MENSILE DI PROGETTAZIONE ELETTRONICA · ATTUALITÀ SCIENTIFICA · NOVITÀ TECNOLOGICHE

Elettronica In

www.elettronicain.it

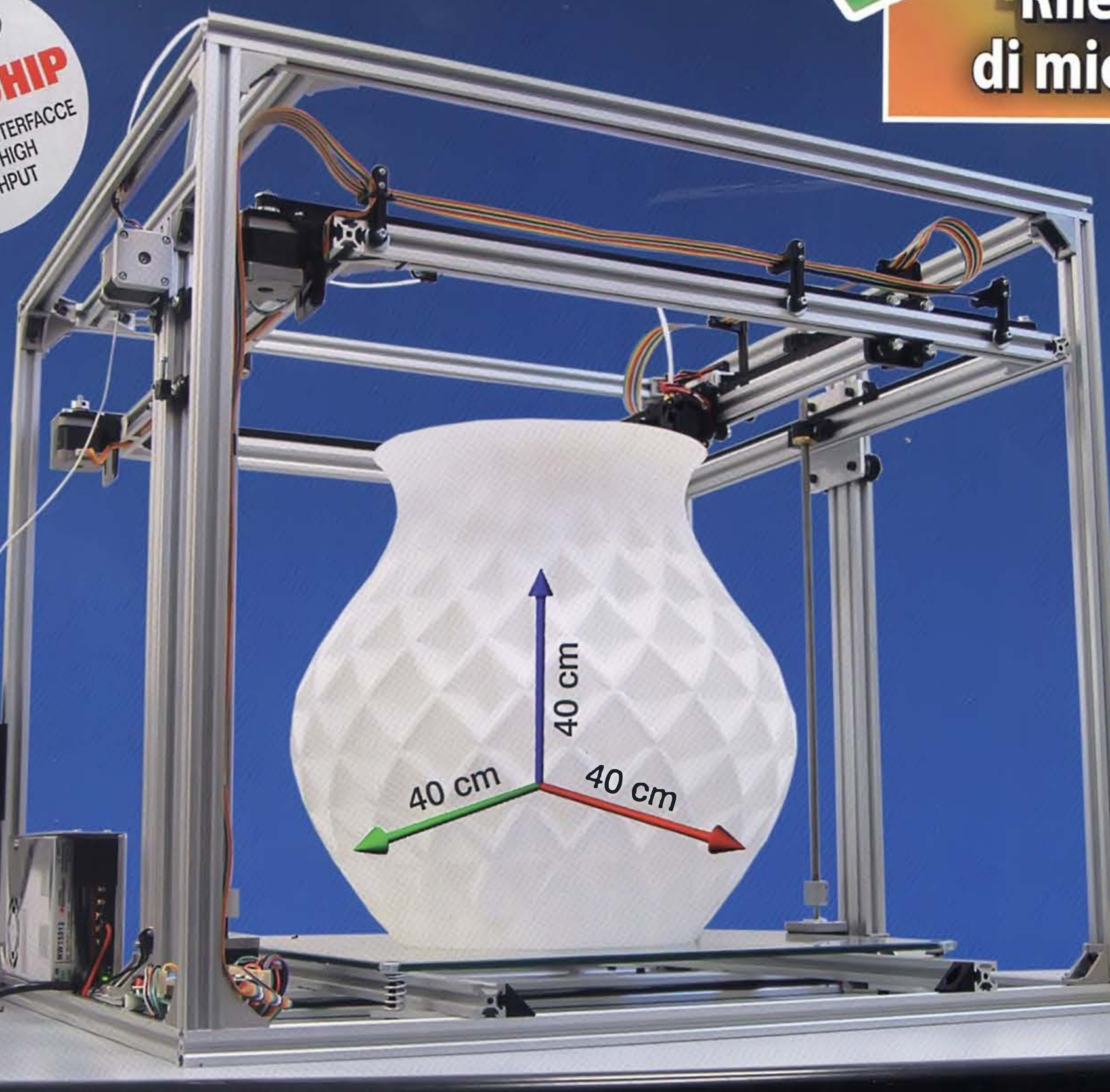
€ 6,00 - Anno XXIII - n. 219 - OTTOBRE 2017

3D4040: stampa in GRANDE!



Rilevatore di microspie

In omaggio MICROCHIP
SOLUZIONI PER INTERFACCE ETHERNET HIGH THROUGHPUT



- STEM Shield, board di sperimentazione per Arduino
- Programmatore con Pololu USB AVR v2
- Una web app per gestire Arduino con Raspberry Pi
- Anti blackout per telecontrolli GSM
- Sensore di qualità dell'aria
- Tracking del viso con RandA

CORSO GRATUITO

Poste italiane Spa - Spedizione in abbonamento Postale: D.L. 353/2003 (conv. in L. 27/02/2004 n°46) art. 1 - comma 1 - DCB Milano. Prima immissione: 03/10/2017.

Futura Energy

Fai il pieno di energia con le nuove batterie alcaline di *FUTURA ELETTRONICA!*



Prezzi IVA inclusa.

ELEVATA CAPACITÀ AD UN PREZZO IMBATTIBILE!

BATTERIE ALCALINE LR20 D - 1,5V

Blister da 2 pezzi

€ 3,20

Cod. **LR20-D-2**

BATTERIE ALCALINE LR14 C - 1,5V

Blister da 2 pezzi

€ 2,30

Cod. **LR14-C-2**

BATTERIE ALCALINE LR6 AA 1,5V

Blister da 4 pezzi

€ 1,50

Cod. **LR6-AA-4**



BATTERIE ALCALINE LR3 AAA - 1,5V

Blister da 4 pezzi

€ 1,50

Cod. **LR03-AAA-4**



BATTERIA ALCALINA 6LR61 - 9V

Blister da 1 pezzo

€ 1,80

Cod. **6LR61-9V-1**



BATTERIE ALCALINE 23A - 12V

Blister da 5 pezzi

€ 2,50

Cod. **12V-23A-5**

FUTURA ELETTRONICA®

www.futurashop.it

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche di questi prodotti e acquisti on-line su *www.futurashop.it*

Elettronica In

www.elettronicain.it

Rivista mensile, anno XXIII n. 219
OTTOBRE 2017

Direttore responsabile:
Arsenio Spadoni
(*Arsenio.Spadoni@elettronicain.it*)

Redazione:
Stefano Garavaglia, Paolo Gaspari, Boris Landoni, Davide Scullino, Alessandro Sottocornola
(*redazione@elettronicain.it*)

Grafica:
Alessia Sfulcini, Filippo Gori
(*Alessia.Sfulcini@elettronicain.it*)

Ufficio Pubblicità:
Monica Premoli (0331-752668)
(*Monica.Premoli@elettronicain.it*)

Ufficio Abbonamenti:
Elisa Guarniero (0331-752668)
(*Elisa.Guarniero@elettronicain.it*)

DIREZIONE, REDAZIONE, PUBBLICITÀ:
FUTURA GROUP srl - Divisione Editoriale
via Adige 11 - 21013 Gallarate (VA)
Telefono 0331-752668 Fax 0331-792287

Abbonamenti:
Annuo 10 numeri Euro 45,00
Estero 10 numeri Euro 45,00 (digitale)
Le richieste di abbonamento vanno inviate a:

FUTURA GROUP srl
via Adige 11, 21013 Gallarate (VA)
tel. 0331-752668.

Distribuzione per l'Italia:
SO.DI.P. Angelo Patuzzi S.p.A.
via Bettola 18 - 20092 Cinisello Balsamo (MI)
Telefono 02-660301 Fax 02-66030320

Stampa:
ROTO3 Spa - Via Turbigo, 11/b
20022 CASTANO PRIMO (MI)

Elettronica In:
Rivista mensile registrata presso il Tribunale di Milano con il n. 245 il 03/05/1995.
Prezzo di copertina Euro 6,00. Gli arretrati nei formati cartaceo e digitale (pdf) sono acquistabili sul sito della rivista al prezzo di Euro 6,00.
Poste Italiane Spa - Spedizione in abbonamento Postale - D.L. 353/2003 (conv. in L. 27/02/2004) art. 1 comma 1 - DCB Milano.
FUTURA GROUP srl è iscritta al Registro Operatori della Comunicazione n. 23650 del 02/07/2013.
Impaginazione ed immagini sono realizzati in DeskTop Publishing con programmi Adobe InDesign e Adobe Photoshop per Windows.
Tutti i contenuti della Rivista sono protetti da Copyright. Ne è vietata la riproduzione, anche parziale, la traduzione e più in generale la diffusione con qualsiasi mezzo senza l'autorizzazione scritta da parte dell'Editore. I circuiti, il firmware ed il software descritti sulla Rivista possono essere realizzati solo per uso personale, ne è proibito lo sfruttamento a carattere commerciale e industriale. Tutti possono collaborare con Elettronica In. L'invio di articoli, materiale redazionale, programmi, traduzioni, ecc. implica da parte del Collaboratore l'accettazione dei compensi e delle condizioni stabilite dall'Editore (*www.elettronicain.it/ase.pdf*). Manoscritti, disegni e foto non richiesti non verranno in alcun caso restituiti. L'utilizzo dei progetti e dei programmi pubblicati non comporta alcuna responsabilità da parte della Società Editrice.

© 2017 FUTURA GROUP srl

Samsung e Intel, uno storico sorpasso

Dopo anni di incontrastato dominio in vetta alla classifica dei produttori di semiconduttori, Intel ha dovuto cedere lo scettro a Samsung: nel secondo trimestre di quest'anno, infatti, i ricavi del colosso coreano hanno superato quelli della multinazionale di Santa Clara. Non che il bilancio di Intel sia un cattivo bilancio. Tutt'altro. Semplicemente Samsung è cresciuta più rapidamente ed ha superato di poco nel trimestre il fatturato di Intel.

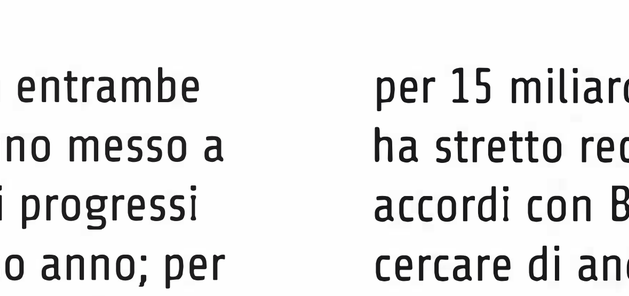
Ma veniamo ai dati. Nel secondo trimestre del 2017 Samsung ha fatturato 15,8 miliardi di dollari contro i 14,4 di Intel, con entrambe le società che hanno messo a segno significativi progressi rispetto allo scorso anno; per quanto riguarda gli utili, fanno impressione quelli di Samsung (7,2 miliardi) anche se i 3,8 miliardi di Intel non sono certo noccioline.

Samsung, dunque, nonostante gli ottimi risultati di Intel, si appresta (la conferma la avremo a fine anno) a occupare ufficialmente il primo gradino del podio.

Merito soprattutto della varietà dell'offerta che spazia dalle memorie all'IoT, dal wearable al "mobile", tutti settori in forte crescita. Al contrario oltre il 50% dei ricavi di Intel è dovuto ai processori per PC, un mercato in lento calo. Sicuramente Samsung è stata più veloce ad intercettare le nuove tendenze del mercato globale, ed ora ne trae il meritato frutto.

Questo tuttavia non significa che Intel, dopo aver perso il treno del "mobile", che sicuramente le è costato il primo posto in classifica, non stia attuando una serie di iniziative per intercettare nuovi business in arrivo. Come quello dei processori per data center, un mercato in rapida crescita, ma soprattutto quello dell'automotive e della guida autonoma. Dopo l'acquisizione della società israeliana Mobileye per 15 miliardi di dollari, Intel ha stretto recentemente forti accordi con BMW e FCA per cercare di andare ad occupare una posizione di primo piano in un mercato che si profila ancora più ampio di quello del "mobile".

Insomma, non è detto che il regno di Samsung duri a lungo.





Arsenio Spadoni
(*Arsenio.Spadoni@elettronicain.it*)

Sommario

Ottobre 2017
n. 219



IN COPERTINA:
Eccola, finalmente, la nuova stampante 3D per oggetti di grandi dimensioni, 40 x 40 x 40 centimetri: un progetto che ha impegnato il nostro laboratorio per oltre un anno!

ELENCO INSERZIONISTI

- Fiera di Scandiano
- Futura Elettronica
- Linear Technology
- Maker Faire Rome
- Microchip
- RM Elettronica
- TEKO
- www.mdsrl.it

ARTICOLI

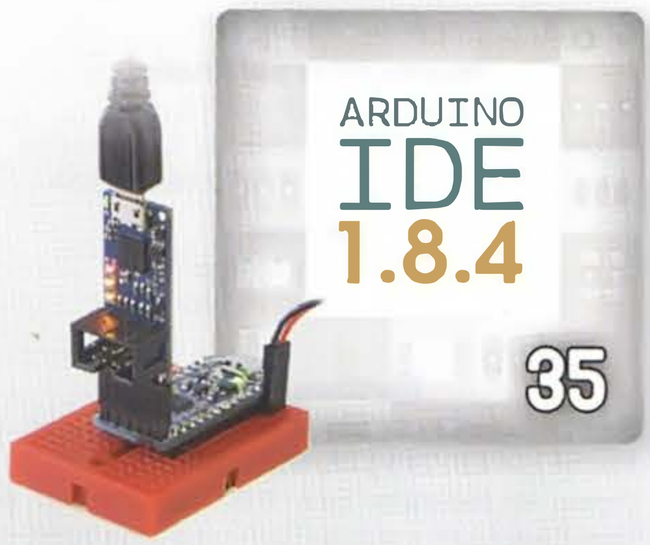
25 ARDUINO
STEM SHIELD

Una board di sperimentazione delle capacità delle schede Arduino, ricca di periferiche specifiche per il mondo Educational.



35 DIDATTICA
PROGRAMMAZIONE AVR STAND-ALONE

Impariamo a programmare i microcontrollori Atmel della serie AVR utilizzando l'IDE Arduino e il nuovo programmatore Pololu USB AVR v2.



41 RASPBERRY PI
HTML5 WEB APP

Controlliamo Arduino tramite una web app residente su Raspberry Pi, utilizzando jQuery Mobile, PHP5 e MySQL, per gestire un hardware che legge lo stato di 4 sensori e comanda altrettanti relé.



