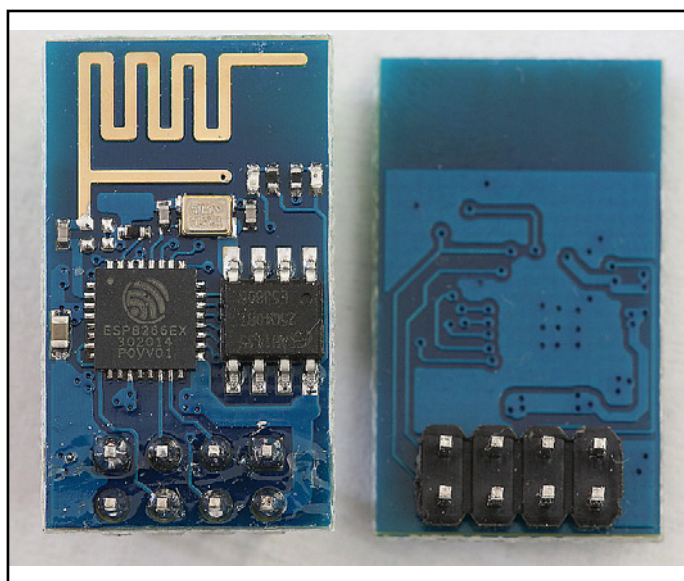


Figura 8: Il modulo ESP8266

Se siete collegati, invece con lo Smartphone alla rete Wifi in cui è connesso Esp8266 utilizzate il browser sul vostro cellulare direttamente senza passare da Team Viewer.

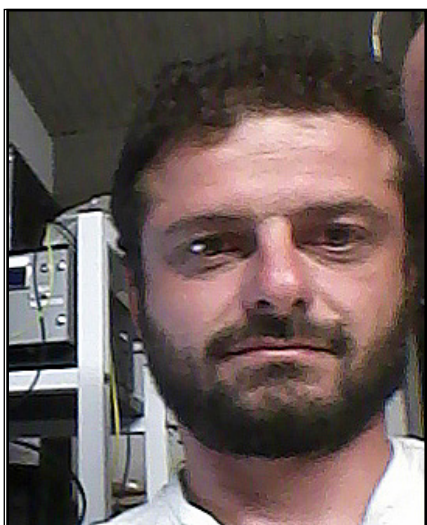
Qui sotto troverete lo Sketch da caricare su Esp8266, lo schema disegnato su Proteus 8.0 e il pcb disegnato su DesignSpark 7.2

Buon divertimento e buona realizzazione a tutti.



Elenco componenti

R1	10 KΩ 1/4 W
R2÷R3	1K Ω 1/4 W
C1÷ C2	10 μF 16 V elettrolitico
D1	1N4148
Q1	BC337 NPN
Q2	BC327 PNP
IC1	Lm1117
Modulo Wi-Fi	ESP8266-01
DTH22	
Presse Micro USB tipo B	



Mi chiamo Girolamo D'Orto, 36 anni sposato, con una figlia e vivo nella provincia di Siena.

Mi sono diplomato come perito elettrotecnico all'ITIS. Non ho potuto continuare gli studi per andare a lavorare nella azienda di famiglia.

Sono un produttore di vino Chianti Classico. L'elettronica è stata sempre la mia passione fin da giovane. A 11 anni ho iniziato a stampare i pcb con i trasferibili, e replicavo gli articoli sulle riviste. Ho abbandonato dopo un paio di anni questo hobby per poi riprenderlo in mano nel 2013. Sono ripartito da zero come autodidatta, leggendo studiando e imparando qualcosa. Mi sono appassionato in seguito dei microcontrollori pic compilando il mikrobasic, per poi passare a Arduino. Normalmente realizzo pcb che mi fanno comodo a casa e in azienda. Amo farli da me, piano piano mi sono avvicinato agli smd che con l'attrezzatura hobbistica spesso sono ostici, ma ho comunque ottenuto risultati soddisfacenti. Stampare pcb, fare lo sbroglio che è la cosa che amo di più, è il modo per rilassarmi la sera quando torno da lavoro.



Alimentatore "CUSTOM"

di Roberto Vallini

r.vallini@elettronicaemaker.it

“Nel mezo del cammin del mio laboratorio...mi ritrovai con una marea di prototipi”. Così vorrei iniziare questo mio primo articolo, basato sulla considerazione che essere un “Maker”, porta dei vantaggi e delle soddisfazioni a livello personale, ma, almeno nel mio caso, ci si trova con tanti prototipi, tutti funzionanti, ma da concludere, ovvero, mancano di una veste fruibile per i terzi, i quali, auspicabilmente andranno ad interessarsi di qualcuno di questi. Per cui inizierò con un progetto utile, se non indispensabile nel proprio laboratorio, un alimentatore da banco custom, dedicato ad Arduino.

Tutti quei prototipi che vedete nella fotografia a destra, sono il frutto di tanto lavoro e tanta passione, si vedono basette con integrati, sensori PIR, azionamenti, caricabatterie, laser, vari tipi di Arduino e altro ancora.

Tutti i progetti sono funzionanti, (per ovvi motivi ho dovuto scollegare e spostare parte delle connessioni) per fare stare tutto nella foto, la quale, nota bene, rappresenta solo una parte del “quantum” del mio lavoro.

Per ognuno di essi è stato necessario, predisporre una basetta, una breadboard, una MCU, un’alimentazione, ma soprattutto tante ore al PC a caccia della libreria giusta. Ma il grosso problema è stato dare a tutti la giusta alimentazione.

COSA MI SERVE

Sarebbe meglio titolare :“Cosa serve ad Arduino” ed al suo mondo. Dunque partiamo dalla alimentazione 5 Volt con presa USB. Il mio alimentatore ne fornisce ben 6, DUE protette da fusibile sulla parte fissa del dispositivo e QUATTRO su quella che ho battezzato “Penisola 2”, di cui parlerò più avanti. Proseguiamo la panoramica. Sempre sull'alimentatore vi è una tensione DUALE, per dispositivi vari che montano magari operazionali senza sdop-

piamento “on-board” (per esempio un generatore di segnali che ho in laboratorio), ed una uscita variabile, (2,5-24 volt) la quale non supera i 500 mA, questa per gingilli vari, come piccoli motori, o bobine di relè ignoti, di fatto una uscita di testaggio.

LA PENISOLA

Molte volte, mi sono trovato di fronte, all'esigenza di alimentare un progetto, con sensori, display, Arduino shield & board, non tutti vicini all'alimentatore.



Figura 1: Una moltitudine di progetti e prototipi sviluppati di recente



Ebbene, l'unica possibilità era quella di tirare cavi e cavetti, con diverse lunghezze, e con diversi tipi di pin, rischiando di confondere le polarità.

Con questa mia idea, penso di aver scoperto “L'uovo di Colombo”, ovvero remotare le varie tensioni su una “Penisola”, attraverso un cavo di 20 centimetri. La penisola può stare a bordo, oppure, essere spostata, per agevolare le connessioni.

Su di essa vi sono alloggiati, per ogni tensione, un pin maschio, uno femmina ed una morsettiera.

Sono presenti le tensioni 3,3 V, 5V e 12 Volt. Dalla penisola 1 se ne diparte una seconda, con un USB driver port a QUATTRO uscite..più di così.

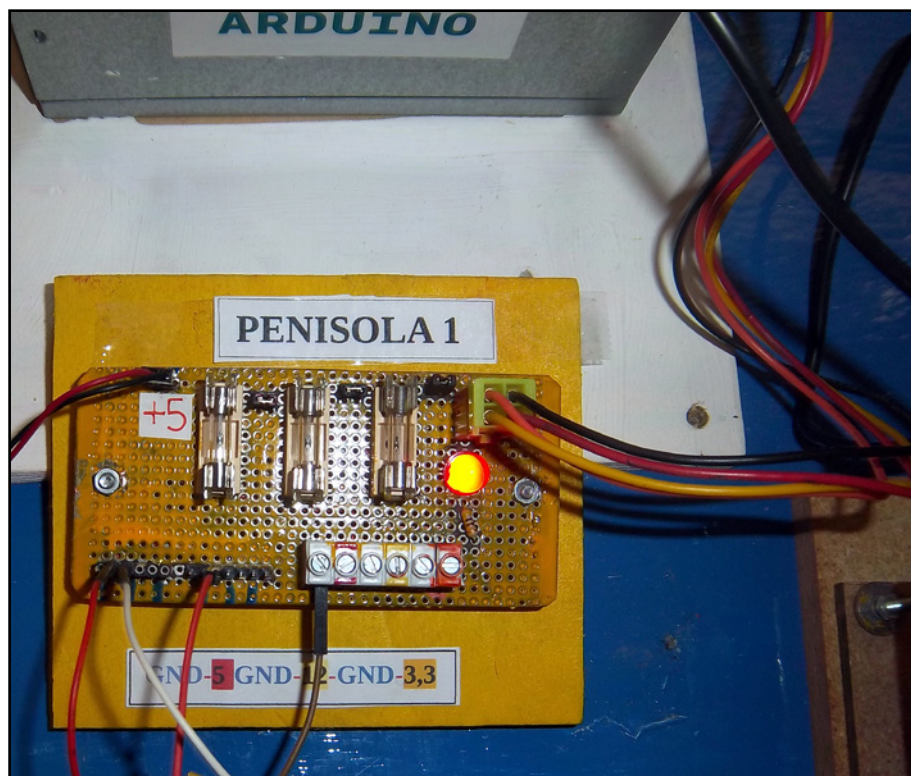


Figura 1: Foto della penisola

ATX, CHE MINIERA!

L'alimentatore ATX che tutti conosciamo, è stata la “miniera e la sorgente da cui prelevare le tensioni, ne ho utilizzati due, uno funzionante e uno danneggiato inesorabilmente che ho svuotato per utilizzarne il case.

Da quello funzionante, attraverso un'opera di eliminazione del superfluo connettore a 24 POLI, e dissaldando molti conduttori interni, ho usato solo il cavo NERO di GND, quello ARANCIONE, per i 3,3 volt, il GIALLO, per i 12 Volt, ed il ROSSO per i 5 Volt. Il cavo VERDE, che viene usato per lo start è stato portato posteriormente, e tramite un interruttore switch, attiva l'intero sistema. Per quanto riguarda

la tensione duale, ho sfruttato un cavo VIOLA (-12 V) ed il GIALLO (+12 Volt) e tramite morsettiera laterale è stato portato al case vuoto dove risiede una presa maschio, con femmina estraibile.

Anche la tensione 220V, necessaria al funzionamento dell'alimentatore variabile interno è stata presa direttamente dall'attacco shuko e portata al case superiore, sempre tramite morsettiera laterale.

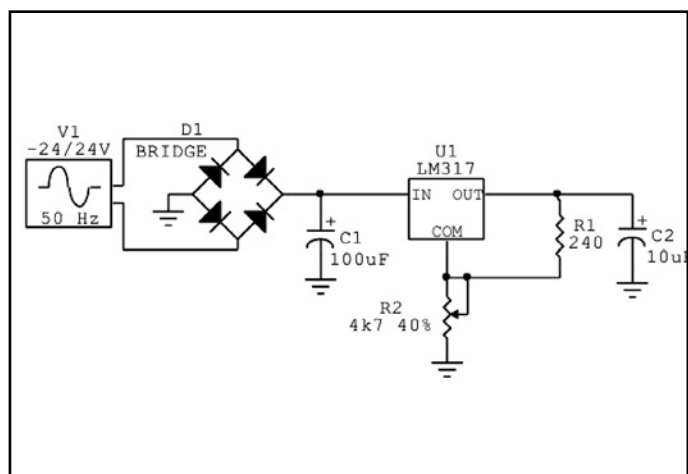


Figura 2: Il circuito variabile

IL CONTENITORE DELL'ALIMENTATORE

Come già detto in precedenza le tensioni necessarie ad Arduino le ho prelevate dall'atx inferiore, per quanto riguarda le altre due (+ -12 e Variabile) ho sfruttato un case vuoto, fissandolo a quello inferiore con dei distanziali cilindrici. Mi serviva quindi, lo spazio per il circuito variabile, creato su una millefori, poi, uno spazio per il connettore duale, il quale si distinguesse bene dal resto, una spia di accensione, l'uscita del potenziometro "var" con relativa manopola e le due USB.

Non è stato semplice far stare tutto, ma con un po' di esperienza ho iniziato l'opera di predisposizione.