65 AUTOMAZIONE
GRUPPO DI CONTINUITÀ PER TELECONTROLLI GSM

Concepito per la serie di telecontrolli di Futura Elettronica, consente di mantenere il circuito in funzione, grazie a una batteria agli ioni di litio, in assenza dell'alimentazione principale.



*I prezzi si intendono iva esclusa e calcolati sul singolo pezzo - ordine minimo 2 pezzi

CIRCUITI STAMPATI IN 24 ORE

GARANTIAMO IL TEMPO DI CONSEGNA!

24 ORE O I CIRCUITI SONO GRATIS

Produciamo circuiti stampati da 1 a 6 strati,
su FR4 di altissima qualità Panasonic® e su Roger®,
con solder e serigrafie per uno stampato di alta qualità
o solo piste stagnate per un prototipo a basso costo.

Prezzi a partire da* **€ 13.68** (doppia faccia
foro metallizzato 75x75 mm) e da **€ 17.79**
(**4 strati** 75x75 mm) per FR4 Panasonic® 1.6mm
comprensivi di attrezzatura e test elettrico.

Nessuna limitazione sul numero dei fori, sul numero
degli utensili (diametri) e sul tipo di scontornatura
(anche tondeggiante).
Isolamenti minimi 6 mils, foro minimo 0,2 mm
senza sovrapprezzo.



PREVENTIVO ANONIMO, GRATUITO ED IMMEDIATO
con il nostro calcolatore online.

Millennium Dataware Srl
Str. per Castelnuovo, snc
15057 TORTONA (AL)

seguici anche su /mdsrl.it

WWW.MDSRL.IT

Plug In

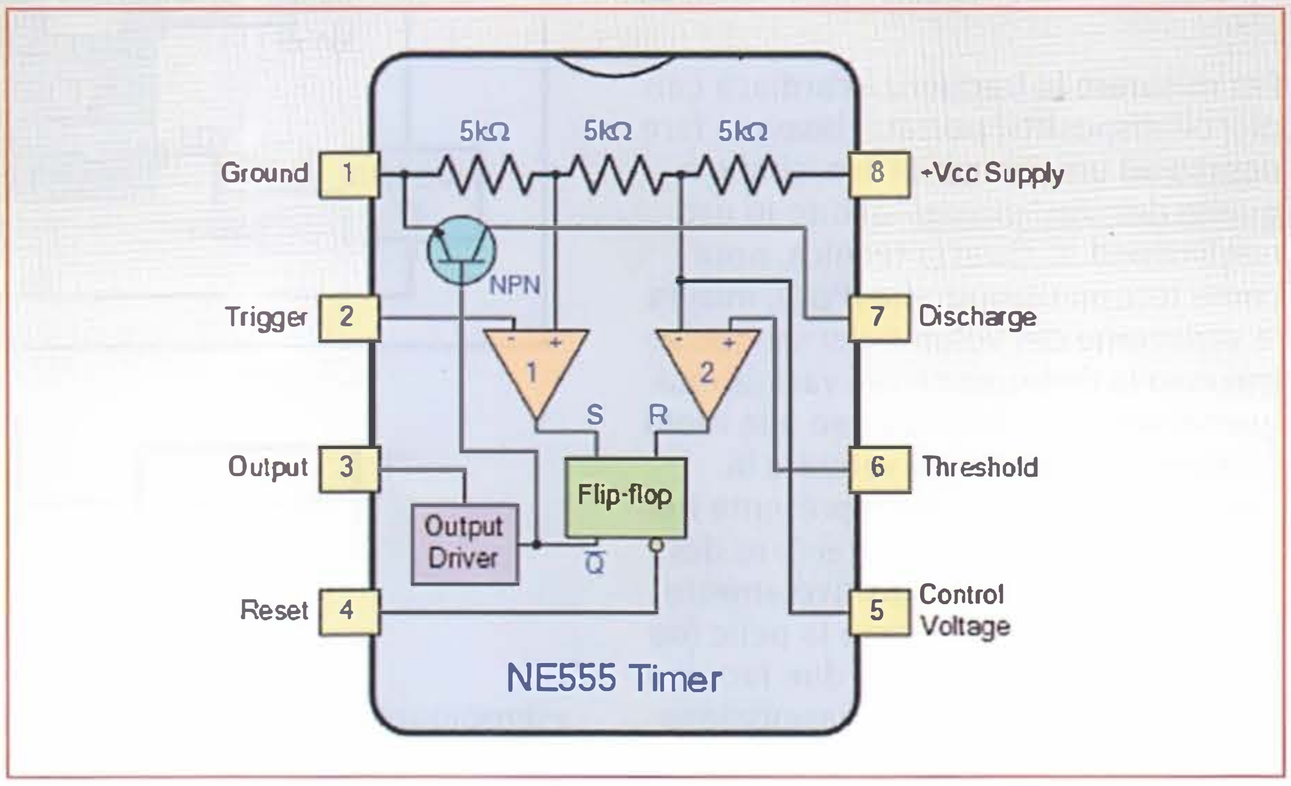
Queste pagine sono dedicate alle richieste, ai suggerimenti ed alle segnalazioni dei lettori. Raccomandiamo, per quanto possibile, di proporre argomenti di interesse generale. Contattateci numerosi all'indirizzo: *redazione@elettronicain.it*

Gli integrati più longevi

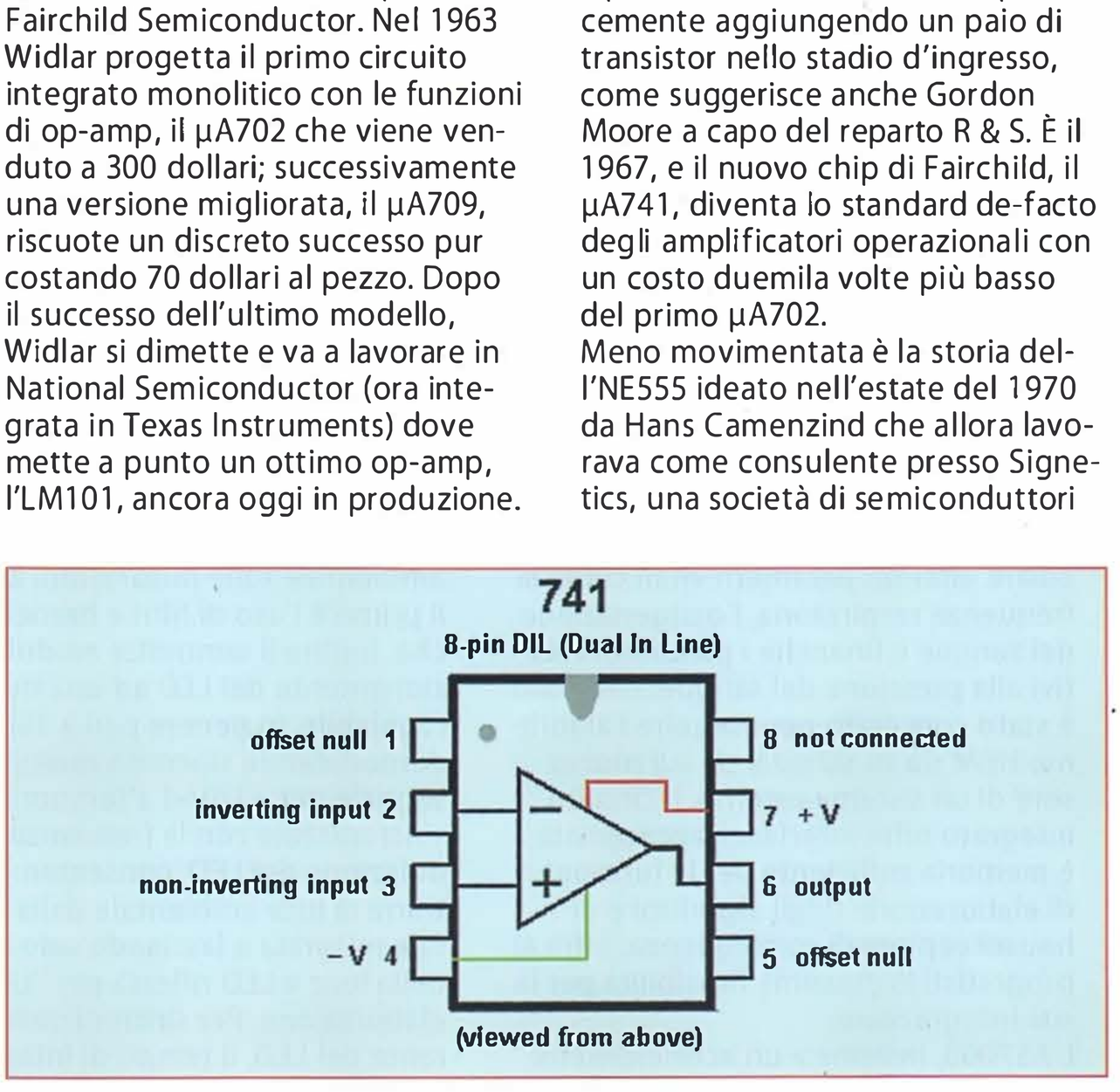
Ho una curiosità: qual è l'integrato più longevo tra quelli ancora in produzione?

Luciano Metta – Bari

Ce ne sono ancora molti, e tutti con storie interessanti che hanno segnato l'evoluzione dell'elettronica. IEEE Spectrum nella sua "Chip Hall of Fame" ne elenca una trentina, dallo Z80 all'ICL8038, solo per citarne alcuni.
Tra quelli più vicini al nostro "mondo" i due più famosi sono l'Op-Amp Fairchild µA741 e il timer/oscillatore Signetics NE555.
La nascita del µA741 è preceduta da altri modelli simili con prestazioni decisamente inferiori e costi improponibili, tutti frutto delle capacità di Robert Widlar, un giovane ingegnere di 26 anni che lavorava presso Fairchild Semiconductor. Nel 1963 Widlar progetta il primo circuito integrato monolitico con le funzioni di op-amp, il µA702 che viene venduto a 300 dollari; successivamente una versione migliorata, il µA709, riscuote un discreto successo pur costando 70 dollari al pezzo. Dopo il successo dell'ultimo modello, Widlar si dimette e va a lavorare in National Semiconductor (ora integrata in Texas Instruments) dove mette a punto un ottimo op-amp, l'LM101, ancora oggi in produzione.



I dirigenti di Fairchild corrono ai ripari assumendo un altro giovane brillante ingegnere, David Fullagar, che si rende conto che l'LM101, pur garantendo buone prestazioni presenta alcuni problemi al front-end, una eccessiva sensibilità al rumore. Il problema viene risolto semplicemente aggiungendo un paio di transistor nello stadio d'ingresso, come suggerisce anche Gordon Moore a capo del reparto R & S. È il 1967, e il nuovo chip di Fairchild, il µA741, diventa lo standard de-facto degli amplificatori operazionali con un costo duemila volte più basso del primo µA702.
Meno movimentata è la storia dell'NE555 ideato nell'estate del 1970 da Hans Camenzind che allora lavorava come consulente presso Signetics, una società di semiconduttori della Silicon Valley.
L'idea del 555 gli è venuta mentre stava lavorando ad un circuito phase-locked loop. Con alcune modifiche, il circuito avrebbe potuto funzionare come un semplice timer. E così fece, realizzando il più semplice timer/oscillatore della storia.
In un primo momento, l'idea venne respinta: Signetics aveva altri circuiti che potevano essere utilizzati come timer. Alla fine, con l'aiuto di Art Fury, responsabile marketing di Signetics, il circuito entrò in produzione: la versione ottimizzata aveva 23 transistor, 16 resistori e 2 diodi.
Quando, nel 1971, il 555 venne immesso sul mercato, il successo fu travolgente. Nel 1975 Signetics venne assorbita da Philips Semiconductors, ora NXP: nel corso degli anni le tre società hanno venduto molti miliardi di pezzi di questo piccolo dispositivo.
Ancora oggi molto popolare, possiamo trovare questo chip in quasi tutte le apparecchiature elettroniche, dagli elettrodomestici ai giocattoli, dalle schede industriali agli orologi.

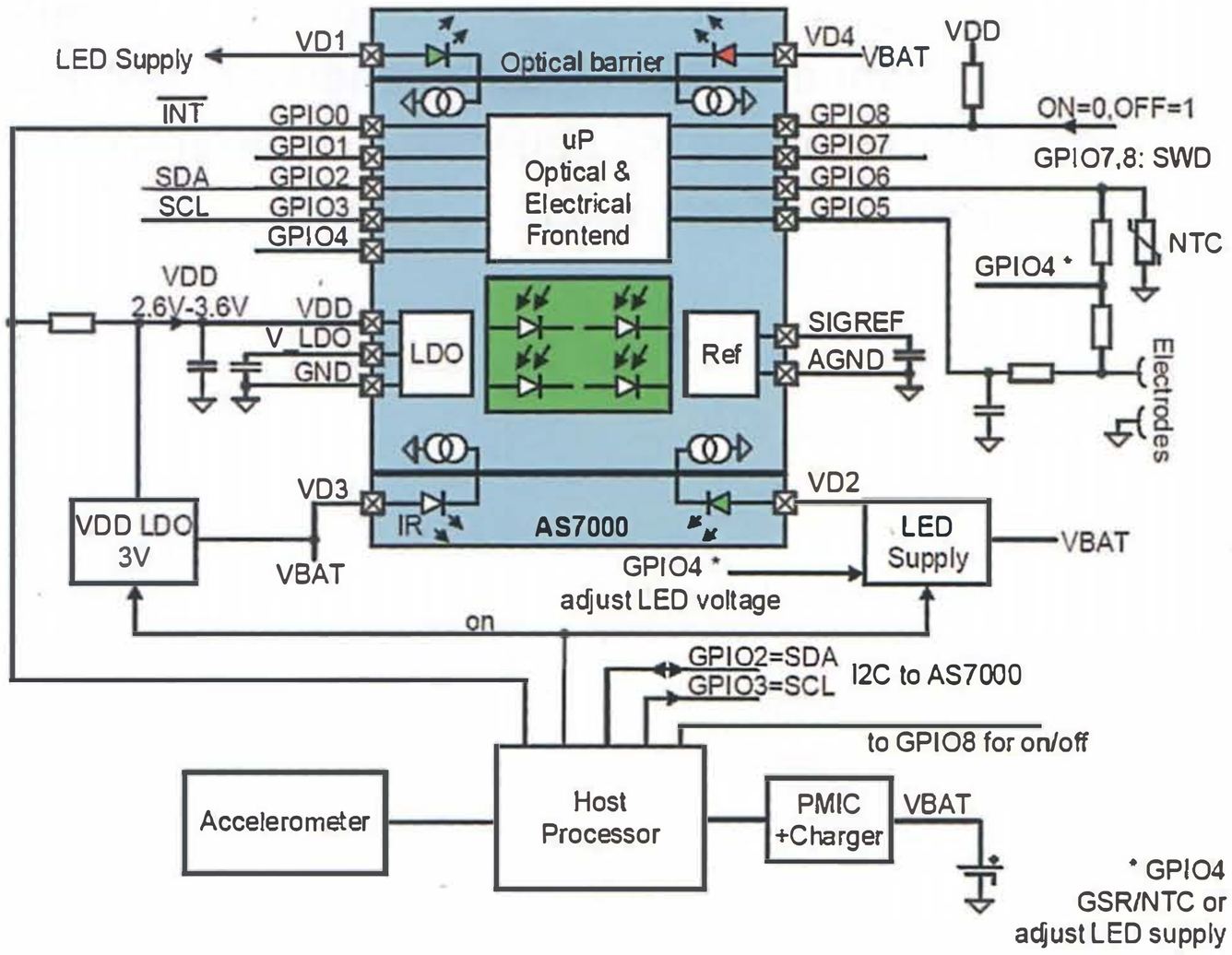




Misuriamo la frequenza cardiaca

Sto realizzando un dispositivo portatile che deve implementare nel braccialetto un sistema di misura della frequenza cardiaca. Quale soluzione mi consigliate?

Giuseppe Caria - Palermo



Per misurare la frequenza cardiaca con piccoli dispositivi portatili bisogna fare ricorso ad un sistema ottico, simile a quello dei clip "pinzati" al dito in uso negli ospedali. Questa tecnica, nota come fotopletismografia (Ppg), misura la variazione del volume del sangue attraverso la distensione dei vasi sanguigni nel tessuto sottocutaneo e in molti casi consente anche di valutare la saturazione dell'ossigeno presente nel sangue (pulsossimetria). Nel caso del clip ospedaliero, il dispositivo emette un fascio di luce attraverso la pelle (da un LED posto su una delle due facce) e misura le variazioni nella trasmissione della luce all'interno del dito (tramite un fotodiodo posto sull'altra faccia del dispositivo). La quantità di luce rilevata in prossimità del fotosensore si abbassa bruscamente e brevemente a ogni impulso, dato che l'aumento volumetrico del sangue assorbe una maggiore quantità di luce. Il segnale proveniente dal fotodiodo è un'onda simile a un dente di sega, la cui frequenza fondamentale corrisponde alla frequenza cardiaca. Per uso medico, il Ppg solitamente funziona sfruttando la trasmissione. Ma questa modalità comporta dei vincoli per la struttura meccanica: il dispositivo deve coprire una parte del corpo, in genere un dito o un lobo dell'orecchio. Si tratta di una soluzione che risulta scomoda da indossare già dopo pochi minuti. Fortunatamente, la Ppg funziona anche sfruttando la riflessione: la quantità di luce riflessa in direzione della sorgente luminosa a LED è ridotta dall'effetto di assorbimento a ogni pulsazione.
Ciò consente di implementare un misuratore di Hrv, Hrm e pulsossimetria attraverso l'epidermide nei dispositivi indossabili, come ad esempio un braccialetto. Un modulo integrato contenente tutti i blocchi funzionali richiesti, ottimizzati per il Ppg e con software dedicato, consente di risolvere qualsiasi problema di implementazione. Tra i dispositivi in grado di rispondere a questa esigenza c'è l'AS7000, un system-on-chip di rilevamento prodotto da *ams* (*Austrian Micro Systems*), una soluzione integrata per il Ppg su braccialetto in un singolo modulo da 4 x 6 x 1 mm. Include LED, fotorivelatori, circuiti per l'elaborazione ottica e un front end analogico. Inoltre è dotato di un core Arm Cortex-M0 a bassa potenza che elabora gli algoritmi di *ams* per calcolare e fornire il dato reale della frequenza cardiaca (in battiti/minuto), nonché per misurare gli intervalli tra picchi (in millisecondi) per le applicazioni basate sulla variabilità della frequenza cardiaca. Il front-end analogico estremamente versatile e flessibile dell'AS7000 può anche essere programmato per ottimizzare la qualità del segnale Ppg, in modo da poter calcolare ulteriori parametri vitali come la frequenza respiratoria, l'ossigenazione del sangue e finanche i parametri relativi alla pressione del sangue. L'AS7000 è stato concepito per eseguire l'algoritmo HRM sia su scheda, sia sul processore di un sistema esterno. Il circuito integrato offre interfacce appropriate e memoria sufficiente per le funzioni di elaborazione degli algoritmi e di housekeeping: di conseguenza, offre ai progettisti la massima flessibilità per la sua integrazione.
L'AS7000, insieme a un accelerometro dedicato alla compensazione delle interferenze da movimento, ad ogni modo può anche utilizzare i dati acquisiti da un accelerometro e condivisi dal resto del sistema, a condizione che i dati possano essere comunicati a intervalli esatti. La struttura meccanica del sistema ottico ha un impatto notevole sulle sue prestazioni. Questa è la ragione per cui il chip AS7000 è fornito come parte di una soluzione completa che integra gli algoritmi per la misurazione di Hrm, Hrv e pulsossimetria ottimizzati per le caratteristiche hardware del modulo, nonché l'esclusivo supporto per l'integrazione opto-meccanica offerto da *ams*.
Questa soluzione può essere integrata nel prodotto direttamente in fase di progettazione, senza alcun ulteriore sforzo per l'integrazione o lo sviluppo di un software dedicato. Può anche essere utilizzato solo come un front-end, controllando il processo di misurazione e fornendo dati ottici non ottimizzati al processore host. *ams* utilizza due metodi per eliminare l'impatto della luce ambientale sulle misurazioni acquisite. Il primo è l'uso di filtri e barriere ottiche. Inoltre il controller modula il funzionamento dei LED ad una frequenza regolabile, in genere pari a 1kHz. Un demodulatore sincrono moltiplica il segnale per +1/0/-1 a temporizzazioni sincronizzate con la frequenza di modulazione dei LED, consentendo di sottrarre la luce ambientale dalla luce totale misurata e lasciando solo il valore della luce a LED riflessa per l'ulteriore elaborazione. Per sincronizzare la corrente del LED, il tempo di integrazione e il tempo di campionamento del con-



vertitore analogico-digitale, può essere utilizzato un sequenziatore integrato. Il sequenziatore genera temporizzazioni a 16 bit basate su un clock con risoluzione da 1 μs e sincronizza la modulazione e la demodulazione dei LED con la conversione analogico-digitale. La circuiteria analogica dell'AS7000 fornisce un flusso di misurazioni filtrate e amplificate della luce riflessa. L'algoritmo proprietario di *ams* produce una lettura precisa e affidabile della frequenza cardiaca con sole 20 acquisizioni al secondo. I dati ricevuti dall'accelerometro sono processati dall'algoritmo a intervalli regolari, in modo da poter rilevare e compensare le interferenze da movimento. Il firmware può essere aggiornato tramite una porta seriale per il debug o interfacce I2C.
Questo permette di utilizzare algoritmi specifici per diverse applicazioni, continuando in ogni caso a sfruttare le capacità di rilevazione ottica dell'AS7000.
Il vantaggio principale dell'utilizzo del system-on-chip AS7000 è la misurazione affidabile dei parametri vitali partendo dalle misurazioni Ppg della circolazione sanguigna. Questi risultati si possono ottenere anche con l'impiego di LED discreti, fotodiodi e componenti della catena del segnale, ma ciò richiederebbe un tempo di progettazione e dei rischi più elevati rispetto all'adozione di una soluzione system-on-chip compatta, a basso consumo e già collaudata.

Lo standard Bluteooth ora è anche Mesh

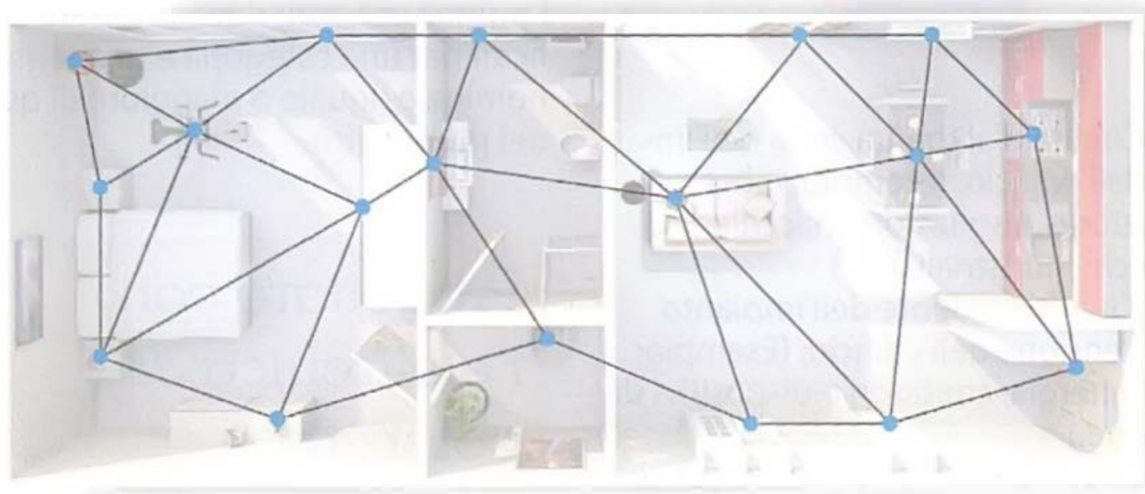
Ho letto che lo standard wireless Bluetooth è ora in grado di gestire reti Mesh. Volevo sapere se esistono già dei dispositivi commerciali in grado di supportare questa funzionalità.

Gianni Tosi – Varese

In effetti da pochissimo (luglio di quest'anno) il SIG (Bluetooth Special Interest Group), l'associazione che definisce le caratteristiche e l'evoluzione dello standard radio Bluetooth, dopo aver recentemente annunciato la versione 5.0, ha rilasciato le caratteristiche della versione Bluetooth Mesh 1.0, aprendo le porte a nuove e interessanti applicazioni che solo una rete di dispositivi, qual è la rete mesh, riesce a soddisfare.
Le nuove funzionalità di rete consentono molteplici comunicazioni tra periferiche (m:m) e sono ottimizzate per la creazione di reti di dispositivi su larga scala. Questa soluzione è l'ideale per l'automazione di edifici, per realizzare reti di sensori e soluzioni Internet of Things in cui decine, centinaia o migliaia di dispositivi comunicano tra loro in modo affidabile e sicuro.
Con questa nuova iniziativa l'associazione Bluetooth SIG conferma la propria vocazione all'innovazione più spinta e al rigore: lo standard Bluetooth, oltre che per le caratteristiche tecniche, si è affermato su scala planetaria anche per la serietà dei processi di qualificazione dei prodotti che garantiscono un'interoperabilità globale. L'introduzione della tecnologia Bluetooth Low Energy ha consentito lo sviluppo di numerose nuove applicazioni che la recente versione 5.0 dello standard promette di incrementare.
La disponibilità della tecnologia Mesh Networking apre ora nuovi scenari, specie nell'ambito dell'automazione degli edifici e del dispiegamento di reti di sensori wireless in differenti ambiti, industriale, civile, ospedaliero.
L'automazione degli edifici e delle fabbriche rappresenta la più interessante opportunità di mercato per la tecnologie mesh wireless. Questi mercati richiedono vere e proprie soluzioni industriali, che la tecnologia Bluetooth Mesh è sicuramente in grado di garantire per l'affidabilità, la scalabilità e la sicurezza del sistema.
Le specifiche della rete Bluetooth Mesh, nonché gli strumenti necessari per qualificare i prodotti Bluetooth con supporto mesh networking, sono già disponibili sul sito Bluetooth (*bluetooth.com/mesh*).
La tecnologia Bluetooth Mesh funziona con Bluetooth Low Energy (LE), è compatibile con le specifiche di base della versione 4.0 o superiore e, come era logico aspettarsi, perde il supporto audio. In compenso consentirà di implementare nuovi servizi a valore aggiunto come la localizzazione degli oggetti, l'orientamento, e forme più avanzate di beacon.
L'implementazione della tecnologia Mesh nei chip versione 4.0 o 5.0 non richiede alcuna modifica hardware ma un semplice aggiornamento del software. Per questo motivo molte case come Toshiba, Nordic, Silicon Labs ha già annunciato che i loro dispositivi sono già compatibili con la nuova tecnologia.
Da segnalare, infine, che con il supporto a Bluetooth Mesh combinato con LE, i prodotti Bluetooth offrono una soluzione completa con una distanza operativa decisamente migliore rispetto alle connessioni Bluetooth precedenti, unicamente di tipo peer-to-peer.







La Categoria di Sicurezza degli strumenti

Leggo sul mio multimetro CAT III – 600V. Cosa significa esattamente?

Giovanni Pizza – Milano

Questa scritta indica la categoria di sicurezza alla quale può operare il tuo strumento, così come definito dalla norma EN 61010.
CAT II: Circuiti direttamente collegati all'impianto a bassa tensione. (Esempio: alimentazione di apparecchi elettrodomestici e attrezzatura portatile.)



CAT III: Circuiti d'alimentazione nell'impianto dell'edificio. (Esempio: tabella di distribuzione, disgiuntori, macchine o apparecchi industriali fissi.)
CAT IV: Circuiti sorgente dell'impianto a bassa tensione dell'edificio. (Esempio: arrivo d'energia, contatori e dispositivi di protezione.)
In pratica un numero di CAT (categoria) più elevato indica un ambiente elettrico con maggiore potenza disponibile e transitori ad elevata energia. Pertanto, un multimetro conforme a uno standard CAT III è in grado di resistere a transitori a energia più elevata rispetto a uno progettato sulla base di standard di CAT II. All'interno di una categoria, un valore di tensione nominale più elevato denota un valore limite delle tensioni transitorie maggiori.
È bene accertarsi che lo strumento utilizzato soddisfi la categoria di sicurezza corretta e rispetti i valori di tensione approvati per l'ambiente in cui deve operare.
I puntali utilizzati devono essere certificati per una categoria e una tensione nominale uguale o maggiore di quella del multimetro.



Progettare con l'elettronica di potenza

Mi capita sempre più spesso di dover progettare dei circuiti di potenza (alimentatori, convertitori, ecc.) ma la mia esperienza nel settore è piuttosto scarsa. Esiste qualche tool on-line che mi possa aiutare in questo?