DUE PAROLE SUL CIRCUITO VARIABILE

L'alimentatore variabile è basato sul conosciutissimo circuito integrato LM317, che provvede a stabilizzare la

corrente ed a regolare la tensione mediante il potenziometro R2. La tensione in uscita può essere calcolata per mezzo della formula

$$R_2 = R_1 \left(\frac{V_{out}}{1.25} - 1 \right)$$

Ad esempio se si vogliono in uscita circa 24 Volt

$24V/1,25 = 19,2$ $19,2-1 = 18,1$ $240 \Omega \times 18,1 = 4.344 \Omega$ (4,7 K Ω valore commerciale).

Il mio circuito l'ho realizzato interamente su millefori, con un piccolo trasformatore da 10 VA

Quella che segue è una piccola foto-galleria che illustra con pochi commenti didascalici le varie fasi della realizzazione,

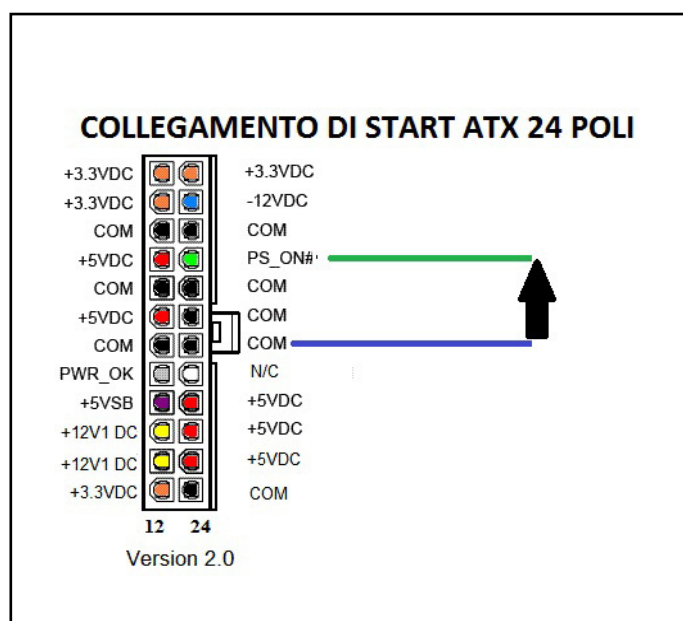
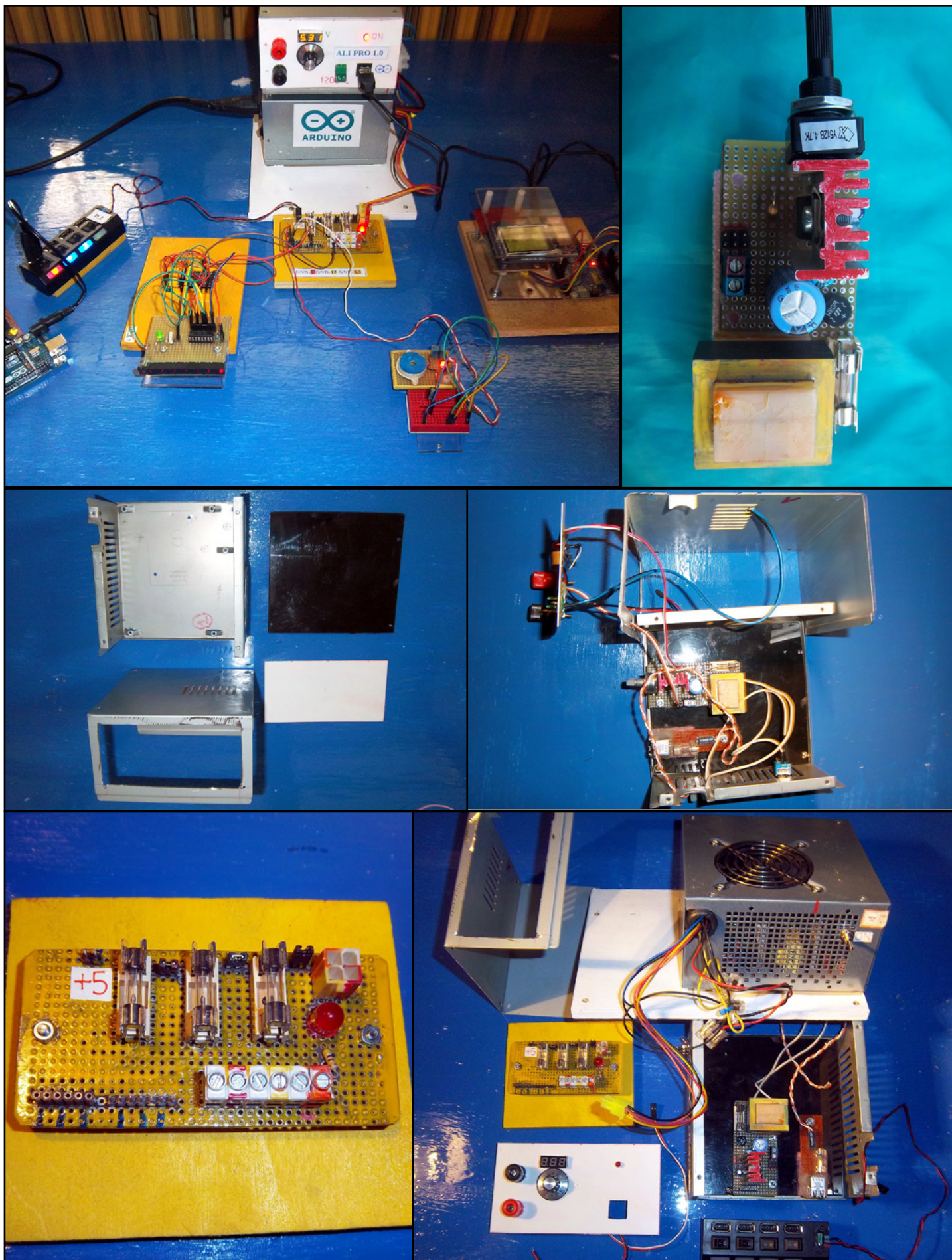


Figura 3: Collegamenti di avvio





Sono nato nel 1968, abito a due passi da Torino.
Dopo aver conseguito la maturità tecnica in elettronica, ho iniziato la mia carriera lavorativa, come cablatore di quadri, mentre la sera studiavo, per il grande interesse che mi sono infatti iscritto, successivamente, alla Facoltà di Lettere e Filosofia

di Torino, dove ho sostenuto gli esami di Filosofia teoretica, Filosofia Morale, Storia della filosofia antica, Antropologia culturale, Antropologia delle Religioni. Nel 1989, Master Regionale, presso la Federazione Nazionale Agenti e Rappresentanti di Commercio in Psicologia delle vendite, Amministrazione aziendale, Marketing aziendale. Ho lavorato per quasi vent'anni in multinazionale dell'elettronica di processo, occupandomi del settore del dialogo uomo - macchina. Partito come semplice promotore, ho raggiunto la qualifica di area manager. Successivamente ho fondato un'azienda personale, nel settore dell'elettromeccanica. Attualmente, mi occupo di informazione, come redattore presso un gruppo editoriale di cronaca. Di recente ho scoperto il mondo di Arduino, con il quale è stato "Amore a prima vista". Ho scritto articoli per GRIX e per la rivista Fare Elettronica. Sono "Amministratore Orgoglioso" della pagina "Elettronica & Maker", un progetto in cui credo fermamente. Spero di avere dato e di dare il mio contributo, con le mie qualità, affinché possiamo brillare nella galassia dell'editoria elettronica.



Rilevatore Ultrasonico del Livello dei Liquidi

di Angelo Lezza

a.lezza@elettronicaemaker.it

PRIMA PARTE

Un economico sistema di rilevazione elettronica del livello dei liquidi di un serbatoio, che nasce come soluzione ad un fastidioso problema personale

Nella città in cui vivo è purtroppo assente il servizio di “acqua corrente”, per cui è necessario attendere il proprio turno di erogazione idrica, che varia in funzione dei periodi stagionali e delle diverse zone della città.

Noi agrigentini, per far fronte a questo stato di cose, abbiamo nel tempo imparato a fare uso di cisterne di accumulo, in maniera tale da poter disporre del “prezioso liquido” anche nei giorni in cui non è prevista l’erogazione. Nonostante ciò, mi è successo diverse volte di rimanere senz’acqua!

Talvolta infatti, durante l’erogazione, detriti di varia natura quali pietrisco, terriccio, ecc., presenti non si sa bene come all’interno delle condotte pubbliche, si collocano giusto a ridosso dell’ingresso a gomito del mio contatore, provocando l’ostruzione delle tubazioni e quindi

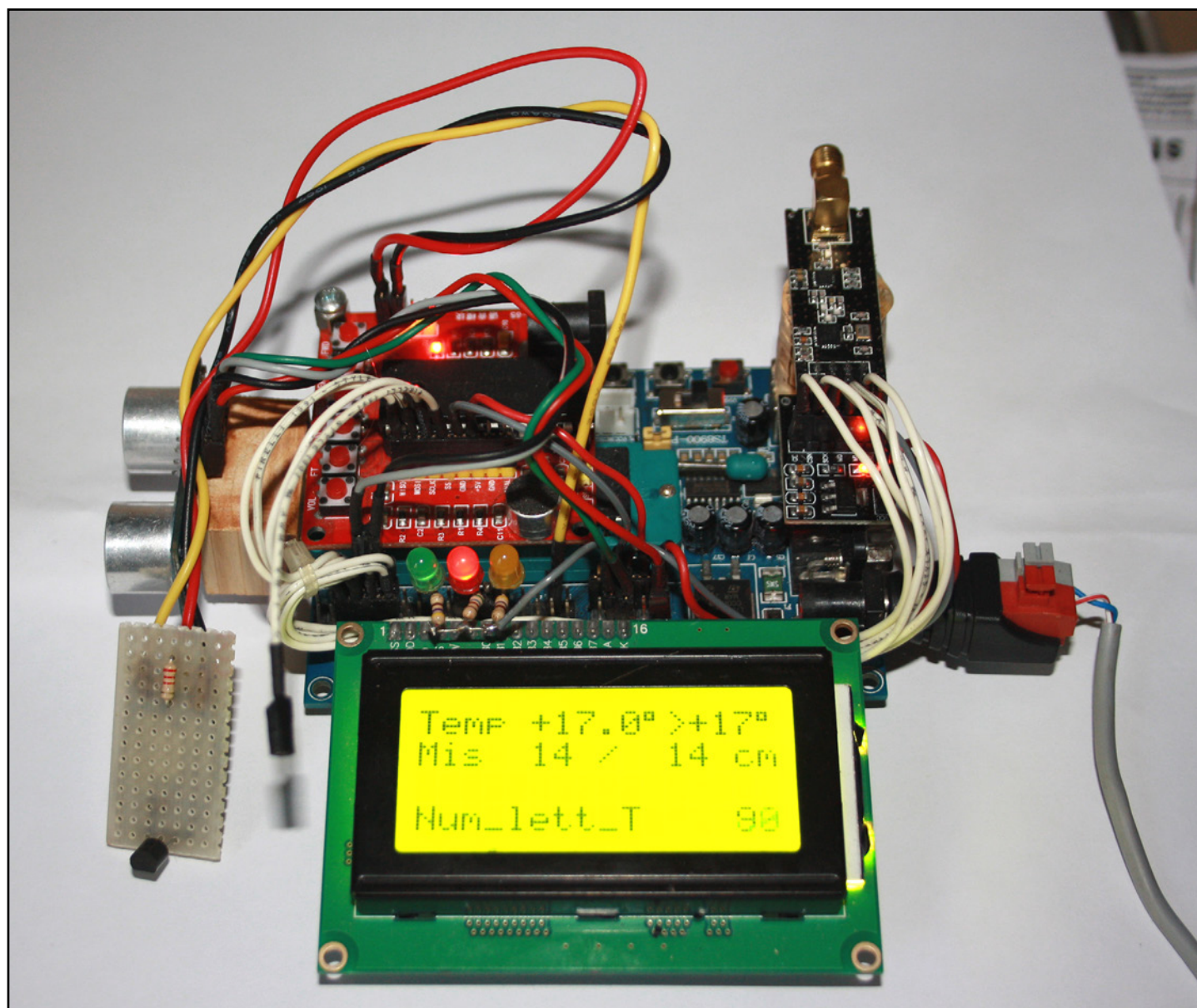
il mancato approvvigionamento del serbatoio.