Mario Cera – Genova

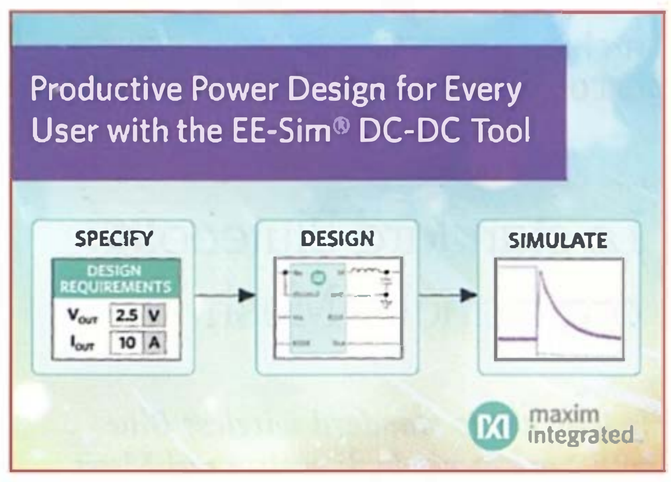
Molti Produttori di semiconduttori (ma anche alcuni distributori globali) propongono degli strumenti on-line adatti allo scopo. Naturalmente ognuno... tira l'acqua al proprio mulino e i componenti proposti da questi tool fanno solitamente parte del portafoglio di quella specifica Casa.
Ultima in ordine di tempo a proporre qualcosa che potrebbe fare per te è Maxim Integrated con la sua nuova versione di EE-Sim, lo strumento gratuito che consente ai progettisti con scarsa dimestichezza con l'elettronica di potenza di progettare e simulare convertitori DC-DC in pochi minuti.
Questo strumento offre funzioni molto gradite anche agli esperti, come la possibilità di modificare facilmente i progetti, il confronto delle prestazioni tra progetti diversi e varie possibilità di personalizzazione della simulazione; l'intuitiva interfaccia è ovviamente gradita sia ai principianti che agli esperti.
Questo il link per accedere al tool: *bit.ly/EE-Sim* .
Segnaliamo anche la risorsa storica di Texas Instruments, ovvero il WEBENCH Power Architect al quale puoi accedere dal sito di TI (*www.ti.com*).

Da I2C a I3C

Ho letto che, dopo 35 anni di onorato servizio, lo standard di comunicazione I2C è pronto ad andare in pensione, sostituito dal nuovo I3C. È così?

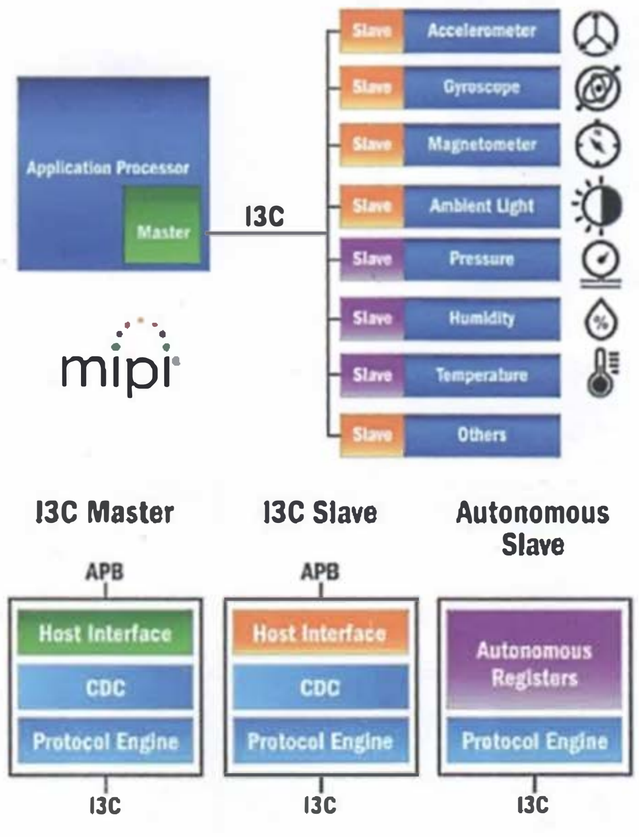
Mario Bianchi – Roma

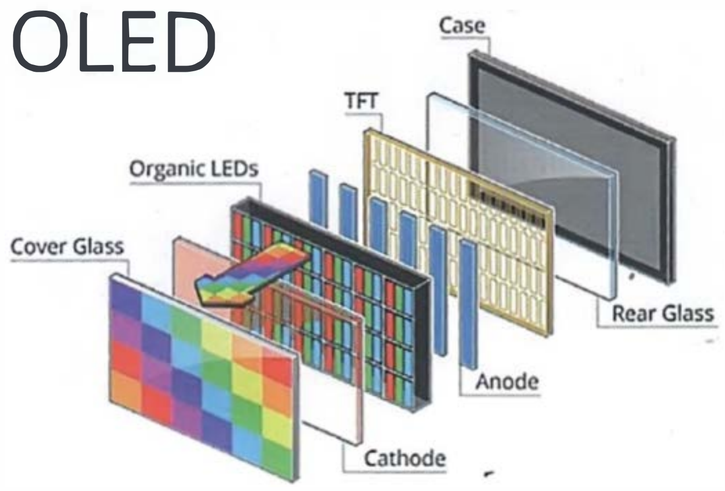
Se uno standard, un componente o un protocollo hanno resistito per tanti anni come l'I2C (oltretutto essendo soggetto a royalties) significa che i vantaggi garantiti sul campo sono sicuramente superiori rispetto ad eventuali piccole limitazioni, e che passeranno





molti anni prima che il nuovo standard (parliamo del MIPI I3C rilasciato di recente) sostituisca il sistema precedente. Sistema precedente (I²C o I2C) che è stato rilasciato per la prima volta nel 1982 da Philips (ora NXP) come un semplice bus seriale per MCU per comunicare con i pin di I/O. Teoricamente l'I2C può collegare fino a 127 dispositivi utilizzando solo due fili e la massa, anche se in pratica non si va mai oltre la decina. Il funzionamento è semplice e flessibile e consente di impegnare un numero ridotto di pin. Ad oggi l'I2C viene utilizzato un po' ovunque, soprattutto per comunicare con i sensori. Il più recente standard I3C mantiene la compatibilità con l'I2C e ne corregge alcuni difetti, oltre a garantire maggior efficienza energetica, velocità superiore e migliori meccanismi di connessione ai sensori.
Ad esempio, con I2C, un chip potrebbe bloccare di fatto il master occupando con lunghe sessioni il bus.
Questo non può accadere con I3C, perché, pur continuando ad utilizzare due fili, il nuovo protocollo prevede degli interrupt. Così, quando un sensore con dati critici deve comunicare con il master, non dovrà più aspettare che il bus si liberi. Le funzionalità avanzate di gestione energetica dello standard I3C estendono la durata della batteria mentre per quanto riguarda la velocità di trasmissione è possibile settare la velocità senza modificare il clock che può raggiungere i 12,5 MHz mentre l'I2C funziona a 400kHz, 1MHz o 5MHz. I3C ha funzionalità multi-master e supporta l'indirizzamento dinamico così come l'indirizzamento statico dei dispositivi I2C. I3C supporta anche il collegamento a caldo. Maggiori informazioni sono disponibili sul sito *mipi.org* .





LED o OLED?

Sono indeciso tra l'acquisto di un TV con tecnologia OLED e un modello a LED. La qualità dei colori dell'OLED è indubbiamente superiore ma il prezzo è quasi il doppio…
Luca Crivelli - Bologna

Pur avendo un nome simile, le due tecnologie sono molto diverse tra loro. I TV OLED (Organic Light-Emitting Diode) sono noti per offrire display più luminosi e decisi e perché utilizzano meno potenza dei LED convenzionali o dei display a cristalli liquidi (LCD). I display LCD/LED utilizzano la retroilluminazione per illuminare in una volta sola tutti i pixel e i filtri per produrre il colore. I TV OLED illuminano ogni singolo pixel, offrendo molti vantaggi alla qualità dell'immagine. Gli LCD funzionano bloccando selettivamente aree della retroilluminazione per creare le immagini che vediamo, mentre i pixel OLED generano loro stessi la luce. Come risultato, offrono un angolo di visualizzazione molto più ampio rispetto agli LCD, che funzionano in base al principio di bloccare la retroilluminazione e creare un ostacolo di visualizzazione da certi angoli. Inoltre, offrono livelli di nero più profondi con questa abilità di disattivare i pixel e di offrire un nero veramente perfetto. Il fatto che i display OLED non hanno bisogno di retroilluminazione li rende più leggeri e flessibili dei display a cristalli liquidi. Ciò significa che consumano molta meno potenza. Dato che gli strati organici degli OLED sono molto più sottili dei corrispettivi a cristalli LED, possono essere utilizzati più strati, il che li rende molto più luminosi dei LED. Se questi vantaggi non bastano, gli OLED offrono tipicamente una gamma di temperatura più ampia e un tempo di risposta più rapida. Proprio come i LED, il colore della luce emessa da un OLED dipende dal tipo di molecola organica dello strato emissivo. I produttori applicano diversi tipi di pellicole organiche sullo stesso OLED per creare i display a colore. L'intensità della luce dipende dalla corrente elettrica applicata, maggiore la corrente, più brillante è la luce. Gli strati organici di un OLED sono composti da polimeri o piccole molecole. La prima generazione utilizzava molecole organiche sottili che richiedevano un costoso e impegnativo processo di produzione. Oggi vengono utilizzate molecole polimeriche molto più larghe che possono essere sviluppate a costi inferiori e in grandi fogli, particolarmente adatti per i televisori di grandi dimensioni. I display OLED possono essere alimentati con schemi di controllo a matrice passiva (PMOLED) o a matrice attiva (AMOLED). In un progetto a matrice passiva, ogni riga del display viene controllata in sequenza, mentre gli schemi a matrice attiva utilizzano un backplane transistor a pellicola sottile per accedere e cambiare indipendentemente ogni pixel, favorendo una maggiore risoluzione e dimensioni più grandi del display. Oltre a PMOLED e AMOLED, esiste una varietà di sottotipi di OLED quali trasparenti, emittenti top, bianchi e pieghevoli. Ciascun tipo è stato progettato per offrire vantaggi per specifiche applicazioni del display. In conclusione, un TV OLED, pur presentando anche lui piccoli difetti, presenta una qualità indubbiamente superiore rispetto ai TV a LED. Resta il problema della differenza di prezzo, ma qui tutto dipende dalle tue disponibilità e dalle tue priorità.

Web Forum e supporto tecnico

Hai un problema con uno dei circuiti pubblicati? Vorresti effettuare una modifica? Accedi al nostro Web Forum dove i nostri tecnici (ma anche gli altri lettori) ti aiuteranno a chiarire qualsiasi dubbio di natura tecnica.
Collegati a:

www.elettronicain.it/webforum

Elettronica In
TI GUIDA NEL FUTURO DELL'ELETTRONICA!

Mese dopo mese realizza i progetti descritti sulla rivista, rimani aggiornato con i nostri Corsi, approfondisci la conoscenza delle tecniche e dei componenti più avanzati con i nostri tutorial e le nostre news.



COME ABBONARSI

① on-line

compilando il modulo riportato nella pagina "**Abbonamenti**" del nostro sito *www.elettronicain.it.*

② mail

scrivendo a **abbonati@elettronicain.it** con i dati riportati nel coupon a lato.

③ posta

compilando il modulo di abbonamento riportato a lato e inviandolo al seguente indirizzo:

FUTURA GROUP srl
Via Adige 11
21013 Gallarate (VA)

④ telefono

telefonando a
+39-0331-752668

⑤ fax

compilando il modulo di abbonamento riportato a lato e inviandolo al seguente numero:
+39-0331-792287



PER I RINNOVI ATTENDERE NOSTRO AVVISO

UNO STRUMENTO INDISPENSABILE PER I TUOI STUDI, IL TUO LAVORO O LA TUA PASSIONE CHE PUOI RICEVERE COMODAMENTE A CASA TUA OGNI MESE SOTTOSCRIVENDO UN...

ABBONAMENTO ANNUALE

10 NUMERI a soli euro 45,00 anziché euro ~~60,00~~



29 Ottobre 1969: nasceva la rete. Da uno stanzino dell'Università della California di Los Angeles veniva inviato il primo messaggio sulla rete Arpanet.

IN OMAGGIO PER TE:

10% di sconto SUI TUOI ACQUISTI FUTURA ELETTRONICA®

effettuati con la **DISCOUNT CARD** presso il punto vendita di Gallarate (VA) o nel nostro store on-line *www.futurashop.it*

+

1 VOLUME A SCELTA della collana "L'Elettronica per tutti".

SÌ desidero abbonarmi per un anno alla rivista *Elettronica In*. Riceverò i seguenti omaggi:

- [x] **DISCOUNT CARD "Futura Elettronica"**
- [ ] **UNO, A SCELTA, TRA QUESTI VOLUMI** della collana "L'Elettronica per tutti".



- [ ] Tecnologia e apparati GPS
- [ ] Impara l'elettronica sperimentando
- [ ] Primi passi in elettronica
- [ ] Arduino e le tecniche di programmazione dei microcontrollori ATMEL
- [ ] Raspberry Pi - il mio primo Linux Embedded
- [ ] Alla scoperta di Arduino YÚN
- [ ] Programmiamo con i PIC
- [ ] l'ABC di Arduino
- [ ] Stampiamo in 3D

Resto in attesa del primo numero e degli omaggi.