Purtroppo, nel momento in cui si presenta l'ostruzione, non si ha alcuna manifestazione evidente del guasto, difatti si percepisce il problema solo qualche giorno dopo quando, a causa dell'esaurimento delle scorte idriche della cisterna, si rimane inesorabilmente a secco e senza alcun preavviso. Un disagio fami-

liare enorme causato da un fenomeno silente e 'non prevedibile'!

Qualche mese fa, al ripetersi dello spiacevole inconveniente, mia moglie, ormai più stanca di me, mi ha invitato ad adottare dei controlli preventivi e periodici del recipiente.

Cioè scendere in giardino ogni due o tre giorni, aprire la botola e verificare il livello del liquido. Comprimerete che que-



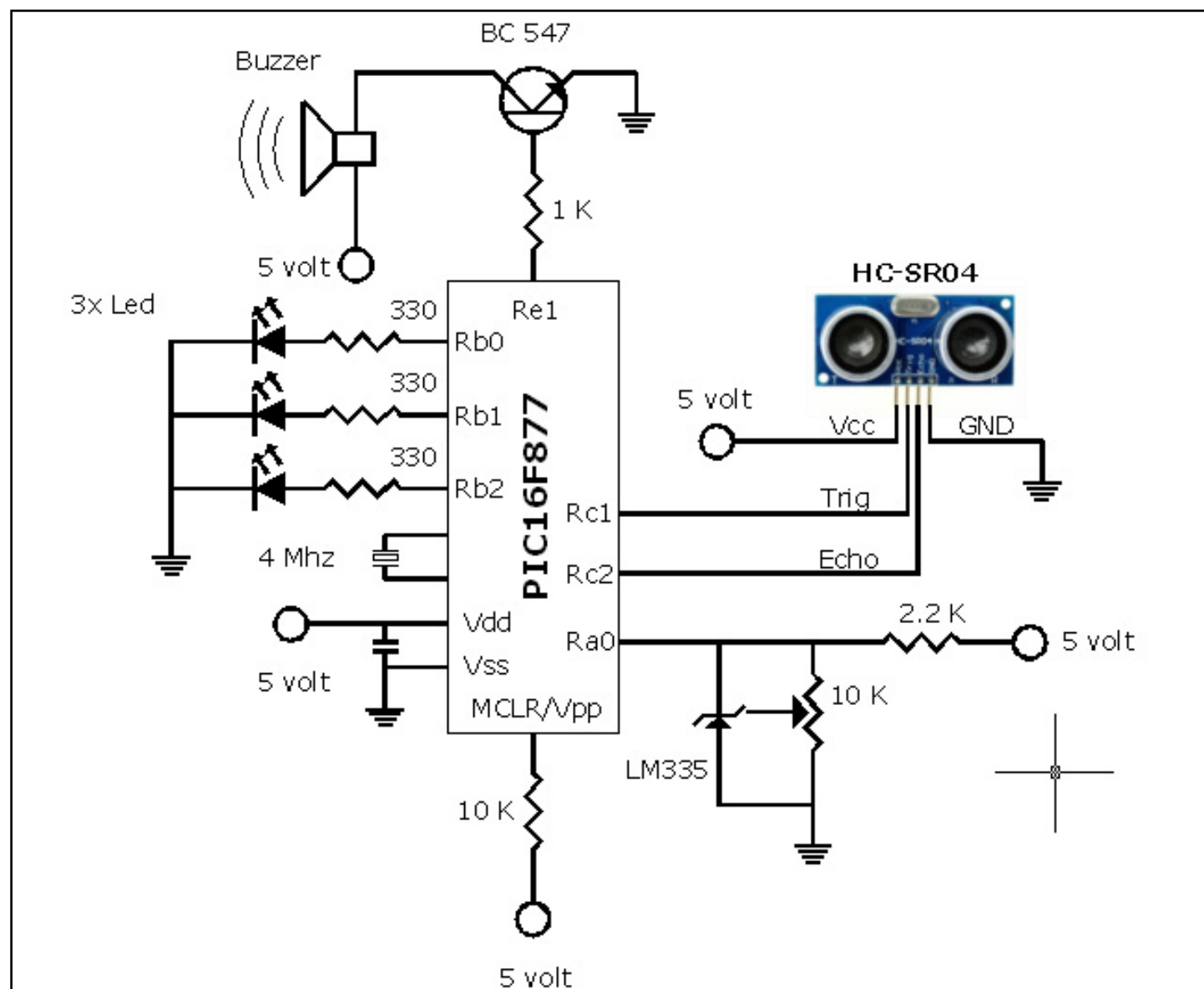
Board con PIC16F877 utilizzata soltanto durante le prove iniziali



sto 'modus operandi' con cui affrontare in maniera pratica il problema, seppur formalmente corretto almeno dal punto di vista suo e della prevenzione, risulta fastidioso per me che lo deve poi mettere in atto. Un controllo ogni tre giorni equivale a circa 118 controlli/anno, una seccatura di enorme entità! Nel mio caso, poi, è pure difficilmente praticabile perché il serbatoio è collocato in una zona impervia.

E allora che fare???

Ho riflettuto non poco sulla questione e sono giunto alla conclusione di provare a costruire un rilevatore elettronico di livello liquidi. Un semplice circuito, dalle caratteristiche economiche e di affidabilità, che mi consenta di rilevare la quantità di acqua della cisterna e mi segnali in tempo utile eventuali emergenze.



Schema elettrico del circuito elettronico e dei sensori (schema finale)

DESCRIZIONE DEL CIRCUITO

Il sistema è sostanzialmente composto dai sensori HC-SR04 ed LM335 e dalla centralina di elaborazione e rilevazione. I sensori li ho collocati all'interno della cisterna, nella parte superiore non accessibile all'acqua, mentre il circuito della centralina è posto esternamente; entrambi installati dentro appositi contenitori a tenuta stagna.

Il circuito elettronico, alimentato con una tensione stabile a 5 volt, elabora i dati, segnala il livello dei liquidi e, in caso di criticità, attiva un dispositivo di allarme (buzzer). Il compito più importante è quindi assegnato alla scheda elettronica con a bordo un più che performante Pic16F877 il quale acquisisce i dati ricevuti dal modulo HC-SR04 e dal sensore di temperatura LM335, effettua i calcoli e giunge alla rilevazione del livello liquidi; ovviamente, tutti i calcoli tengono nella giusta considerazione la geometria del serbatoio e i livelli minimo e massimo da adottare (troppo vuoto e troppo pieno).

IL SENSORE AD ULTRASUONI HC-SR04

Il 'lavoro sporco', se è lecito definirlo, è conferito ad un interessantissimo modulo HC-SR04, composto da una scheda con tre integrati ed altri componenti. Il modulo presenta quattro pin di connessione, e precisamente: alimentazione 5v, Gnd, TRIG ed ECHO.

Il funzionamento è tutto sommato semplice ed intuitivo.

Il modulo invia ultrasuoni (frequenza di 40 KHz circa) i quali, rimbalzando contro



Sensore HC-SR04

l'ostacolo posto frontalmente (nel nostro caso l'acqua), vengono riflessi indietro, per essere nuovamente ricevuti dal sensore HC-SR04 che elabora le informazioni e restituisce, sul pin d'uscita "echo", un impulso di durata pari al tempo intercorso per compiere la tratta di andata e ritorno. Quest'ultimo dato, come vedremo, è necessario al microcontrollore per calcolare la distanza.

Per attivare il sensore HC-SR04 ho eseguito la seguente procedura:

1. Invio un impulso non inferiore a 10 us sul pin "TRIG" del sensore (l'eventuale ripetizione dell'impulso deve avvenire ad intervalli di tempo non inferiori a 60 us);
2. Il sensore invia 8 impulsi sonici;
3. Misuro l'impulso ricevuto dal pin "ECHO" (tempo intercorso tratta



andata - ritorno).

Per le misurazione da pin "echo" ho deciso di utilizzare il modulo CCP del microcontrollore, in modo "Capture". Configuro il prescaler del Timer 1, il registro CCPxCON<3:0> (CCP1M3:C-CP1M0), i registri dell'interrupt, ecc.

Appena ricevo l'interrupt, sottraggo il valore salvato all'istante t1 (fronte di salita) dal valore catturato all'istante t2 (fronte di discesa).

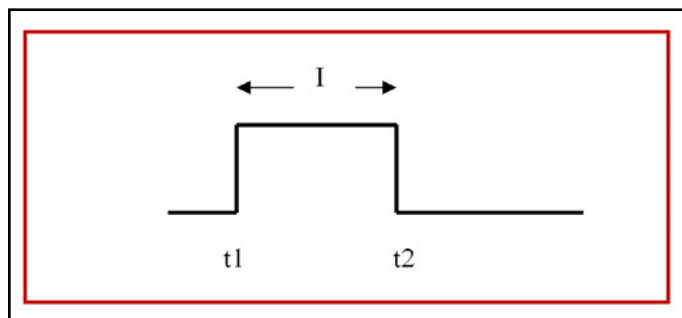
Tale sottrazione mi restituisce la larghezza dell'impulso I (in microsecondi). Per il calcolo della distanza (D) adotto la formula base:

$$D = \text{tempo} * V/2 \quad (\text{Formula n. 1})$$

Dove:

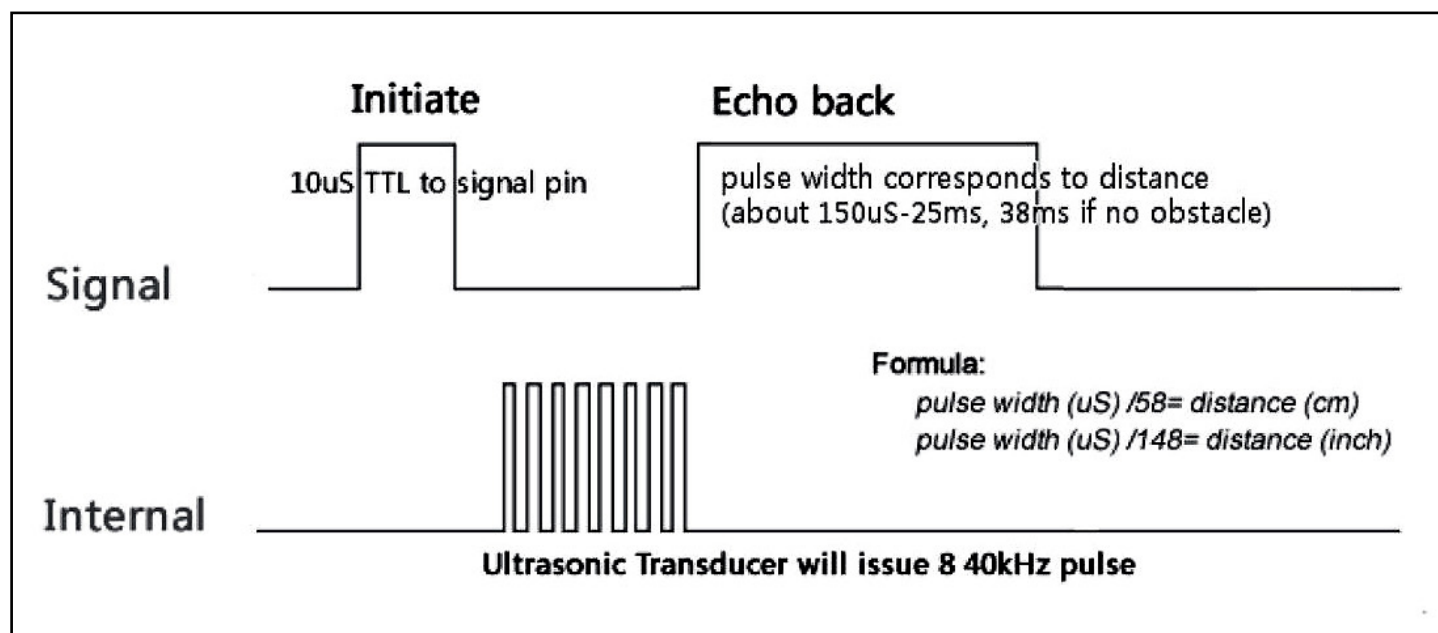
D = Distanza tra il sensore e l'ostacolo;
tempo = misura dell'impulso ricevuto dal pin "ECO" (percorso andata e ritorno);

V = Velocità di propagazione delle onde sonore.