Cognome:

Nome:

Via:

N°: CAP:

Città:

Provincia: Telefono:

e-mail:

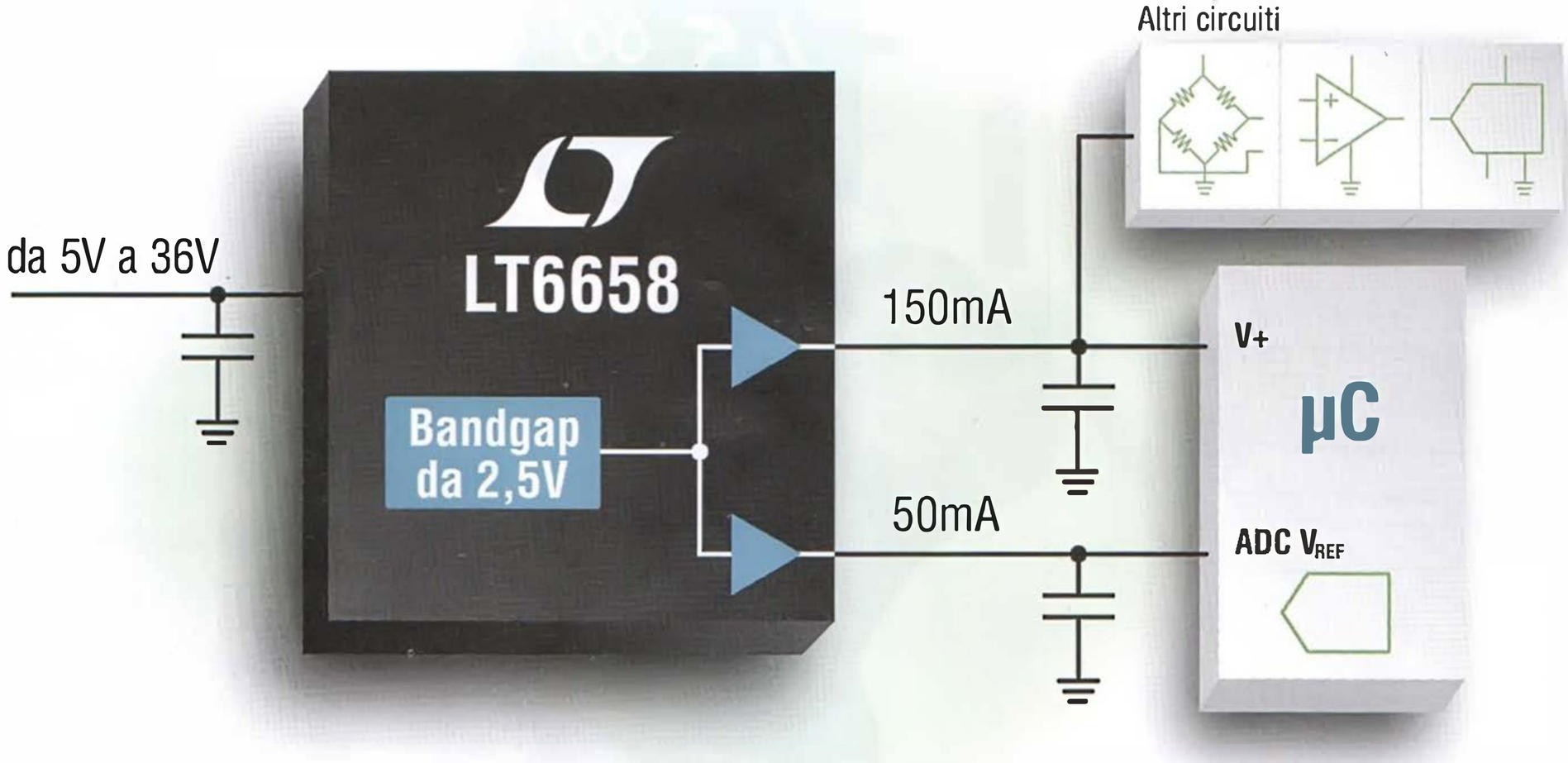
Data: Firma:

Resto in attesa di vostre disposizioni per il pagamento.
Ai sensi dell'art. 13 del D.Lgs. 196/2003, in tema di protezione dei dati personali, FUTURA GROUP srl informa che i dati liberamente forniti attraverso la compilazione del presente modulo saranno trattati e conservati in conformità a tutte le normative vigenti. I dati suddetti saranno trattati su supporto cartaceo e informatico, mediante sistemi di protezione atti alla tutela della riservatezza, per finalità connesse all'esecuzione dei rapporti contrattuali instaurati.
Inviare in busta chiusa a:

FUTURA GROUP srl
Via Adige 11 • 21013 Gallarate (VA)
Tel. +39-0331-752668 • Fax: +39-0331-792287

"Refulator" da 200mA

Deriva max 10ppm/°C • Precisione ±0,05% • Rumore 1,5ppm$_{P-P}$

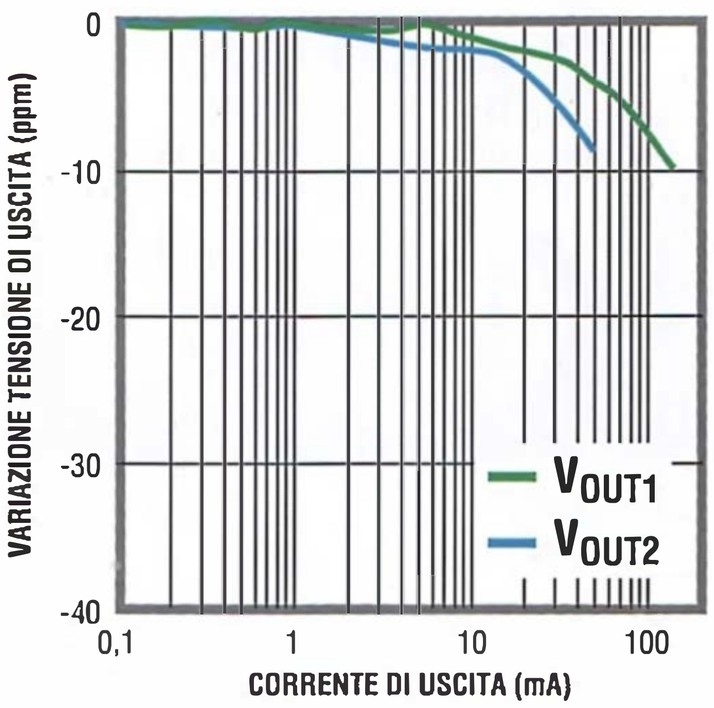


Il LT6658 aumenta la capacità dei riferimenti di tensione di precisione integrando due buffer di uscita in grado di assorbire 20mA e di fornire rispettivamente fino a 150mA e 50mA. Sono disponibili sei opzioni con tensioni di riferimento interne differenti e ogni tensione di uscita può essere regolata usando resistori esterni. Dotato di due uscite, l'LT6658 può fungere sia da riferimento che da regolatore, oppure da riferimento più messa a terra virtuale come alimentazione raziometrica con dual tracking, oppure ancora, con le uscite combinate, da riferimento di tensione singolo da 200mA.

Caratteristiche

- Riferimento di tracking a due uscite:
  - Uscita 1: 150mA sink / 20mA source
  - Uscita 2: 50mA sink / 20mA source
- Basso drift:
  - Classe A: 10ppm/°C max
  - Classe B: 20ppm/°C max
- Precisione elevata:
  - Classe A: ±0,05% max
  - Classe B: ±0,1% max
- Rumore ridotto: 1,5ppm$_{P-P}$ (da 0,1Hz a 10Hz)
- Regolazione del carico: 0,1ppm/mA
- Tensioni disponibili: 1,2V, 1,8V, 2,5V, 3V, 3,3V e 5V

Regolazione del carico



Info e acquisto online

www.linear.com/product/LT6658

Tel.: +39-039-596 50 80

Fax: +39-039-596 50 90



video.linear.com/6035

, LT, LTC, LTM, Linear Technology e il logo Linear sono marchi registrati di Analog Devices, Inc. Tutti gli altri marchi sono di proprietà dei rispettivi titolari.

LINEAR TECHNOLOGY | NOW PART OF ANALOG DEVICES

Cerca il tuo ufficio commerciale locale: **www.linear.com/contact**

È tempo di NIDays

Per la prima volta NIDays si svolgerà in autunno, precisamente il 14 novembre 2017 presso l'Atahotel Expo Fiera di Pero, alle porte di Milano. L'evento annuale di National Instruments dedicato a innovatori, tecnici, ingegneri, professori e ricercatori conferma invece la struttura in un'unica giornata, con due keynote con speaker d'eccezione, un'area espositiva ricca di soluzioni e applicazioni e numerose sessioni tecniche e pratiche.
Le due keynote sveleranno le nuove tendenze e le best practice per affrontare le nuove sfide dell'ingegneria: "*Testare e realizzare la futura generazione tecnologica*" al mattino, e "*La piattaforma NI: costruisci ora le soluzioni future*", nel pomeriggio.
Saranno più di 30 le sessioni tecniche disponibili ed anche per questa edizione non mancheranno le aree pratiche dove provare con mano LabVIEW e le piattaforme hardware NI.
L'area espositiva comprende due piani interamente dedicati ai prodotti e alle tecnologie dei partner tecnologici e degli Alliance Partner di NI.
Da oltre 40 anni National Instruments fornisce soluzioni innovative per la misura, il controllo e i sistemi di test. Durante una sola giornata i visitatori avranno la possibilità di scoprire i trend, i prodotti e le soluzioni per affrontare le sfide dell'ingegneria e fare networking.
In particolare quest'anno il focus sarà su NI LabVIEW, l'ambiente di programmazione grafico di sistemi più famoso al mondo che dopo 30 anni si presenta in due versioni, con LabVIEW NXG completamente nuova e con tante interessanti novità, da un'integrazione hardware immediata alla realizzazione di user-interface moderne e intuitive.
Durante NIDays sarà anche possibile sostenere gratuitamente l'esame di certificazione CLAD (LabVIEW Certified Associate Developer).
L'esame di certificazione NI Certified LabVIEW Associate Developer (CLAD) costituisce il primo dei tre livelli di certificazione disponibili per l'uso di LabVIEW. La certificazione attesta la conoscenza del software e fornisce una attestazione importante ai fini di future opportunità di carriera e di nuovi successi professionali. Il superamento dell'esame CLAD permette di accedere all'esame di secondo livello Certified LabVIEW Developer (CLD).
La certificazione CLD attesta la capacità di progettare e sviluppare programmi funzionali in LabVIEW, per ridurre i tempi di sviluppo e assicurare la mantenibilità del codice attraverso idonea documentazione.
www.ni.com/it



IoT Day



L'Internet of Things, con la crescita di devices connessi alla rete per la raccolta e la condivisione di dati, rappresenta una grande sfida e al contempo un'opportunità per un'ampia serie di settori, dall'Automazione Industriale, alla Home Automation, la Robotica e molti ancora. L'IoT Day, la giornata dedicata all'aggiornamento tecnico sulle ultime tecnologie e soluzioni per applicazioni IoT, nasce per dare la possibilità di conoscere le soluzioni per il mondo IoT direttamente dalla voce dei maggiori brand dell'elettronica, in una giornata dedicata all'aggiornamento tecnico teorico e di prodotto.
I temi trattati durante la giornata saranno:

- TRASMISSIONE DATI
- POWER MANAGEMENT
- SENSORI
- PIATTAFORME DI SVILUPPO
- SINGLE BOARD COMPUTERS

Oltre alla possibiltà di vedere e toccare con mano gli ultimi prodotti, sarà possibile partecipare a training tecnici tenuti nella Tech Arena da esperti del settore. Questa l'Agenda:

9.00 Registrazione
9.45 Apertura
10.00 The evolution of Raspberry PI (Raspberry PI Foundation)
10.45 Privacy&Security, Low Power, connettività e sensoristica: tutti i requisiti necessari per lo sviluppo di applicazioni IoT passano da NXP, (NXP).
11.30 Soluzioni Xilinx interamente programmabili per applicazioni IoT in campo Industriale. (Xilinx)
12.15 Soluzioni Tektronix-Keithley per le misure elettroniche per la progettazione e il collaudo di dispositivi IoT. (Tektronix-Keithley)
13.00 Buffet Lunch
14.30 La piattaforma SimpleLink di TI per la connettività avanzata in ambito <1GHz, BT e Wi-Fi. (Texas Instruments)
15.15 OMRON introduce i nuovi sensori per l'IoT. (Omron)
16.00 Chiusura

Il 26 ottobre 2017 presso l'Hotel Bologna Fiera (Piazza della Costituzione 1) a Bologna. I posti sono limitati. La partecipazione è gratuita, previa registrazione.
it.farnell.com/iotday_bologna_17

ON Semiconductor Power Seminars

ON Semiconductor®

Il 12 ottobre, unica tappa italiana, arriva a Padova l'ON Semiconductor Power Seminars 2017, un evento gratuito che fornisce una visione unica sul mondo dell'elettronica di potenza, con soluzioni all'avanguardia e informazioni sulle nuove tendenze tecnologiche.
Un seminario adatto a quanti si occupano di progettazione e sviluppo di nuove soluzioni di potenza in ambito industriale, tenuto dai principali esperti del settore, con la "densità di potenza" che sarà uno degli argomenti chiave di tutte le presentazioni. Il raggiungimento di elevati livelli di densità di potenza implica infatti l'utilizzo degli ultimi progressi topologici nella conversione di potenza, con metodi di controllo avanzati e con le migliori tecnologie a semiconduttore disponibile per il progettista, principalmente gli switcher di potenza GaN che possono aiutare a raggiungere l'obiettivo finale di un'elevata efficienza, elevate prestazioni, alta affidabilità e costo contenuto.
Durante il seminario si parlerà anche di MOSFET SiC e dei relativi criteri di progettazione per massimizzare le prestazioni di commutazione dei SiC; non mancherà una panoramica sulle tecnologie IGBT, componenti utilizzati in applicazioni ad alta potenza e alta tensione.
Altre presentazioni riguarderanno le applicazioni e le attuali tendenze di controllo motore, le tecnologie per misurare e ottimizzare un loop di controllo e le metodologie di modellazione SPICE.
Una panoramica sui più recenti dispositivi di potenza di produzione ON Semiconductor completerà l'evento.
Questa l'agenda del seminario:
08:30 a.m. Registration
09:00 a.m. IGBT Technologies & Applications
10:00 a.m. SiC MOSFETs
11:00 a.m. Break
11:20 a.m. Modern Motor Control
12:20 p.m. Lunch
01:15 p.m. Power-Components
01:45 p.m. Control Loop Design
02:45 p.m. Break
03:00 p.m. Physically Based SPICE Modeling
03:40 p.m. The Quest for Power Density
04:30 p.m. End
A Villa Ottoboni, Padova, il 12 Ottobre 2017.
www.onsemi.com



Festival della Scienza di Genova



Dal 26 ottobre al 5 novembre torna a Genova il Festival della Scienza; per questo nuovo appuntamento il paese ospite sarà il Giappone mentre la parola chiave attorno alla quale ruoterà tutto il Festival sarà #contatti.
La scienza è fatta di contatti: contatto tra teoria e sperimentazione, sinergie tra discipline, interazione tra elementi semplici che generano complessità. Ci sono contatti che uniscono e contatti che innescano competizione e conflitti. I nostri pensieri nascono dal contatto di miriadi di neuroni nel nostro cervello, e il nostro esistere si fonda su contatti tra persone, culture e idee che determinano progresso e sviluppo sociale, economico e industriale. Di questo e di molto altro si occuperà la quindicesima edizione del Festival della Scienza.
Il Festival della Scienza è uno dei più grandi eventi di diffusione della cultura scientifica a livello internazionale, un'occasione di incontro per ricercatori, appassionati, scuole e famiglie con l'edizione 2016 che ha visto la presenza di oltre 200 mila visitatori, con un significativo incremento rispetto all'anno precedente.
Incontri, laboratori, spettacoli e conferenze per raccontare la scienza in modo innovativo e coinvolgente, con eventi interattivi e trasversali.
Undici giorni in cui le barriere fra scienze matematiche, naturali e umane, verranno abbattute e la ricerca si potrà toccare, vedere, capire senza confini
www.festivalscienza.it

Linux Day 2017

Sabato 28 ottobre torna la principale manifestazione italiana dedicata a Linux, al software libero, alla cultura aperta ed alla condivisione: decine di eventi in tutta Italia, centinaia di volontari coinvolti e migliaia di visitatori per celebrare insieme la libertà digitale!
L'edizione 2017 del Linux Day è dedicata alla privacy e alla riservatezza individuale. Perché tutti abbiamo qualcosa da nascondere: i fatti nostri.