Note: E' a questo punto opportuno ricordare che la velocità di propagazione di un'onda sonora dipende soprattutto dal mezzo che deve attraversare e dalla temperatura. La relazione matematica che lega questi due fattori può così sintetizzarsi: la velocità delle onde sonore aumenta con l'aumentare della temperatura e della densità del mezzo da attraversare. Se il mezzo da attraversare è l'aria avremo, ad esempio, una velocità pari a circa m/s 331 a 0°C e circa m/s 344 a 20°C.

Quindi, per calcolare la velocità delle onde sonore (V), utilizzo la seguente for-



mula:

$$V = 331,4 + (0,62 \times T) \quad (\text{Formula n. 2})$$

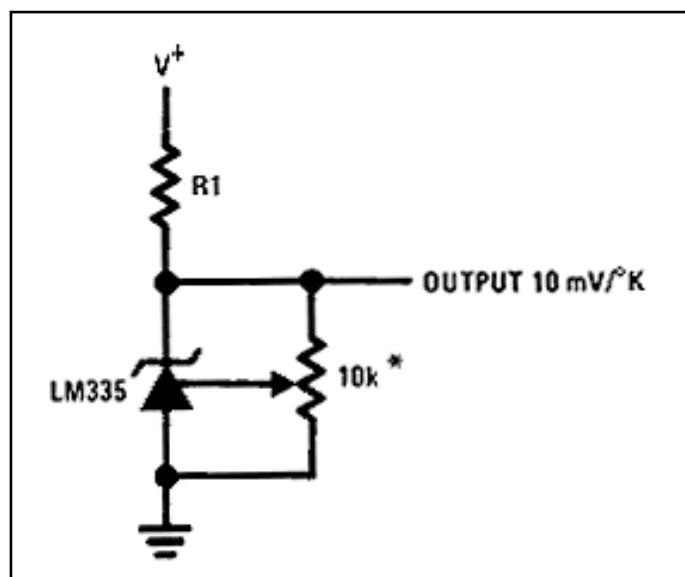
Dove:

V = Velocità di propagazione delle onde sonore, T = Temperatura.

IL SENSORE LM335

Per rilevare fisicamente la temperatura impiego un semplice LM335 della Texas Instruments, che è un sensore di tipo analogico, di costo basso, con uscita lineare e che lavora nell'intervallo da -40°C a 100 °C. Ove opportunamente calibrato, presenta un'accuratezza di circa 1° in un intervallo di 100° C, cioè un errore più che accettabile per un'applicazione del genere. Il valore analogico viene acquisito tramite il canale AN0 (porta RA0) del Pic16f877, nell'istante 't', grazie al convertitore analogico/digitale a 10 bit dello stesso micro.

Per la conversione ho scelto la tensione



Schema LM335 e circuito calibrazione

di riferimento interna al microcontrollore e compresa tra Vss (0v) per -VREF e VDD (5v) per +VREF.

I registri del Pic necessari alla conversione sono ADCON0, ADRESH (banco 0) e ADCON1, ADRESL (banco 1). ADC00 e ADCON1 servono per l'impostazione dei convertitori mentre ADRESH e ADRESL restituiscono il risultato dell'operazione adc (10 bit).

Per rilevare la temperatura, il micro dovrà compiere una serie di operazioni matematiche. Innanzitutto occorre tener presente che l'uscita lineare del sensore LM335 fa espresso riferimento alla temperatura riferita in gradi ° Kelvin.

Per cui (Formula n. 3):

$$^{\circ} \text{Kelvin} = \frac{\text{ADRESH} / \text{ADRESL} \text{ (10 bit)} * 500}{1023}$$

E siccome la relazione che lega i ° Kelvin ai ° Celsius è la seguente: ° Kelvin - 273,15 = ° Celsius, avremo quindi:

$$^{\circ} \text{C} = ^{\circ} \text{K} - 273,15 \quad (\text{Formula n. 4})$$

Durante le fasi preliminari del progetto, mi sono però imbattuto in alcuni problemi inizialmente insormontabili ma successivamente risolti. La lettura della temperatura risultava talmente instabile e ben oltre la tolleranza del costruttore che influiva negativamente anche sulla lettura della distanza, probabilmente a causa dei seguenti fattori:

- Errore del sensore LM 335 (mancata



calibrazione)

- Rumore in fase di conversione dovuto al modulo A / D del PIC16F877

- Errore di quantizzazione della conversione A / D: è un errore sempre presente anche nel caso in cui utilizzassimo un convertitore A/D ideale. Un segnale analogico convertito nell'istante 't' presenta un errore casuale, indipendente dall'ingresso (rumore di quantizzazione). Rappresenta una variabile imprevedibile dovuta alla risoluzione finita dell'ADC ed è una imperfezione intrinseca di tutti i tipi di ADC.

Per ridurre tali inconvenienti ho dovuto mettere in atto le seguenti strategie. Ho innanzitutto provveduto a calibrare il sensore LM335 alla temperatura di 25° ottenendo un'accuratezza di circa 1°C.

Mi sono avvalso della possibilità di far



L'HC-SR04 all'interno del contenitore

funzionare il PIC16F877 in modalità SLEEP, con sorgente dell'orologio in modalità RC. Con queste impostazioni, il modulo A / D attende un ciclo di istruzioni prima di avviare la conversione consentendomi, grazie anche all'interrupt, di far diminuire sensibilmente il "fattore rumore".

Ho scelto inoltre di acquisire 100 campioni di temperatura (letture adc) ad intervalli costanti, effettuando poi una media ponderale della somma delle stesse letture. L'intervallo T, tra due istanti di campionamento (periodo di campionamento), l'ho fissato pari 1 ms, ottenendo così dei valori veramente accettabili.

OSSERVAZIONI

Altro aspetto che può causare problemi nella conversione analogico digitale è la variazione anche lieve dell'alimentazione del circuito, che è anche la tensione di riferimento del modulo adc. Se, ad esempio, forniamo al circuito 4,9 volt al posto dei 5 volt richiesti, è necessario sostituire nella formula 3 il numero 500 col numero 490:

RIEPILOGANDO

Per rilevare la capienza della mia cisterna provvedo a far eseguire al microcontrollore delle misurazioni ogni 15 minuti (circa). Dapprima rilevo la temperatura in °Kelvin (vedi formula

n. 3), attraverso 100 campionamenti per ogni misurazione della distanza. La somma dei 100 campionamenti la divido per 100, conseguendo una media ponderale della temperatura in ° Kelvin. Alla temperatura così acquisita (°Kelvin) sottraggo il valore 273,15 ed ottengo la temperatura in ° Celsius (vedi formula n. 4). Calcolo quindi la Velocità di propagazione delle onde sonore tramite la formula n. 2 e successivamente, con l'ausilio del modulo CAPTURE del PIC16F877 in gestione interrupt, ottengo il valore dell'impulso, in termini di tempo, relativo al tragitto di andata e ritorno che gli ultrasuoni compiono (tra modulo - ostacolo e viceversa).

Grazie ai dati acquisiti e alle operazioni matematiche di cui alla formula n. 1, calcolo la distanza tra il sensore e il pelo dell'acqua. Conoscendo a priori la geometria del serbatoio, tale distanza mi consente quindi di assistere a tre possibili scenari:

- **Led giallo acceso**
(troppo pieno = oltre il limite – allarme) - suono buzzer intermittente;
- **Led verde acceso**
(situazione ideale – contenuto recipiente = 20% - 100%);
- **Led rosso acceso**
(troppo vuoto = 0 – 20% – allarme) - suono buzzer intermittente;

IL SOFTWARE DEL MICROCONTROLORE

Allego alcune parti del codice utilizzato, la routine interrupt del misuratore ad ultrasuoni, del termometro nonché il settaggio dei registri.

Sorvolo sulle operazioni matematiche della routine PROGRAMMA_PRINCIPALE e sulle altre routine/subroutine in quanto parti di codice 'facilmente' assemblabili, grazie alle info presenti in rete. Del resto, ogni recipiente possiede caratteristiche geometriche proprie per cui chi decide di realizzare un progetto simile dovrà comunque mettere in conto di dover necessariamente approfondire gli argomenti.

CONCLUSIONI

Il circuito è in fase di collaudo e sembra funzionare al di là di ogni rosea aspettativa, pur tuttavia, tempo permettendo, non escludo di eseguire ulteriori modifiche e implementazioni che saranno oggetto della seconda parte di questo articolo.

Vi ringrazio per l'attenzione e rimango a disposizione per eventuali chiarimenti.



```
ORG    00H
GOTO   PROGRAMMA_PRINCIPALE
```

```
-----
ORG    04H
movwf  w_temp
movf   STATUS,W
bcf    STATUS,RP0
movwf  status_temp
movf   PCLATH,W
movwf  pclath_temp
-----
BTFSC  PIR1,ADIF
GOTO   TEMPERATURA
```

CAPTURE_METRO

```
BTFSS  CCP1CON,0
GOTO   Fronte_discendente
-----
```

Fronte_ascendente

```
BCF    CCP1CON,0
MOVF   CCP1L,W
MOVWF  ATPL
MOVF   CCP1H,W
MOVWF  ATPH
BCF    PIR1,2          ; CCP1IF
BSF    INTCON,7        ; GIE
GOTO   SALTA
```

Fronte_discendente

```
BSF    CCP1CON,0      ; sottraggo i precedenti
MOVF   ATPL,W
SUBWF  CCP1L,W
MOVWF  Lmetro
MOVWF  L_metro
BTFSS  STATUS,C
INCF   TEMPH
NOP
MOVF   ATPH,W
SUBWF  CCP1H,W
MOVWF  Mmetro
MOVWF  M_metro        ; fine della sottrazione
NOP
BSF    Flag,2
BCF    PIR1,2          ; CCP1IF
BSF    INTCON,7
GOTO   SALTA
```

TEMPERATURA

```
BCF    PIR1,ADIF
movf   ADRESH,W
movwf  Valore_convertitoH
BANKSEL ADRESL        ; BANK_1
movf   ADRESL,W
bcf    STATUS,RP0     ; BANK_0
bcf    STATUS,RP1
movwf  Valore_convertitoL
-----
```

SALTA

```
movf   pclath_temp,w
movwf  PCLATH
movf   status_temp,W
movwf  STATUS
swpf   w_temp,f
retfie                                ; Ritorna da interrupt
```

```
----- FINE INTERRUPT -----
-----
```

PROGRAMMA PRINCIPALE

```
-----
; ingressi PORT A analogici, allineamento a dx (10 bit)
; Vref=V alimentazione -- ADRESH:ADRESL
-----
```

```
banksel ADCON1
movlw   10001110b
movwf   ADCON1
clrf    TRISD
banksel ADCON0
```

```
-----
; bit 7-6 settati a 1 - 1 freq. di conversione interna al pic FRC
; bit 5-4-3 settati a 0 selezioniamo il canale AN0 = PORTA,0
-----
```

```
movlw   11000001b    ; 11 Frc
movwf   ADCON0
```

```
-----
; 0=uscita 1=ingresso
-----
```

```
bsf     STATUS,RP0
bcf     STATUS,RP1    ; BANK 1
movlw   00000001b     ; RA0 = INPUT Sonda
temperatura
```

```
movwf   TRISA
movlw   00000000b
movwf   TRISB
movlw   00000100b    ; RC2 / CCP1 = input - capture
movwf   TRISC
movlw   00000000b
movwf   TRISD
movlw   00000001b
movwf   TRISE
```

```
-----
MOVLW   B'00000100'
MOVWF   PIE1
; abilita int. CCP1IE
bcf     STATUS,RP0
bcf     STATUS,RP1    ; BANK 0
MOVLW   B'00000001'
MOVWF   T1CON
MOVLW   B'00000101'
MOVWF   CCP1CON
```

```
-----
MOVLW   00H
MOVWF   PORTB
movwf   Segno
```

```
-----
CALL    DispInit
GOTO    INTR_INIT    ; TERMOMETRO
```

Metro

```
MOVLW   B'11000000'
MOVWF   INTCON
CALL    IMP           ; Subroutine Impulso
```

Chiamata

```
NOP
NOP
NOP
BTFSS   Flag,2
GOTO    Chiamata
CLRF    Flag
BCF     INTCON,7
```

```
.....
.....
```



Ho 47 anni, sono nato e vivo ad Agrigento, sposato con due figli. Dopo il diploma di Geometra ho continuato gli studi all'università di Palermo dove nel 1995 ho conseguito la laurea in Architettura, indirizzo Tutela e Recupero del Patrimonio Storico Architettonico, mi sono subito abilitato all'esercizio

della professione e dopo qualche anno ho anche conseguito un Master di II livello in Sistemi Informativi Territoriali. Pertanto il lavoro che svolgo e la formazione scolastica ed universitaria che possiedo non hanno un'attinenza diretta con l'elettronica! Pur tuttavia mi interessa di informatica e di elettronica sin da giovane effettuando, da autodidatta, un percorso di studi parallelo a quello scolastico/universitario. Dal 2001 mi sono poi soffermato allo studio e alla programmazione dei microcontrollori Pic della Microchip. Sono radioamatore e per hobby amo trascorrere parte del mio tempo libero progettando e/o realizzando circuiti elettronici di mio interesse. Programmo correntemente i microcontrollori PIC in assembly. Ho scritto alcuni articoli di elettronica, uno di questi pubblicato sulla rivista FE n. 375/376 (chiave dtmf a 5 canali). Spero per il futuro di poter continuare a scrivere al fine di fornire alla comunità il mio piccolissimo contributo.