Il Linux Day è una manifestazione nazionale articolata in eventi locali che ha lo scopo di promuovere GNU/Linux e il software libero. Il Linux Day è promosso da ILS (Italian Linux Society) e organizzato localmente dai LUG italiani e dagli altri gruppi che condividono le finalità della manifestazione.
Si compone di 100 e più eventi locali, organizzati autonomamente da gruppi di appassionati nelle rispettive città, tutti nello stesso giorno. In tale contesto è possibile trovare talks, workshops, spazi per l'assistenza tecnica, gadgets, dibattiti e dimostrazioni pratiche.
L'elenco delle località dove si svolgerà il Linux Day è disponibile sul sito della manifestazione e viene aggiornato quotidianamente. La prima edizione del Linux Day si è tenuta il 1 dicembre 2001 in circa quaranta città sparse su tutto il territorio nazionale. Il numero è notevolmente cresciuto negli anni divenendo ormai un appuntamento immancabile nel panorama del software libero italiano.
www.linuxday.it

RISERVATEZZA NATURALE

28 ottobre 2017
www.linuxday.it

A Smart Mobility World tutta la mobilità innovativa

SMART MOBILITY WORLD, alla sua V edizione, rappresenta la più importante manifestazione europea dedicata alla mobilità sostenibile, digitale e integrata.
L'evento, attraverso l'innovativo format dell'EXPOFORUM, rappresenta un hub professionale per generare concrete opportunità di business, con oltre 3500 presenze di operatori internazionali, grazie a formule di partecipazione che integrano visibilità e contenuti.
Evoluzione dei veicoli, smart cities e progresso delle tecnologie sono al centro di un dialogo aperto con realtà pubbliche e private, aziende di grandi dimensioni e start-up.
È un evento professionale che incrocia le aziende di riferimento con le pubbliche amministrazioni, le associazioni di categoria e le start-up in uno spazio espositivo volto al networking.
La manifestazione, in programma a Lingotto Fiere di Torino il 10 e 11 ottobre prossimi, continua a suscitare grande interesse da parte delle aziende leader.
Smart sarà presente con la nuova famiglia electric drive, una scelta ideale per una mobilità urbana intelligente, versatile e davvero sostenibile. *smart* è, infatti, oggi l'unico marchio al mondo ad offrire sull'intera gamma la motorizzazione 100% elettrica, anche in versione Cabrio, prima e unica 'scoperta' a zero emissioni del mercato.
Smart Mobility World ospita come in ogni edizione anche diversi eventi verticali, i cosiddetti Hosted Event. Quello di più recente conferma è previsto il 10 ottobre, coordinato da ACI e dedicato al progetto europeo I-HeERO, che punta a costruire un servizio



europeo di chiamata automatica di emergenza (e-Call), disponibile sia per automobili che per veicoli commerciali e industriali. L'evento, che avrà anche una propria area espositiva dedicata ai partner ed ai "pilot" attivati nei diversi Paesi, rappresenta la tappa italiana del roadshow del progetto in corso in tutta Europa.
Un altro hosted event verrà organizzato sempre il 10 ottobre dal Politecnico di Torino. Si tratta della conferenza internazionale "Funding and policies for Smart Mobility", che presenterà opportunità di finanziamento e cooperazione per progetti di Ricerca e Innovazione in ambito Smart Mobility. Verranno illustrate le linee di finanziamento europee in Horizon2020 e le iniziative internazionali finalizzate al supporto della ricerca di soluzioni innovative per il trasporto e la mobilità. Tra gli speaker, un rappresentante della Strategy Transport Unit – DG Research and Innovation, European Commission, Angela Di Febbraro (Università di Genova, Delegato nazionale italiano nel Comitato di Programma per i Trasporti "Smart, green, and integrated transport" del Programma di Ricerca europeo Horizon 2020) e rappresentanti delle principali piattaforme europee nell'ambito della mobilità.
www.smartmobilityworld.net

Soluzioni avanzate Microchip per Gigabit Ethernet

Il portfolio GigEpack combina chip, strumenti e test di conformità



Il GigEpack di Microchip, riducendo la complessità di reti embedded ad alta velocità, apre nuove opportunità per Gigabit Ethernet in applicazioni Industriali, Automotive, e Consumer.

GigEpack offre, insieme, chip testati UNH-IOP, un completo supporto software, e schede di valutazione copy-ready, con i driver per tutti i principali sistemi operativi. Il portfolio di chip avanzati include il primo switch Ethernet fault-tolerant sul mercato ed il primo bridge Automotive-grade USB 3.1 Gen 1 -to- Gigabit Ethernet. Il LANCheck® Service può essere usato per verificare che tutte le best practice siano state applicate in ognuno dei progetti.



microchip DIRECT
www.microchipdirect.com

MICROCHIP

www.microchip.com/GigEpack

Il nome e logo Microchip, il logo Microchip, e EtherSynch sono marchi industriali registrati di Microchip Technology Incorporated negli U.S.A. e altri Stati. Tutti gli altri marchi industriali menzionati nel presente documento appartengono ai rispettivi titolari. © 2017 Microchip Technology Inc. Tutti i diritti riservati. DS00002421A. MEC2150Ita08/17

■ 10-12 OTTOBRE 2017

It-sa

Norimberga • Germania

La sicurezza in campo informatico, comprese le sempre più diffuse applicazioni di dispositivi connessi in ambito IoT e IIoT, è ormai l'esigenza primaria di qualsiasi sistema di trasmissione, elaborazione e memorizzazione dati, tanto che questo settore, ormai, rappresenta un comparto a sé stante nel gran mare dell'Information Technology.
Un settore al quale è dedicata la manifestazione it-sa di Norimberga, ormai una delle più importanti del settore, con numeri in crescita a due cifre.
Quest'anno It-sa si svolgerà dal 10 al 12 ottobre, occupando per la prima volta due padiglioni del centro espositivo di Norimberga, al fine di soddisfare l'elevata domanda dei fornitori di sicurezza tedeschi e internazionali.
Recentemente il ransom Wanna Cry ransom ha paralizzato centinaia di migliaia di computer in tutto il mondo, inclusi i sistemi utilizzati dagli ospedali nonché dalle ferrovie tedesche (Deutsche Bahn). Questo attacco ha mostrato, ancora una volta, la vulnerabilità delle attuali infrastrutture IT. It-sa illustrerà come le aziende e le organizzazioni possono proteggersi da una vasta gamma di minacce. Con la sua decisa focalizzazione sulla sicurezza, questo evento copre una posizione eccezionale nel panorama espositivo europeo. Con 489 espositori l'anno scorso, ha raggiunto i limiti di capacità del padiglione 12, dove si è tenuta dalla sua edizione inaugurale nel 2011. L'evento si trasferirà quindi nei padiglioni 9 e 10 a partire da quest'anno.

www.it-sa.de



■ 7-10 NOVEMBRE 2017

Ecomondo e Key Energy

Rimini

Dal 7 al 10 novembre prossimi alla Fiera di Rimini l'appuntamento è con ECOMONDO E KEY ENERGY, la piattaforma europea leader per l'industria green al fianco delle aziende dell'ambiente e dell'economia sostenibile. Debuttano il progetto su Trasporto e Mobilità Sostenibili, pilastri dell'Urban Circular Economy, e l'area sul dissesto idrogeologico e la prevenzione dei rischi climatici.
Riconfermati, il 7 e l'8, gli Stati Generali della Green Economy, organizzati dal Consiglio Nazionale della Green Economy, composto da 64 associazioni di imprese green, in collaborazione con i Ministeri dell'Ambiente e dello Sviluppo Economico, con il supporto tecnico della Fondazione per lo Sviluppo Sostenibile presieduta da Edo Ronchi.
Torna inoltre SAL.VE., il Salone biennale del Veicolo per l'Ecologia, in partnership con Anfia, su 6.000 mq con l'intera gamma di veicoli industriali e speciali per la raccolta dei rifiuti solidi e liquidi.
ECOMONDO E KEY ENERGY rappresentano un appuntamento europeo che ha anticipato urgenza e declinazioni della green economy e consente a tutti gli attori di essere partecipi delle sfide e opportunità della nuova era per lo sviluppo sostenibile globale, con benefici in termini occupazionali, di risparmio, di tutela ambientale. Tagliato lo scorso anno il traguardo delle venti edizioni con 105.574 visitatori (11.000 stranieri) e 1.200 imprese - ECOMONDO metterà a disposizione nel 2017 un numero ancora più ampio di metri quadri espositivi, grazie ai lavori di ampliamento del quartiere fieristico.
La kermesse riminese, appuntamento chiave per i trend dello sviluppo dell'economia green in tutti i suoi settori si concentrerà sui Paesi del bacino del Mediterraneo; mercati centrali anche Brasile e Cina.



www.ecomondo.com

MOSTRE MERCATO

PADOVA
TUTTINFIERA 2017 - TRONICA
Quartiere Fieristico
Organizzazione: Padova e Boot srl
Tel: 3286467529
ebootsrl@gmail.com
www.eboot.it
4 e 5 Novembre 2017

BASTIA UMBRA (PG)
EXPO ELETTRONICA
Quartiere Fieristico - Bastia Umbria (PG)
Organizzazione: Blu Nautilus
Tel: 054153295 - Fax: 0541439584
info@expoelettronica.it
www.expoelettronica.it
4 e 5 Novembre 2017

LIVORNO
FIERA DELL'ELETTRONICA DI LIVORNO
Pala Livorno
Organizzazione: Prometeo
Tel: 057122266 - Fax: 05710831142
info@prometeo.tv
www.prometeo.tv
4 e 5 Novembre 2017

MARIANO COMENSE (CO)
FIERAMERCATO DELL'ELETTRONICA
Palazzo storico delle Esposizioni
Organizzazione: Printmix
Tel: 0362999186 - Fax: 0362865244
info@fieramercatodellelettronica.it
www.fieramercatodell'elettronica.it
4 e 5 Novembre 2017

PORDENONE
RADIOAMATORE 2
Quartiere Espositivo - Pordenone
Organizzazione: Pordenone Fiere spa
Tel: 0434232111
Fax: 0434570415
info@fierapordenone.it
www.radioamatore2.it
18 e 19 Novembre 2017

BOLOGNA
MONDO ELETTRONICA
Bologna Fiere - Via Aldo Moro
Organizzazione: Expo Fiere srl
Tel: 054527548
Fax: 0545291179
info@expositionservice.it
www.mondoelettronica.net
18 e 19 Novembre 2017

PARMA
PARMA TRIX
Fiere di Parma -
Via delle Esposizioni 393
Organizzazione: eBoot srl
Tel: 3286467529
ebootsrl@gmail.com
www.eboot.it
25 e 26 Novembre 2017

VERONA
ELETTROEXPO
Quartiere fieristico Verona
Organizzazione: Veronafiere
Tel: 0458298111
Fax: 0458298288
info@veronafiere.it
www.elettroexpo.it
25 e 26 Novembre 2017

PESCARA
FIERA MERCATO NAZIONALE DEL RADIOAMATORE DI PESCARA
Centro IBISCO – Via Lungofino 187
Città S. Angelo (PE)
Organizzazione: ARI sezione di Pescara
Tel: 0854714835 - Fax: 0854711930
arifiera@aripescara.org
www.aripescara.org
25 e 26 Novembre 2017

FERRARA
ELETTRONICA
La Grande Fiera Campionaria
Fiera di Ferrara - Via della Fiera 11
Organizzazione: Expo Fiere srl
Tel: 054527548
Fax: 0545291179
info@mondoelettronica.net
www.mondoelettronica.net
25 e 26 Novembre 2017

L'elenco aggiornato di tutte le Mostre Mercato del 2017 è disponibile sul sito *www.elettronicain.it* sul quale è anche possibile scrivere un commento sulle fiere visitate.

L'elenco completo dei più importanti eventi nazionali e internazionali di elettronica, sicurezza e fonti rinnovabili è disponibile sul nostro sito:
http://www.elettronicain.it/myMain/Fiere/p_fiere.asp

■ 18 - 19 OTTOBRE 2017

SAVE 2017

Verona

Il 18 e 19 ottobre a Veronafiere è in programma l'undicesima edizione di SAVE, Mostra convegno dedicata alle Soluzioni e Applicazioni Verticali di Automazione, Strumentazione, Sensori, l'appuntamento verticale più atteso del settore. Dopo i brillanti risultati del 2016, SAVE torna con un calendario fitto di appuntamenti, convegni, dibattiti, tavole rotonde, approfondimenti dal taglio tecnico-applicativo che affronteranno tematiche di grande attualità. Si spazierà dalle novità dell'IoT e Industria 4.0 per il processo e per il manifatturiero alle frontiere della nuova predittiva, dai sistemi di visione all'asset management alla sicurezza informatica, dalla sensoristica alle soluzioni specifiche per l'efficienza energetica e molto altro ancora, per implementare le migliori strategie e soluzioni per i differenti mercati industriali - quali Energia, Ambiente, Food & Beverage, Pharma, Plastica, Chimica e Petrolchimica, Automotive, Gomma e Plastica e così via. I maggiori player del mercato hanno già confermato la loro presenza all'evento, tra questi i Platinum Sponsor Beckhoff, Keyence, Rittal; da anni SAVE è infatti un appuntamento fisso nell'agenda degli operatori, un momento fondamentale di aggiornamento professionale, di confronto diretto aperto a esperti, aziende, enti, istituzioni, per offrire la più completa rassegna di tecnologie e soluzioni nell'ambito della strumentazione e dell'automazione, oltre che quale punto d'incontro per sviluppare business. SAVE 2017 si svolgerà in concomitanza con MCM (Manutenzione e Asset management), ACQUARIA (Trattamento acqua e aria), VPC (Valvole, Pompe e Componentistica) e HOME & BUILDING (automazione dell'edificio e building automation).

www.exposave.com

SAVE

■ 13-16 OTTOBRE 2017

HKTDC Hong Kong Electronics Fair

Hong Kong • SAR

Sarà l'Hong Kong Convention and Exhibition Centre ad ospitare dal 13 al 16 ottobre 2017 la 37^ edizione dell'Hong Kong Electronics Fair (Autumn Edition) e la 21^ edizione di ElectronicAsia; complessivamente le due fiere riuniranno circa 4.200 espositori provenienti da tutto il mondo, uno dei più grandi mercati del settore elettronico con un'ampia gamma di prodotti smart, idee innovative e una vasta scelta di prodotti e componenti elettronici. Nelle edizioni del 2016 le due fiere hanno attirato un totale di quasi 94.000 buyer provenienti da circa 150 paesi.

L'iniziativa Hall Tech, lanciata lo scorso anno, tornerà a presentare prodotti tecnologicamente all'avanguardia in cinque aree tematiche: 3D Printing, Robotics & Unmanned Tech, Smart Tech, Startup e Virtual Reality.

Le start-up sono una forza crescente dell'industria elettronica in grado di portare nuovi concetti e prodotti innovativi. Per questo motivo in un'area dedicata, la fiera di Hong Kong ospiterà circa 100 start-up provenienti da tutto il mondo, consentendo agli investitori di esplorare opportunità di investimento e partnership. Saranno organizzati una serie di eventi a tema e sessioni di pitching in cui le start-up presenteranno soluzioni o idee di prodotto a potenziali investitori: "Startup, Smart Launch". Non mancheranno le sessioni di Mentoring dove gli imprenditori di successo condividono le loro esperienze con le start-up.

Nell'area dedicata alla tecnologia di stampa 3D sarà possibile trovare macchine con tecnologie differenti, prodotti per la stampa e servizi correlati. Robotics and Unmanned Tech presenterà gli ultimi modelli di telecamere aeree, robot industriali e domestici, scooter elettrici e tanto altro ancora. Nell'area Smart Tech verranno presentati i dispositivi e le applicazioni relativi all'Internet of Things mentre l'area Virtual Reality proporrà visori e applicazioni per la realtà virtuale.

Come consuetudine, la Hall of Fame continuerà a mettere sotto i riflettori i prodotti più raffinati proposti da più di 520 marchi di prestigio come Binatone, Goodway, Motorola e VTech. Altre novità di rilievo saranno presentate nelle aree i-World, Wearable Electronics, Healthcare Electronics, Audio Visual Products e Personal Electronics.

Parallelamente alla Fiera principale, electronicAsia offre ad una platea internazionale una piattaforma ineguagliabile per i fornitori di componenti elettronici e tecnologie di produzione.

www.hktdc.com



■ 15 - 17 NOVEMBRE 2017

Sicurezza 2017

Milano

In un momento in cui il mercato della security continua a crescere e il comparto si dimostra vitale e ricettivo sia verso le richieste interne che verso le opportunità offerte dall'internazionalizzazione, SICUREZZA - dal 15 al 17 novembre 2017 a Fiera Milano - si conferma uno strumento al servizio di aziende e operatori.

Le opportunità di sviluppo non mancano: il 2016, secondo i dati recentemente diffusi da ANIE Sicurezza, ha infatti registrato una crescita del fatturato totale del 5,1% sul 2015. Un trend anche migliore di quello dell'anno precedente (+4,5%), con un fortissimo slancio per i sistemi anti-intrusione, incrementati del 7,3%, e in particolare per la Videosorveglianza, in aumento di quasi il 10%. Pubblica Amministrazione, Terziario, Retail e GDO figurano tra i principali ambiti di sbocco nel mercato interno, mentre la crescita del settore edilizio degli ultimi anni ha contribuito a un consolidamento della Building Automation e dell'Antincendio. E sempre più numerosi sono poi i produttori italiani che guardano con interesse anche al mercato estero. rispetto alle scorse edizioni.

www.sicurezza.it

Teko S.p.A.
Via dell'Industria 5, 40068 - S.Lazzaro di Savena (BO)
www.teko.it • vendite@teko.it

ENCLOSURES Contenitori per elettronica dal 1957

Standard

Più di 1500 articoli standard, in plastica e metallo, sempre disponibili a magazzino.
Dal piccolo telecomando al Rack 19", senza tralasciare contenitori palmari e case dedicati al mondo IoT.
La gamma TEKO comprende un'ampia selezione di prodotti adatti ad ogni tipo di applicazione.



Personalizzato

Tutti i contenitori possono essere personalizzati con forature, fresature, verniciature decorative e schermanti, serigrafia, tampografia, stampa digitale, etichette e tastiere a membrana. Verificando la perfetta integrazione dell'elettronica del cliente possiamo fornire il contenitore pronto per l'assemblaggio.



Custom

Non trovi il contenitore delle dimensioni giuste o preferisci un design distintivo del tuo prodotto?
Grazie all'esperienza maturata in oltre 50 anni di attività mettiamo a disposizione del cliente tutte le nostre tecnologie e servizi: design, prototipazione rapida (FDM, Stampa 3D, SLA, SLM, PolyJET), realizzazione stampi e stampaggio. E se vuoi un contenitore di metallo possiamo offrirti: Contenitori realizzati in carpenteria metallica e fototranciatura chimica.



Famiglie di micro SAM con numerose interfacce per connettività

Microchip annuncia le famiglie SAM D5x e SAM E5x di microcontroller (MCU) a 32-bit caratterizzate da numerose interfacce per connettività, potenti performance e robusta sicurezza hardware-based per una vasta gamma di applicazioni. I microcontroller SAM D5/E5 combinano le prestazioni di un processore ARM Cortex-M4 con una Floating Point Unit (FPU). Questa architettura sposta il carico dalla Central Processing Unit (CPU), aumentando l'efficienza di sistema e consentendo applicazioni process-intensive su una piattaforma a basso consumo energetico. Con



frequenza di clock fino a 120 MHz, le famiglie D5x e E5x dispongono di un massimo di 1 MB di memoria Flash dual-panel con Error Correction Code (ECC), che consente performanti live update anche quando il sistema è in funzione. Inoltre, queste famiglie sono disponibili con un massimo di 256 KB di SRAM con ECC, importanti per applicazioni mission-critical (dispositivi medicali, server, ecc.). Questi nuovi MCU dispongono di interfacce multiple in grado di offrire flessibilità di progettazione per le più esigenti necessità di connettività nonché supporto hardware e software crittografico, consentendo agli sviluppatori di incorporare misure di sicurezza sin dall'avvio di un progetto.

www.microchip.com

TLP5214A, fotoaccoppiatore per il pilotaggio intelligente dei gate



Toshiba Electronics Europe ha annunciato il TLP5214A un nuovo fotoaccoppiatore per il pilotaggio intelligente dei gate di IGBT e MOSFET di media potenza. Il TLP5214A è pensato per varie applicazioni in cui sono usati i MOSFET e gli IGBT, inclusi i server industriali, gli inverter per l'energia solare, gli inverter per i condizionatori d'aria e i servoamplificatori.

Il TLP5214A integra funzionalità quali l'ottimizzazione del tempo di sblocco del fronte di desaturazione, del tempo di filtraggio e delle prestazioni di spegnimento soft. Queste ultime contribuiscono ad assicurare l'esecuzione delle applicazioni, sopprimendo il rumore ad impulsi corti durante la commutazione e la misura di desaturazione.

Migliorando le prestazioni del TLP5214, il nuovo dispositivo vanta un tempo di sblocco del fronte di DESAT di 1,1 μs (tipici) e un tempo di filtraggio di DESAT di 90ns (tipici). Il tempo di avvio soft di 7 μs è il doppio di quello del suo predecessore.

Il TLP5214A opera con un'alimentazione compresa fra 15V e 30V ed è in grado di fornire una corrente di uscita di picco max. di $\pm$ 4.0A. Il ritardo di propagazione ridotto di 150ns (con una deviazione di $\pm$80ns) ne supporta l'utilizzo nelle applicazioni ad alta velocità.

www.toshiba.semicon-storage.com

Sensore differenziale di corrente ad elevata precisione

Allegro MicroSystems Europe ha presentato un nuovo integrato sensore di corrente ad alta precisione basato sull'effetto di Hall, con uscite multiple di guasto programmabili, destinato sia al mercato industriale che a quello consumer, con particolare attenzione alle applicazioni di controllo motore e inverter di potenza. Uno dei vantaggi fondamentali dell'ACS720 di Allegro è quello di fornire un elevato isolamento con un ridotto numero di componenti, in gran parte merito del package proprietario SOIC-16W. L'ACS720 funziona con una singola tensione di alimentazione a 5 V e presenta una tensione di uscita compresa tra 0 e 3 V e un'uscita stabilizzata a 1,5 V zero current. Questo consente all'ACS720 di funzionare a 5V, pur avendo un'uscita compatibile con i livelli a 3,3 V di molti ADC che si trovano all'interno dei microcontrollori. Inoltre, l'elevato valore di PSRR del chip ACS720 blocca il rumore generato dagli stadi di potenza, mantenendo alta precisione anche in ambienti molto rumorosi.

Il sensore dispone di due uscite di segnalazione guasti che sono ampiamente configurabili. Denominate Fast e Slow, sono in grado di rilevare sia corto circuiti che correnti eccessive. L'uscita di guasto è di tipo open-drain.

www.allegromicro.com



SoC R-Car e Automotive Grade Linux (AGL) per le applicazioni di infotainment

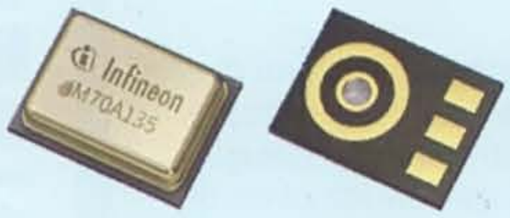
Renesas Electronics ha annunciato oggi che è iniziata la spedizione in volumi del suo primo sistema SoC R-Car completo di software basato sulla piattaforma Automotive Grade Linux (AGL). AGL rappresenta uno sforzo trasversale per sviluppare uno stack software aperto per le auto connesse. Renesas considera AGL come un passo essenziale per espandere la base degli sviluppatori software. Lo Starter Kit dell'R-Car Renesas e il software basato su AGL stanno accelerando lo sviluppo delle applicazioni di in-vehicle infotainment (IVI) e consentono ai costruttori di veicoli di avvicinarsi sempre di più all'obiettivo dell'auto connessa. Renesas è un platinum member dell'AGL Collaborative Project, e anche costruttori come Toyota Motor Corporation stanno partecipando a questa iniziativa per promuovere lo sviluppo di software open source.



La prima spedizione in volumi del Soc R-Car di Renesas che incorpora il software basato sulla piattaforma AGL riguarda la nuova Toyota Camry che debutterà alla fine di questa estate negli USA e sarà il primo modello di auto Toyota ad utilizzare la piattaforma software Automotive Grade Linux (AGL) per il suo sistema di infotainment.

www.renesas.com

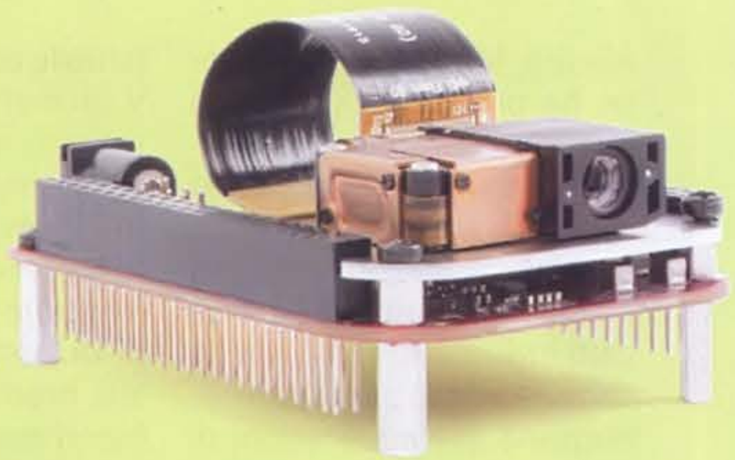
Microfoni MEMS con un rapporto segnale/rumore di 70 dB



Infineon Technologies entra nel mercato dei microfoni MEMS con contenitore ad elevate prestazioni e bassissimo rumore di fondo. I microfoni analogici e digitali si basano sulla tecnologia di Infineon denominata Dual Backplate (DBP) e sono caratterizzati da un rapporto segnale rumore (SNR) di ben 70 dB. A ciò si unisce un livello di distorsione eccezionalmente basso pari al 10 percento per un livello di pressione acustica di 135 dB (SPL). Con dimensioni di appena 4 x 3 x 1,2 mm, i nuovi microfoni MEMS sono particolarmente indicati per registrazioni acustiche di elevata qualità e per la cattura della voce a grande distanza.
Il livello di 70 dB rappresenta un miglioramento del valore di SNR di 6 dB; questo miglioramento equivale a raddoppiare la distanza da cui un utente può fornire un comando vocale catturato ad un microfono. Inoltre, i microfoni analogici e digitali MEMS presentano limitate differenze tra lotti di produzione (± 1 dB per la sensibilità e ± 2° per la fase), rendendoli particolarmente adatti per l'implementazione in array. Per questo motivo i microfoni MEMS sono perfetti per la formazione di fasci ultra-precisi e per l'annullamento del rumore.
www.infineon.com

Economico EVM per iniziare ad operare con la tecnologia DLP Pico

Texas Instruments apre la porta agli sviluppatori che intendono implementare la tecnologia DLP ad alte prestazioni utilizzando qualsiasi processore low-cost proponendo il nuovo chipset da 0,2 pollici DLP2000 e il modulo di valutazione DLP LightCrafter Display 2000 da 99 dollari; queste risorse rendono più facilmente accessibile la tecnologia DLP e consentono la progettazione di applicazioni di visualizzazione personalizzate quali le smart TV mobili, Pico proiettori, sistemi di digital signage, Pico proiettori per domotica, smartphone e tablet, display di controllo per Internet of Things (IoT).
Il nuovo DLP LightCrafter Display 2000 EVM consente agli sviluppatori di tutto il mondo di progettare facilmente con la tecnologia DLP.