Cubo 4x4x4 a LED BLU

di Giuseppe La Rosa

g.larosa@elettronicaemaker.it

In questo articolo vi proponiamo un progetto semplice e molto interessante che vi consentirà di testare l'illuminazione a LED a partire da una costruzione cubica che potrete animare a vostro piacimento.

Se siete sempre stati affascinati dai LED (quelle lucine vivaci e colorate che esplodono in tanti colori armoniosi) questo è il progetto che fa per voi!

Stiamo parlando di 64 LED (4x4x4) di colore Blu, collegati in multiplexing (montati con una struttura a cubo) e comandati da un microcontrollore PIC16F876A.

La scheda permette di eseguire diversi giochi di luce ed inoltre dispone di due pulsanti: uno per regolare la velocità dei giochi di luce, ed uno per fermare l'esecuzione dei giochi di luce.

Inoltre dispone di un connettore per la programmazione e l'aggiornamento del Firmware senza rimuovere il microcontrollore della scheda (vedi figura 1).

Il tutto può essere alimentato tramite batteria a 9 V o con alimentatore da 9 V.

IL CIRCUITO ELETTRICO

In figura 2 vediamo il complessivo elettronico della scheda del Cubo a LED.

Iniziamo la descrizione dal circuito di alimentazione, l'intero circuito funziona con una tensione di alimentazione di 9 V.

Per questo progetto è stato usato il consueto integrato stabilizzatore L7805CV (IC2), esso fornisce la tensione a 5 V, necessaria ad alimentare il microcontrollore IC1 e la matrice di LED che forma il cubo di LED.

Il microcontrollore IC1 (PIC16F876A) gestisce tutte le operazioni di multiplexer dei LED e di memorizzazione delle sequenze luminose da eseguire. Il clock per il funzionamento del microcontrollore è generato da una rete RC

formata dalla resistenza R24 e dal condensatore ceramico C4.

Il pulsante S1 ha la funzione di regolare la velocità delle sequenze luminose in esecuzione, mentre il pulsante S2 mette in pausa l'esecuzione delle sequenze luminose. Come si evince dallo schema di figura 2 parte della porta RB ed RC sono state usate per pilotare i sedici anodi corrispondenti alle colonne del Cubo a LED, mentre una porzione della porta RA e la rimanente parte della porta RB è stata usata per pilotare le quattro righe o piani (catodi dei LED) del Cubo a LED utilizzando i quattro transistor Q1 a Q4.

Nel prossimo paragrafo passiamo alla descrizione della costruzione del Cubo di LED.

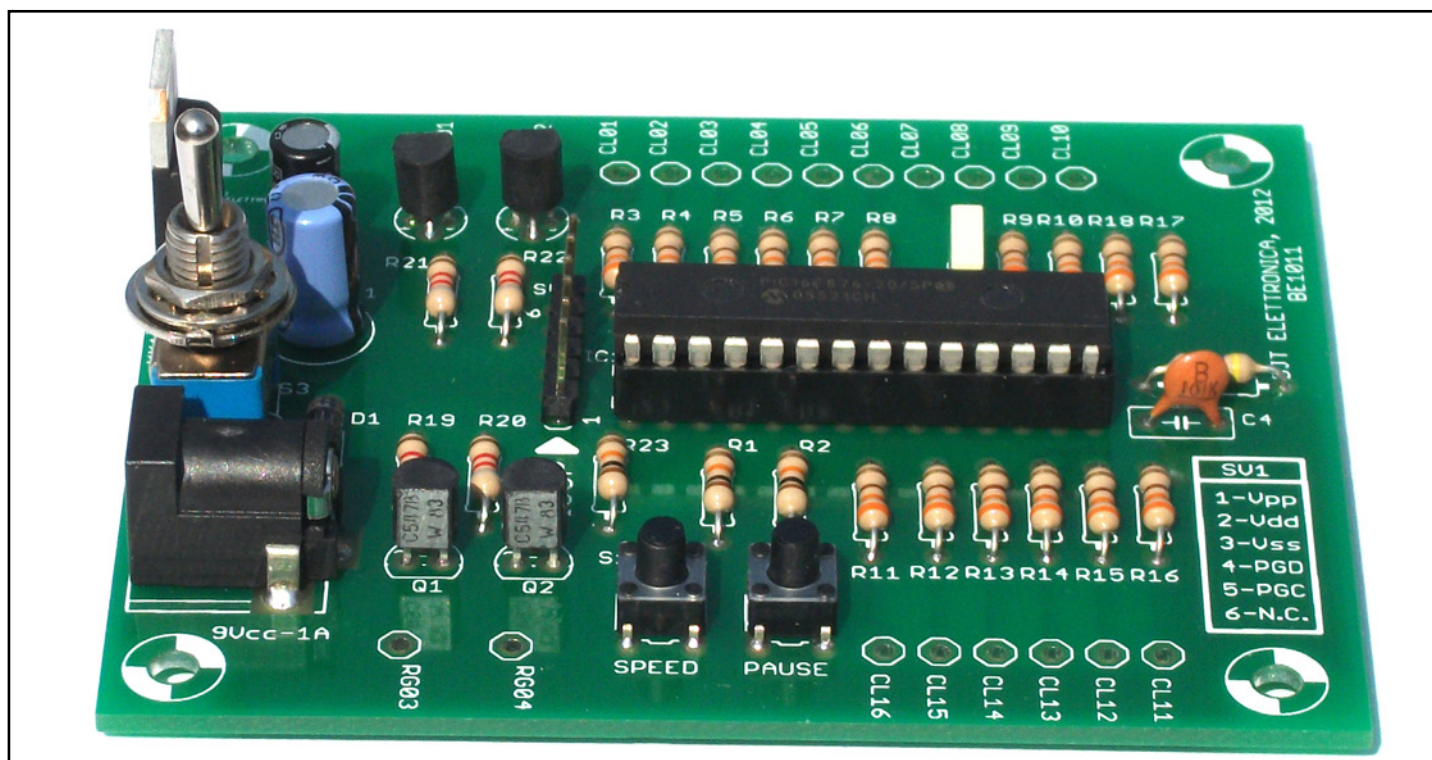


Figura 1: foto della scheda del Cubo a LED

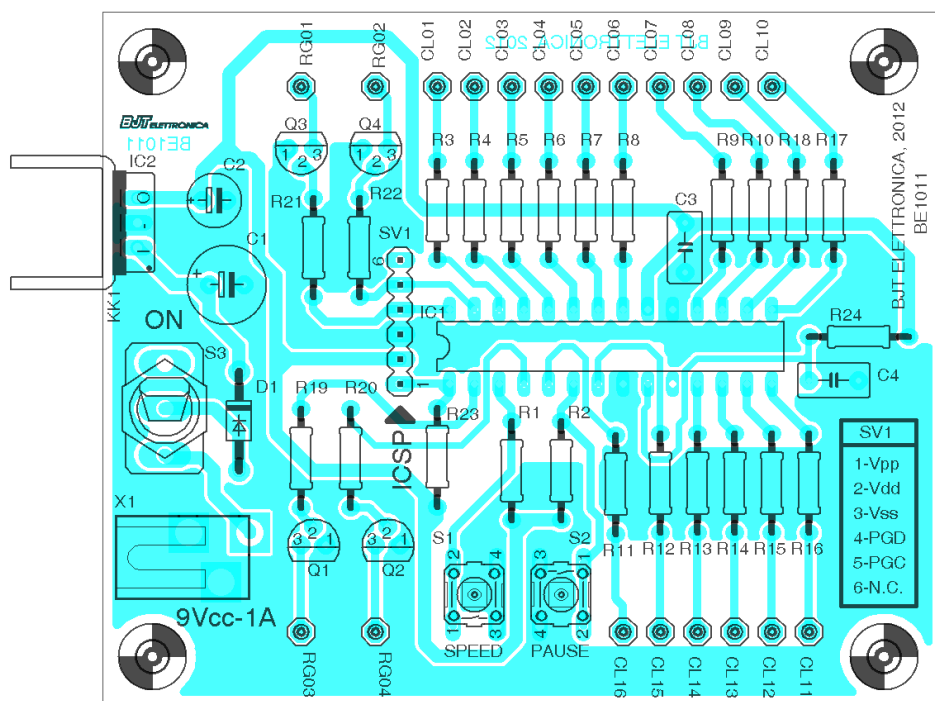


Figura 3: piano di montaggio della scheda del Cubo a LED

di legno lasciate solo il terminale lungo (positivo), servirà per il collegamento agli altri piani. Ogni volta che avrete terminato di saldare un piano testate tutti i LED uno per uno meglio accorgersi subito se qualche LED è bruciato oppure saldato male.

Dovete fare 4 griglie di LED uguali e poi collegarle insieme in modo da formare un cubo come visibile in figura 7.

FIRMWARE E FUNZIONAMENTO

Il Firmware per la gestione della scheda è stato scritto in linguaggio Assembly per PIC ed è possibile scaricarlo al Link che troverete a fine articolo.

Una volta scaricato bisogna compilarlo con MPASM di Microchip e ottenere il “File oggetto binario” con estensione HEX.

Sulla scheda è disponibile un connettore (SV1) a 6 poli che consente la programmazione (ICSP In Circuit Serial Programming) on board del microcontrollore.

Questo permette tramite il programmatore PICKIT2 o PICKIT3 di caricare il “File oggetto binario”, velocizzare la programmazione del microcontrollore e di poter aggiornare velocemente i giochi di luce, inserendo giochi personalizzati ogni volta che si vuole.

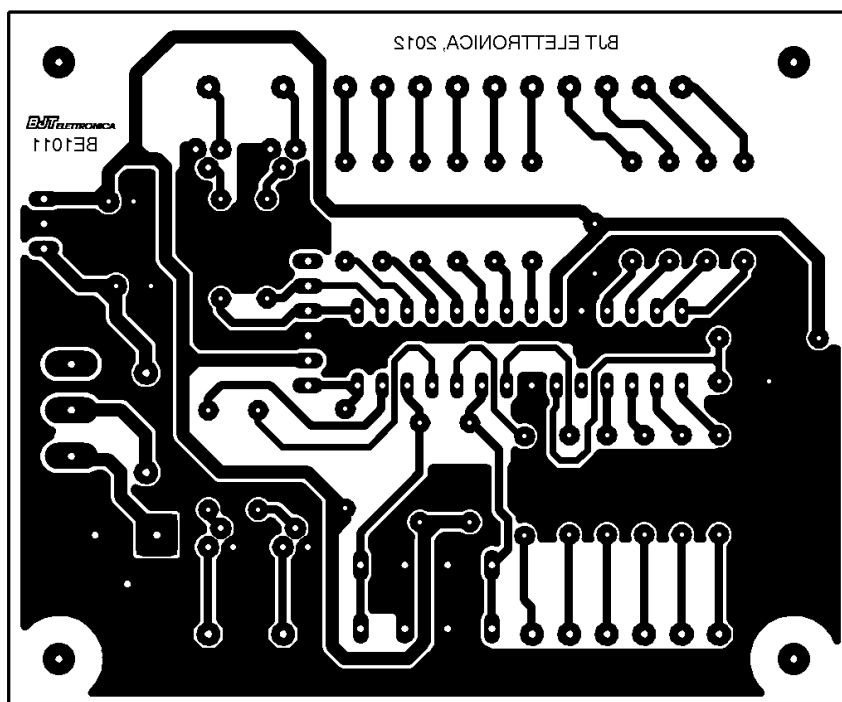


Figura 4: master del lato saldature del circuito stampato della scheda del Cubo a LED

Nel Firmware sono stati implementati 6 giochi di luci, di cui è possibile cambiare la velocità di esecuzione tramite il pulsante S1.

Le velocità di esecuzione sono quattro, essa aumenta di uno Step ad ogni pressione del pulsante S1 per poi diminuire alla quinta pressione del pulsante.

Il pulsante S2 ferma la sequenza dei giochi e rimane in pausa, anche se viene staccata l'alimentazione e riattaccata in quanto lo stato del pulsante viene salvato nell'EEPROM insieme alla velocità dei giochi di luce.

Le sequenze sono modificabili tramite

le tabelle "COL" (per le colonne) e "RIG" (per le righe) che si trovano alla fine del listato del sorgente.

Nella prossima sezione passeremo alla realizzazione della scheda.

REALIZZAZIONE DELLA SCHEDA E COLLAUDO

Passiamo adesso alla realizzazione della scheda che si presenta abbastanza semplice, la scheda è del tipo mono faccia e si prepara a partire dalle tracce di figura 4.

Ottenuto il circuito stampato, iniziate a montare i componenti sulla scheda (se-

guendo il piano di montaggio figura 3) e i componenti richiesti dall'“Elenco componenti”.

Inserite tutte le resistenze come da “Elenco componenti”, proseguite con il diodo D1.

Saldare lo zoccolo per l'integrato IC1, proseguite con i condensatori non polarizzati e poi gli elettrolitici, lo Strip maschio SV1, il connettore X1, poi i transistor Q1 a Q4, e IC2, ed infine l'interruttore da circuito stampato S3.

In figura 5 è rappresentata la scheda ed i collegamenti che si devono eseguire al cubo usando del flat cable a 20 pin.

Tutti i collegamenti in rosso rappresentano le colonne, esse sono nominate con CL0X, rispettivamente devono essere collegati tra la scheda e il cubo, ad esempio CL01 della scheda va collegato con CL01 del cubo è così per tutte le restanti colonne. Stessa procedura per le righe (RG0X in nero) sempre usando del flat cable collegare RG01 della scheda con RG01 del cubo, è così per le righe restanti.

L'alimentazione come mostra la figura 5 può essere alimentato tramite alimentatore da 9 V/1 A con connettore da 5,5x2,1 mm, ma per applicazioni porta-

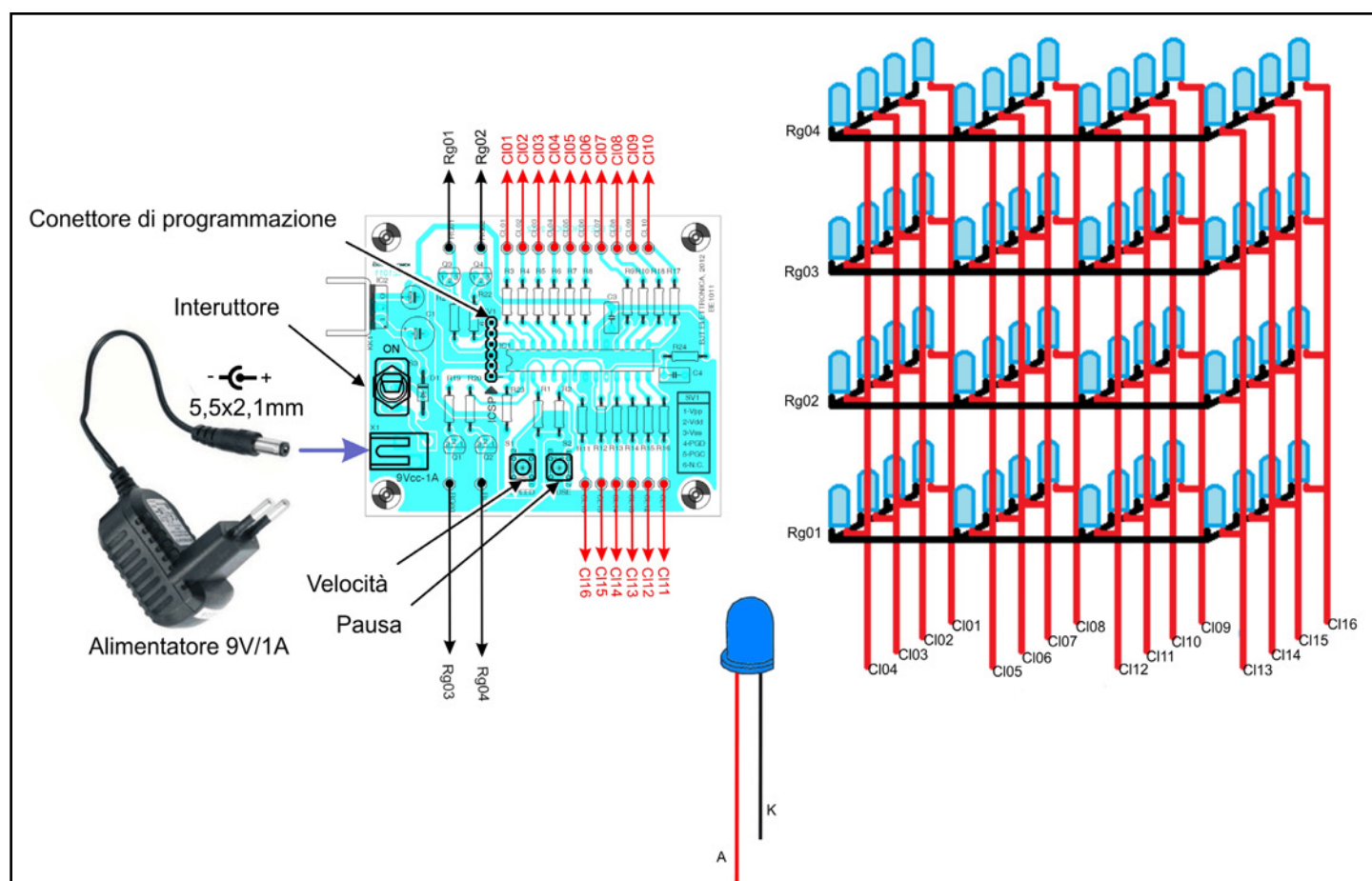


Figura 5: schema di collegamento tra la scheda e il Cubo di LED.



tili si può usare una batteria alcalina da 9 V.

Tramite l'interruttore S3 si può spegnere e accedere il cubo, tramite il pulsante S1 si può cambiare la velocità di esecuzione dei giochi di luce mentre con il pulsante S2 si mettono in pausa i giochi di luce. In figura 8 è illustrato il disegno quotato della scheda in cui sono riportati i quattro fori da praticare per il fissaggio ad un contenitore o basetta di legno come visibile in figura 7.

Per il fissaggio si possono usare distanziatori in metallo da 3 mm o distanziatori adesivi sempre da 3 mm.

Il progetto è stato pensato per essere racchiuso con facilità dentro qualsiasi contenitore anche auto costruito.

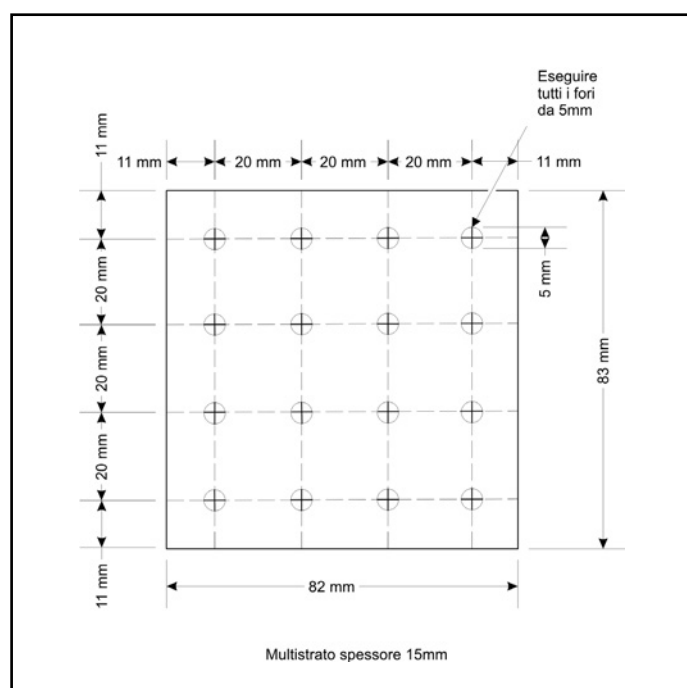


Figura 6: piano di foratura della dima per il posizionamento dei LED

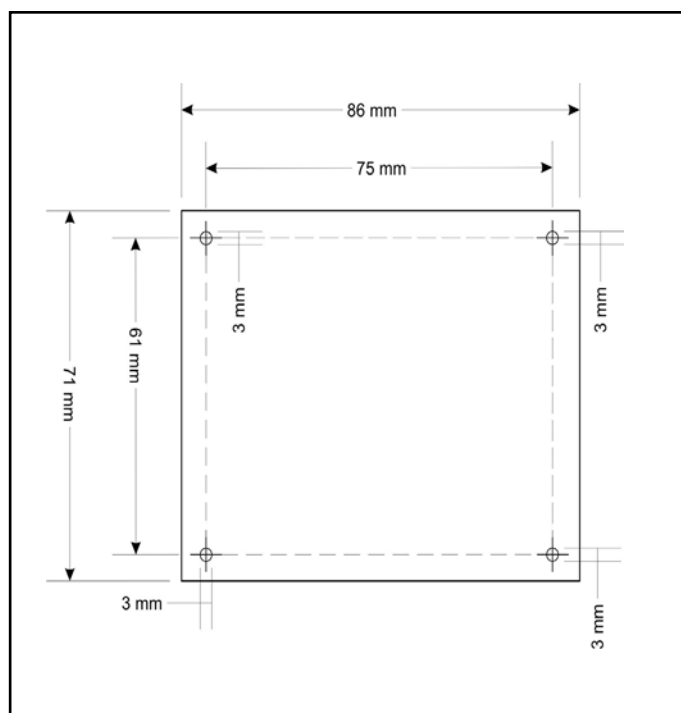


Figura 8: piano di foratura della scheda del Cubo a LED

CONCLUSIONE

Le possibilità di migliorare questa scheda sono infinite perché una volta che avete capito la logica, sarà facile per voi immaginare una matrice più grande, con LED dalle caratteristiche diverse. E così vi ritroverete in un battibaleno a fare esperimenti sempre originali, interessanti e divertenti.

Di seguito trovate il Link per scaricare i File per la realizzazione del progetto.



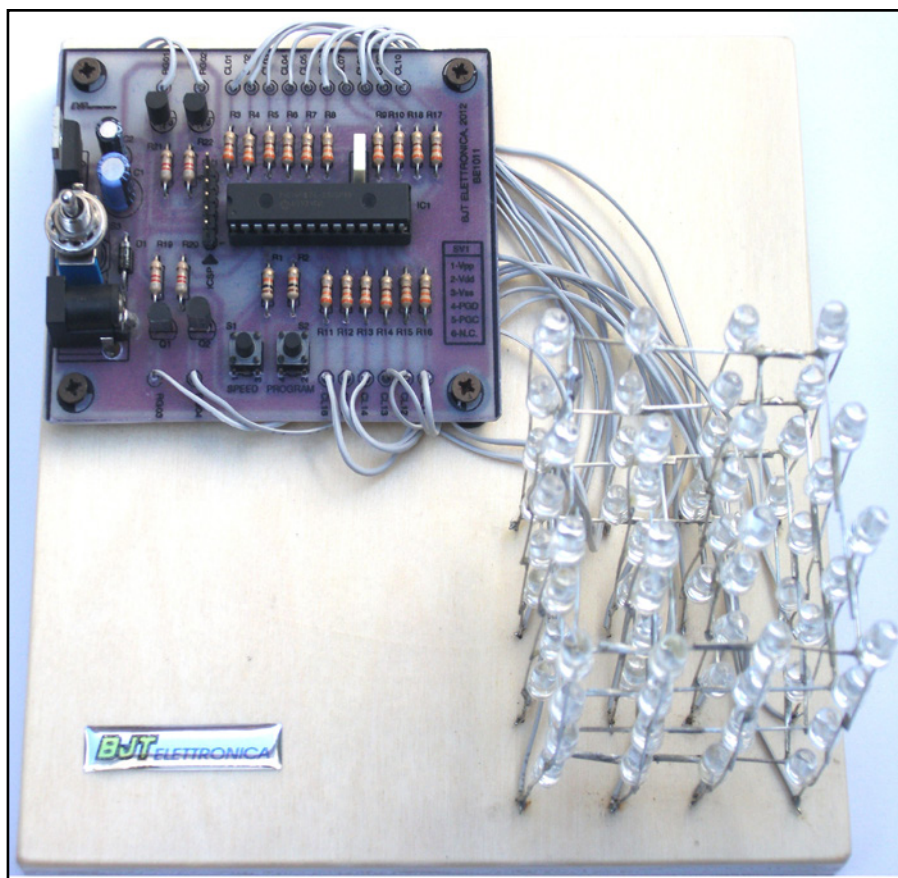


Figura 7: cubo a LED 4x4x4 montato su una basetta in multistrato

Qui sotto, invece, trovate il collegamento ad un video dove potrete vederlo in funzione, sia nella versione descritta in questo progetto, sia con un'altra interessante e scenografica disposizione della matrice di diodi. Per eventuali domande resto a vostra disposizione, grazie per l'attenzione.



Elenco componenti

Resistenze

R1÷R2 10 K Ω 1/4 W

R3÷R18 330 Ω 1/4 W

R19÷R22 8,2 K Ω 1/4 W

R23 10 K Ω 1/4 W

R24 4,7K Ω 1/4 W

Condensatori

C1 100 μ F 35 V elettrolitico

C2 10 μ F 35 V elettrolitico

C3 100 nF poliestere

C4 100 pF ceramico

Diodi

D1 1N4007 diodo

Transistor

Q1÷Q4 BC547

Integrati

IC1 PIC16F876A

IC2 L7805CV

Varie

S1÷S2 Pulsante c.s.

S3 Interruttore c.s.

X1 Presa DC 90° 5,5X2,1 mm

SV1 Strip maschio 6 pin

CL01÷CL16 Ancorante c.s.

RG01÷RG04 Ancorante c.s.

KK1 Dissipatore TO220

N.1 Zoccolo 14+14 pin

N.64 LED Blu alta luminosità

N.1 Flat cable a 20 poli



Sono Giuseppe La Rosa. Appassionato di "elettricità" sin da giovane, ho terminato gli studi in Elettronica e Telecomunicazioni durante l'anno 2002 presso I.T.I.S. "G. Ferraris" di Acireale. In

seguito ho cominciato lo studio per passione e da autodidatta di sistemi a Microcontrollori; in particolare i microcontrollori PIC prodotti da Microchip e poi la piattaforma Open Source Arduino UNO. Nel corso degli anni ho maturato varie esperienze che mi permettono di realizzare diversi prototipi, di cui molti progetti sono stati pubblicati in varie riviste del settore: tra cui: Fare Elettronica, CQ Elettronica e ultimamente nel blog di EMC Elettronica per cui ho collaborato per la realizzazione di vari progetti. Attualmente mi occupo di sistemi di sicurezza (videosorveglianza e anti intrusione) e software per la gestione dei punti vendita. Spero che i miei articoli possano essere di aiuto per le vostre realizzazioni personali.



Riavvio Automatico per Caricabatteria

di Victor Khasiev

Sr. Applications Engineer,

Power by Linear

Analog Devices Inc.

L'ostacolo principale a cui alcuni progettisti devono far fronte è l'assenza di funzioni automatiche o d'attivazione automatica necessarie per un caricabatteria per celle agli ioni di litio che si possa definire completamente funzionale

Non sempre i circuiti caricabatteria possiedono le caratteristiche di "intelligenza" necessarie per poterli definire e considerare del tutto funzionali. Per esempio, il regolatore è in grado di rilevare la corrente C/10 ma non interrompe il processo di carica quando la corrente di carica è ridotta a un valore C/10. Esiste inoltre il problema dell'assenza della funzionalità di riavvio automatico, necessaria per qualificare completamente il caricabatteria.

Il regolatore LTC4012 è un chip molto diffuso per la carica di batterie agli ioni di litio in un'ampia gamma di applicazioni. Offre all'utente ingressi gate driver e di rilevamento della corrente per consentire di costruire un gruppo di alimentazione con topologia step-down, oltre a segnali di controllo come quelli di stato della carica e di presenza dell'adattato-



re. È disponibile anche una gamma di funzioni utili descritte dettagliatamente nella scheda dati.

In applicazioni in cui non si utilizza la CPU, i progettisti preferiscono non usare del tutto l'LTC4012 o creare soluzioni sviluppate internamente. Per esempio, alcuni utilizzerebbero un resistore dal pin PROG per il rilevamento della corrente di carica, collegando tale pin a un buffer a bassa impedenza e dal costo contenuto, praticamente alterando la misura.

Il presente articolo si propone di suggerire una semplice soluzione di riavvio automatico che fa a meno del pin PROG e utilizza invece il segnale logico per il rilevamento della corrente C/10.

Tale rilevamento è basato sul circuito "Indicatore C/10 digitale" descritto nella scheda dati dell'LTC4012 e mostrato nel diagramma a blocchi.

DIAGRAMMA A BLOCCHI DEL CIRCUITO A RIAVVIO AUTOMATICO

Il diagramma a blocchi del circuito suggerito è presentato nella Figura 1. Il circuito è basato su un flip-flop D. Quando il valore della corrente di carica si riduce al valore preprogrammato C/10, il clock del flip-flop D ne porta l'uscita allo stato alto, il transistor esterno Q1 porta il pin RUN a massa e disinserisce l'LTC4012. La batteria viene caricata e la tensione è prossima al valore massimo. Con il caricabatteria disinserito, la batteria inizia

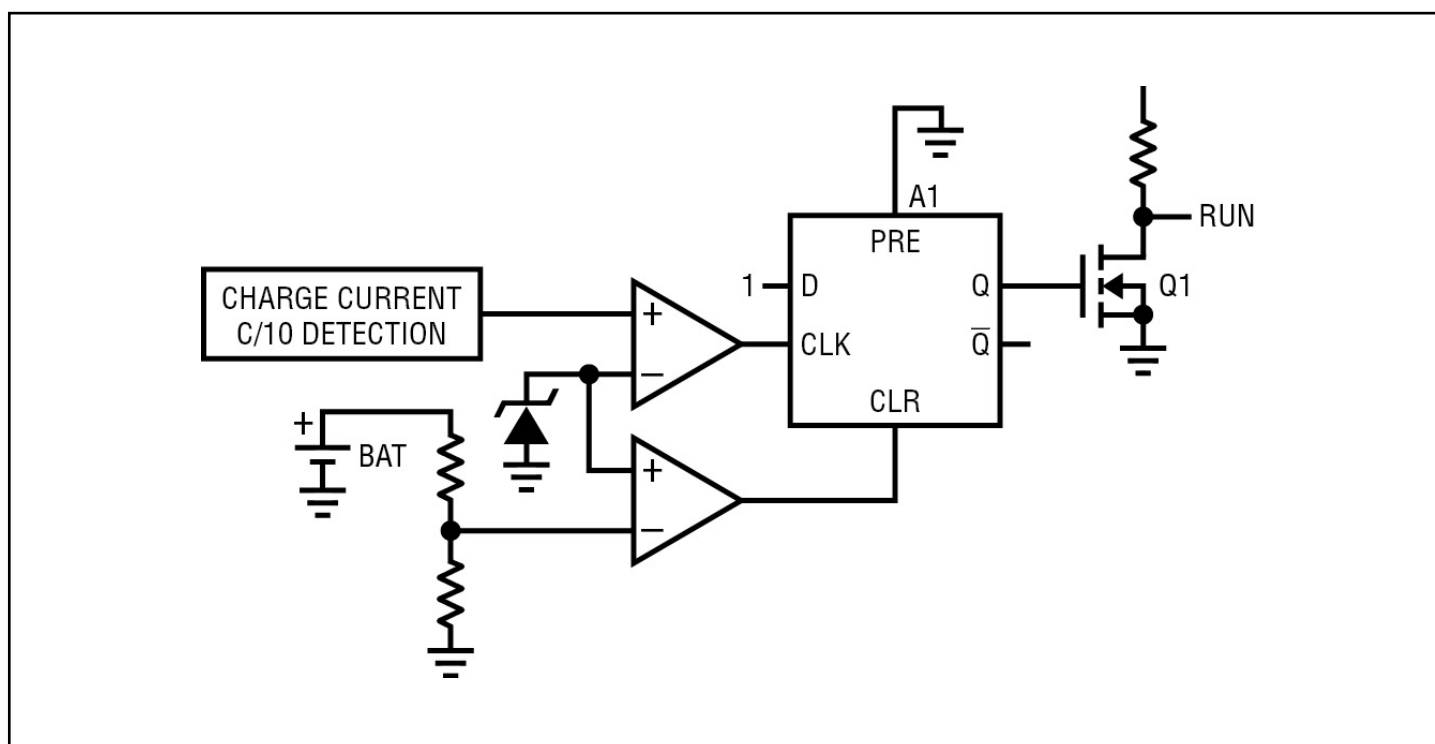


Figura 1: Diagramma a blocchi del circuito a riavvio automatico.

a scaricarsi, raggiungendo alla fine la tensione minima, e l'uscita del flip-flop D torna allo stato basso, quindi Q1 si disinserisce e il ciclo di carica si ripete. Per realizzare la maggior parte delle funzioni descritte in precedenza si è selezionato il dispositivo di monitoraggio della tensione LTC2912-3, dotato di funzione di controllo del bistabile, visibile al centro dello schema elettrico presentato nella Figura 2.

SCHEMA ELETTRICO E DESCRIZIONE DEL CIRCUITO

Si è provata una batteria agli ioni di litio da 12V e 3000 mA/h, caricandola mediante il caricabatteria montato sulla scheda demo DC1256A, che presenta

l'LTC4012-2. Il caricabatteria è controllato dal circuito di riavvio automatico mostrato nella Figura 2.

Se la batteria è carica, la tensione all'ingresso VH dell'LTC2912-3 è maggiore di 0,5V e l'uscita /UV è allo stato alto. Q2 è inserito, il pin /LATCH è allo stato basso e l'uscita OV viene mantenuta a uno stato alto. Corrispondentemente, Q1 è inserito e l'LTC4012 è nella modalità di disinserimento. Fintantoché il pin /LATCH è allo stato basso e OV viene mantenuta allo stato alto, l'LTC4012 rimane disinserito.

Alla fine, a causa dell'autoscarica del carico applicato, la tensione della batteria inizia a diminuire; quando scende sotto 12V, la tensione al pin VH (parti-

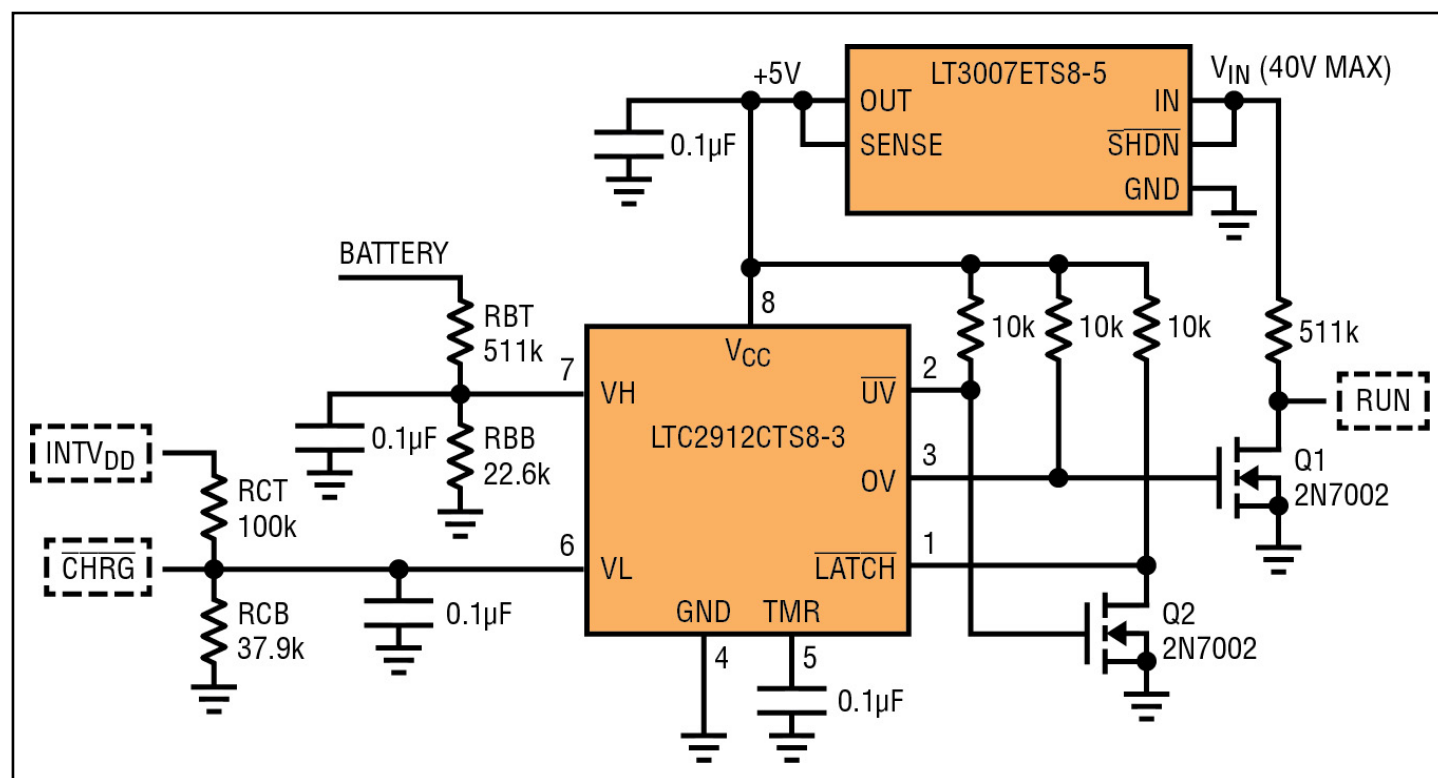


Figura 2: Figure 2. Schema elettrico della soluzione proposta

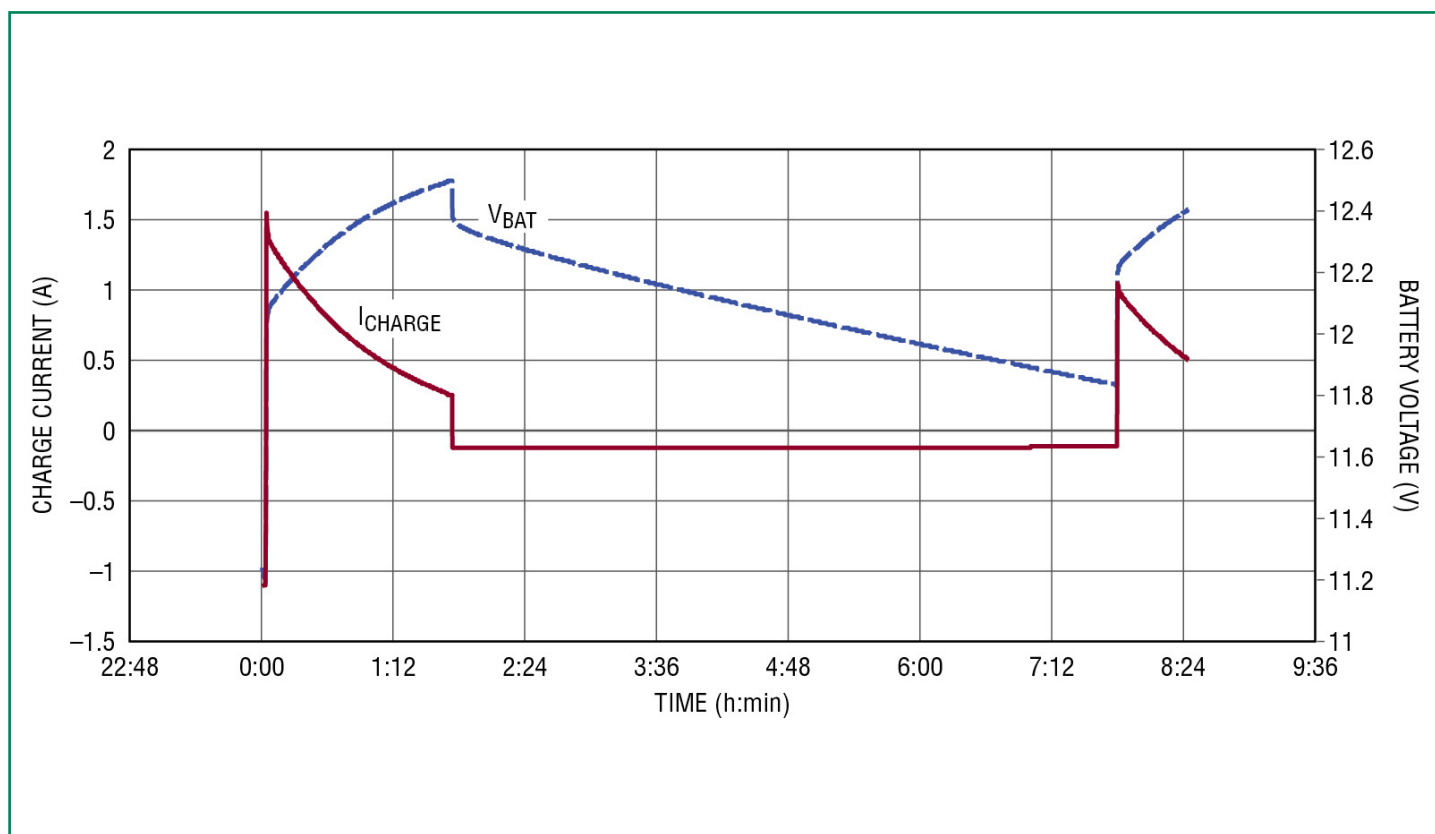


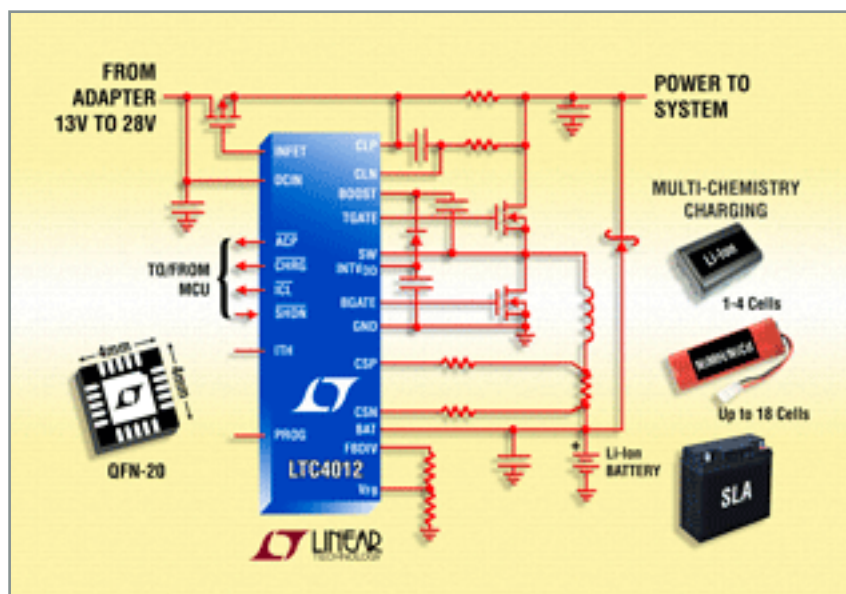
Figura 3: Diagramma di temporizzazione del ciclo di riavvio automatico

tore resistivo RBT/RBB) scende sotto il livello di 0,5V e l'uscita /UV passa dallo stato alto a quello basso. Quindi Q2 si disinserisce, resettando il bistabile. L'uscita OV del segnale passa allo stato basso, Q1 si disinserisce e viene avviato il processo di carica.

Durante il processo di carica, il pin /CHRG dell'LTC4012 è in uno stato di bassa impedenza e mantiene il pin VL collegato a GND. Quando la corrente di carica si riduce al livello C/10, il pin /CHRG cambia stato e la tensione sul pin VL aumenta. Quando la tensione di questo pin raggiunge 0,5V (partitore resistivo RCT/RCB), il pin OV passa dallo stato alto allo stato basso, Q1 si inseri-

sce, il pin RUN passa allo stato basso e l'LTC4012 va nella modalità di disinserimento. Il pin /LATCH è allo stato basso e U1 mantiene l'uscita OV allo stato alto finché la tensione della batteria non scende di nuovo a 12V e il processo si ripete. Il circuito suggerito si interconnette con l'LTC4012 e i pin corrispondenti sono cerchiati dalla linea tratteggiata per distinguerli dai pin U1 dell'LTC2912. Verifica del circuito e misure

Per verificare la funzionalità del circuito, si è collegato un data logger al sistema caricabatteria-batteria. La tensione e corrente della batteria sono state campionate e memorizzate. Per facilitare



molto diffuso, l'LTC4012. Questo circuito consente la realizzazione di una funzione di riavvio automatico utilizzando un singolo dispositivo di monitoraggio della tensione LTC2912-3, senza bisogno di impiegare sistemi di controllo digitali alquanto costosi.

l'autoscarica della batteria e ridurre il tempo di scarica, si è collegato il resistore da 100Ω tra i poli della batteria. Il grafico che ne risulta è presentato nella Figura 3. All'inizio del ciclo, la tensione d'ingresso VIN è disinserita, la batteria è in fase di scarica e invia corrente 'negativa' sul carico. Non appena si applica VIN si avvia il ciclo di carica e nella batteria circola la corrente di carica 'positiva'. Quando la batteria è completamente carica, il caricabatteria si disinserisce e la batteria passa alla fase di autoscarica. Quando la tensione della batteria raggiunge il valore minimo, il caricabatteria si inserisce e il ciclo si ripete.

CONCLUSIONE

Il circuito relativamente semplice suggerito nel presente articolo e le applicazioni basate su di esso possono costituire un accessorio importante per il caricabatteria di batterie agli ioni di litio

Bibliografia

1. <http://cds.linear.com/docs/en/datasheet/4012fa.pdf>, scheda dati del caricabatteria LTC4012
2. <http://cds.linear.com/docs/en/datasheet/4012fa.pdf>, LTC4012-2, DC1286A, circuiti demo
3. <http://cds.linear.com/docs/en/datasheet/2912fb.pdf>, scheda dati del dispositivo di monitoraggio della tensione LTC2912

In Copertina:

http://www.esa.int/spaceinimages/Images/2013/02/Silicon_wafers_etched_with_integrated_circuits_for_space_missions



Elettronica & Maker

DIRETTORE

Roberto Armani

Art Director

Shylock-58

Hanno collaborato a questo numero:

Giovanni Carrera, Antonello Della Pia, Girolamo D'Orio, Victor Khasiev, Giuseppe La Rosa, Angelo Lezza, Emanuele Paiano, Mario Rotigni, Ivan Scordato, Roberto Vallini

AVVERTENZE

Chiunque decida di fare uso delle nozioni riportate in questi articoli o decida di realizzare i circuiti esposti, è tenuto a prestare la massima attenzione in osservanza alle normative in vigore sulla sicurezza. Gli Autori di Elettronica&Maker sopracitati, che hanno collaborato alla realizzazione degli articoli pubblicati in questo numero, declinano ogni responsabilità per eventuali danni causati a persone, animali o cose derivante dall'utilizzo diretto o indiretto del materiale, dei dispositivi o del software presentati. Si avverte inoltre che quanto riportato negli articoli viene fornito così com'è, a solo scopo hobbistico, senza garanzia alcuna di correttezza e di funzionamento certo. L'editore e gli autori ringraziano anticipatamente per la segnalazione di ogni eventuale errore.

Su Elettronica & Maker

Elettronica&Maker è una testata pubblicata in formato esclusivamente elettronico e sfogliabile elettronicamente, sul sito web <https://www.elettronicaemaker.it>, diffusa esclusivamente per via telematica, non soggetta all'obbligo di registrazione presso il Tribunale, né al R.O.C. né agli obblighi dell'AgCom n° 666/08 del 26/11/08, a fronte del D.L. n° 63 del 18 Maggio 2012.

© Copyright

Tutti i diritti di riproduzione o di traduzione degli articoli pubblicati sono riservati. Manoscritti, disegni e fotografie sono di proprietà di E&M

È vietata la riproduzione anche parziale degli articoli salvo espressa autorizzazione scritta dell'editore. I contenuti pubblicitari sono riportati senza responsabilità, a puro titolo informativo.

Collaborare con Elettronica & Maker

Le richieste di collaborazione vanno indirizzate all'attenzione di redazione@elettronicaemaker.it e accompagnate, se possibile, da una breve descrizione delle vostre competenze tecniche e/o editoriali, oltre che da un elenco degli argomenti e/o progetti che desiderate proporre.