Il doppio connettore EVM è configurato per supportare qualsiasi interfaccia video GPIO con capacità fino a 24 bit RGB, inclusa la scheda di sviluppo BeagleBone Black basata su processori TI Sitara AM335x; l'EVM include driver software nonché esempi di codice per Black BeagleBone che facilitano la programmazione. Tutto ciò rende la tecnologia più accessibile agli sviluppatori, consentedo di incorporare il sistema di visualizzazione ad alte prestazioni DLP Pico in nuove applicazioni.
www.ti.com

Sensore ambientale multifunzione per sistemi IoT

Omron Electronic Components Europe ha presentato un sensore ambientale multifunzione che consente ai progettisti di monitorare sette parametri utilizzando un unico modulo estremamente semplice da integrare. Progettato appositamente per i sistemi IoT autonomi, il nuovo sensore Omron 2JCIE-BL01 misura temperatura, umidità, luce, UVI, pressione barometrica, rumore e accelerazione. Nonostante le sue dimensioni compatte, il modulo 2JCIE dispone di una propria memoria interna per la registrazione dei dati rilevati dall'ambiente circostante. Le potenziali applicazioni includono il monitoraggio e il controllo di spazi industriali e d'ufficio per rendere più confortevole l'ambiente di lavoro. Il modulo è adatto anche ad applicazioni domestiche e outdoor. I dati acquisiti possono essere scaricati via Bluetooth (BTLE) tramite l'apposita applicazione gratuita scaricabile dall'App Store Android / iOS / Windows, consentendo ai progettisti di accedere facilmente alle informazioni registrate. Il modulo ha un consumo energetico molto basso: per questo può essere alimentato con una normale batteria al litio a 3VDC per una durata di circa sei mesi, in base all'intervallo di misura impostato. Il nuovo sensore ambientale 2JCIE è compatibile con il cloud e non richiede hardware aggiuntivo. Per informare l'utente di eventuali letture anomale possono essere impostati dei valori di soglia su misura.
components.omron.eu



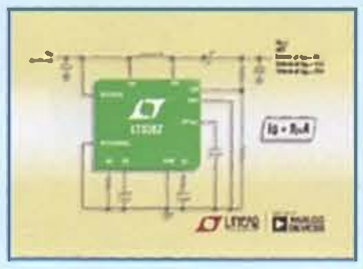
Convertitore step-down per soluzioni USB Type-C multicella



I progettisti di soluzioni USB Type-C multicella che necessitano di correnti più alte, doppio ingresso e supporto I²C dispongono oggi di una opzione altamente flessibile: MAX77756, il nuovo convertitore step-down da 24V e 500mA a bassissima corrente di quiescenza (Iq) realizzato da Maxim Integrated.
I prodotti USB Type-C devono generare un rail di alimentazione a 3,3V sempre attivo per rilevare l'inserimento di dispositivi USB. I prodotti che impiegano la gamma di tensioni Power Delivery (PD), da 5V a 20V, possono generare un rail di alimentazione digitale 'always-on' per il controllore della porta (1,8V/3,3V/5,0V) utilizzando il nuovo convertitore step-down MAX77756. Inoltre il MAX77756 ha una corrente di quiescenza di soli 20µA, che permette di prolungare la durata della batteria riducendo il consumo di energia nello stato inattivo. Per semplificare il progetto del sistema, il MAX77756 integra un circuito di ingresso duale basato su ORing con diodi ideali, che consente al chip di alimentarsi dalla batteria oppure – quando questa è scarica – da una fonte USB esterna.
www.maximintegrated.com

Convertitore step-up DC/DC Boost / SEPIC / inverter con IQ = 9μA

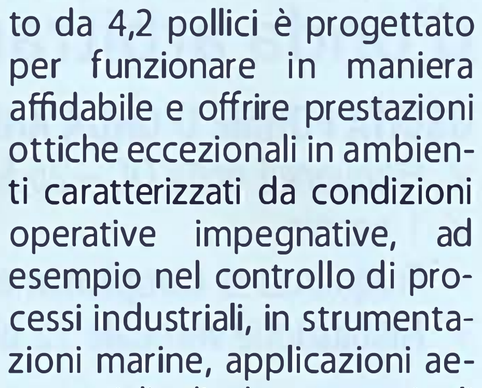
Analog Devices annuncia l'LT8362, un convertitore step-up DC / DC con frequenza di funzionamento fino a 2MHz e switch interno da 2A 60V. Con una tensione d'ingresso compresa tra 2,8V e 60V, il nuovo dispositivo è adatto per sorgenti che vanno da una batteria Li-Ion a singola cella fino a tensioni disponibili in ambito automobilistico e industriale.



L'LT8362 può essere configurato come boost, SEPIC o come convertitore invertente. La frequenza di commutazione può essere programmata tra 300kHz e 2MHz, consentendo ai progettisti di ridurre al minimo le dimensioni dei componenti esterni ed evitare bande di frequenza critiche, come quella AM. Garantisce inoltre un'efficienza del 90% con una frequenza di commutazione di 2MHz. La tecnologia proprietaria Burst Mode riduce la corrente a riposo a soli 9 μA pur mantenendo il valore del ripple di uscita al disotto di 15mVP-P. La combinazione di un contenitore DFN da 3mm x 3mm (o di un package MSOP-16E per alta tensione) con minuscoli componenti esterni consente una soluzione molto compatta ed economica.
Lo switch interno garantisce un'efficienza superiore al 95% grazie alla resistenza di conduzione di soli 165mΩ. Esso offre anche una modulazione di frequenza di tipo spread spectrum per ridurre il valore delle emissioni EMI. Un singolo pin di retroazione consente di impostare la tensione di uscita, sia per le tensioni positive che per quelle negative.
www.linear.com

Pannello TFT compatto da 4,2 pollici Rugged+

KOE ha annunciato l'introduzione di un nuovo modulo di visualizzazione TFT Rugged+ da 4,2 pollici. Il TX11D201VM0BAA è dotato della tecnologia IPS (In-Plane Switching) di ultima generazione, risoluzione WQVGA (480 x 272 pixel), formato wide 16:9 e un'ampia gamma di temperature di esercizio.
Progettata e sviluppata per applicazioni con interfaccia man-machine, la più recente tecnologia IPS offre una saturazione e una stabilità dei colori eccezionali, un contrasto elevato e livelli di nero profondi, con angoli di visualizzazione di 170° (in verticale e orizzontale). Il display da 4,2 pollici presenta un rapporto di contrasto di 1500:1 e una retroilluminazione LED con specifica di luminosità di 750 cd/m² e controllo dimming PWM integrato. Tutte queste caratteristiche assicurano immagini sempre luminose, vivaci e nitide. Il modulo TFT rinforzato da 4,2 pollici è progettato per funzionare in maniera affidabile e offrire prestazioni ottiche eccezionali in ambienti caratterizzati da condizioni operative impegnative, ad esempio nel controllo di processi industriali, in strumentazioni marine, applicazioni aerospaziali, display per veicoli e apparecchiature mediche. La resistenza in ambienti ostili è facilitata dalla sua gamma di temperature di esercizio compresa tra -30°C e +85°C e dalla gamma di temperature di stoccaggio compresa tra -40°C e +90°C.
www.koe.j-display.com



Nuovi TVS di protezione da 80A in grado di assorbire ESD ripetitive



Littelfuse ha presentato gli array di diodi TVS bidirezionali discreti da 80 A quale ultima aggiunta ai suoi array di diodi TVS di serie SP11xx (diodi SPA).
I diodi TVS bidirezionali discreti da 80 A della serie SP1103C offrono ai progettisti di circuiti opzioni di tensione di stallo inferiore per proteggere i bus elettrici a tensione inferiore dal danneggiamento da scariche elettrostatiche (ESD).
Questi diodi, prodotti utilizzando una tecnologia proprietaria a valanga di silicio, proteggono ogni piedino di I/O per offrire un elevato livello di protezione alle apparecchiature elettroniche sensibili alle ESD distruttive. Questi robusti diodi possono assorbire in sicurezza ESD ripetitive a ± 30 kV senza degrado delle prestazioni. Inoltre, ciascun diodo può dissipare tranquillamente 80 A di corrente di sovratensione 8/20 μs con tensioni di bloccaggio molto basse.
Le applicazioni per array di diodi TVS della serie SP1103C si trovano comunemente nell'elettronica automobilistica, in prodotti industriali, nell'elettronica di consumo, in interruttori/pulsanti, in apparecchiature/strumenti di test, in terminali di punti vendita, in apparecchiature medicali, in notebook.
www.littelfuse.com

BlueNRG-2, nuovo SoC Bluetooth Low Energy da ST



Il System-on-Chip di STMicroelectronics che integra la tecnologia di nuova generazione Bluetooth Low Energy (BLE) è in grado di accelerare la diffusione di oggetti intelligenti in ambito domestico, industriale, nel campo dell'assistenza sanitaria, nelle infrastrutture, nei giocattoli connessi e nei giochi intelligenti. Questo chip è in grado di supportare l'espansione esponenziale degli oggetti connessi e intelligenti.
Presente in tutti gli smartphone, tablet e PC, la tecnologia wireless Bluetooth Low Energy consente ai consumatori di interagire con qualsiasi oggetto che abbia capacità BLE dai propri dispositivi. Allo stesso modo, i fornitori di servizi possono sfruttare questa tecnologia per la fornitura di servizi e la raccolta dati basata sul cloud.
ST vede importanti opportunità per questa semplice tecnologia in grado di garantire ottima connettività e che può funzionare per mesi o anni semplicemente con una batteria a bottone. Il nuovo chip BlueNRG-2 soddisfa tutte queste esigenze grazie al suo potente processore programmabile a basso consumo, tra cui una modalità stand-by con assorbimento irrisorio. L'elevata potenza del segnale RF garantisce una comunicazione wireless affidabile mentre l'ampia memoria interna offre spazio a sufficienza per il software BLE e per il codice applicativo, evitando l'impiego di componenti esterni. Il dispositivo BlueNRG-2 è certificato Bluetooth 5.0, che assicura l'interoperabilità con le ultime generazioni di smartphone e supporta funzioni avanzate quali un maggior livello di sicurezza, maggior velocità e una maggiore capacità di trasferimento dati.
www.st.com

NUOVA GAMMA DI STRUMENTI PER IL LABORATORIO

2 Canali 200 MHz con generatore di funzioni



- ✓ Frequenza di campionamento fino a 1 GS/s
- ✓ Generatore di forme d'onda arbitrarie fino a 25 MHz con convertitore A/D 12-bit
- ✓ Funzione FFT integrata
- ✓ Software per l'analisi e il controllo in tempo reale con il PC
- ✓ Porta USB Host / USB Device

DISPLAY LCD DA 7" 64.000 COLORI 800 X 480 PIXEL

COD. **DSOFG202**

€ 549,00

Prezzi IVA inclusa.



2 Canali 70 MHz

- ✓ Frequenza di campionamento fino a 1 GS/s
- ✓ Funzione FFT integrata
- ✓ Software per l'analisi in tempo reale con il PC
- ✓ Porta USB Host

COD. **DS072** **€ 325,00**

Disponibile anche nella versione a 200MHz

COD. **DS0202** **€ 519,00**

Oscilloscopio a forma di penna - 1 canale



- ✓ Larghezza di banda 20 MHz
- ✓ Frequenza di campionamento 96 MSa/s
- ✓ Tensione max IN 50V
- ✓ Funzione FFT integrata
- ✓ Software per l'analisi in tempo reale con PC
- ✓ Alimentazione tramite porta USB
- ✓ Dimensioni 230 x 42 x 25 mm
- ✓ Peso 263 grammi

COD. **PSO2020**

€ 95,00



Oscilloscopio USB per PC 2 Canali 20 MHz

- ✓ Frequenza di campionamento 48 MS/s
- ✓ Funzione FFT integrata
- ✓ Interfaccia USB 2.0 per PC (alimentazione da USB)
- ✓ Software e memorizzazione delle forme d'onda

COD. **DSOPC22** **€ 94,00**

Disponibile anche nella versione a 100MHz

COD. **DSOPC102** **€ 325,00**

Generatore di funzioni e forme d'onda arbitrarie

€ 224,00

COD. **FAW1025G**

USCITA FORME D'ONDA ARBITRARIE

- ✓ Frequenza onda DC~25 MHz
- ✓ 1 canale
- ✓ Frequenza di campionamento 200 MSa/s
- ✓ Risoluzione verticale 12 bit

FREQUENZIMETRO

- ✓ Frequenza fino a 50 MHz
- ✓ Impedenza d'ingresso > 100 kohm
- ✓ Ampiezza segnale d'ingresso da 400 mVpp a 18 Vpp



Analizzatore logico con 34 Canali 500 Msa/S



€ 154,00 COD. **ANALOG5034**

- ✓ 34 Canali
- ✓ Impedenza d'ingresso 200 Kohm
- ✓ Tensione d'ingresso da -60 V a 60 V
- ✓ Frequenza di campionamento max. 500 MHz
- ✓ Larghezza di banda segnale d'ingresso max. 150 MHz
- ✓ Ingresso compatibile TTL, LVTTL, CMOS, LVCOMS, ECL, PECL, EIA
- ✓ Interfaccia porta USB

FUTURA ELETTRONICA®

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche e vendita on-line su: **www.futurashop.it**

STEM SHIELD

Una board di sperimentazione delle capacità delle schede Arduino, ricca di periferiche specifiche per il mondo Educational.

di BORIS LANDONI

Le pagine della nostra rivista hanno più volte ospitato, da quando il progetto Arduino è nato, articoli didattici finalizzati a insegnare la programmazione e l'utilizzo dei microcontrollori Atmel, ma anche tanto hardware di supporto sotto forma di svariati shield, ovvero di schede di espansione del microcontrollore Atmel ATmega nate per interfacciarlo con il mondo esterno, allo scopo di controllare dispositivi o acquisire segnali e dati. Molti degli shield proposti sono stati dedicati a specifiche applicazioni, ma ne esistono anche altri generici, nati per la sperimentazione e la didattica, il cui più essenziale esponente è il protoshield, costituito solamente dagli header e da piste e piazzole per montare ogni genere di componente. Quello che vi proponiamo in queste pagine è, invece, uno shield nato specificamente per rientrare nell'ambito dei programmi scolastici STEM (Science, Technology, Engineering and Mathematics) quindi risponde a una serie di direttive che vedono il mondo Arduino integrato nella scuola. Essendo un prodotto pensato per la sperimentazione, è di fatto un insieme di pe-

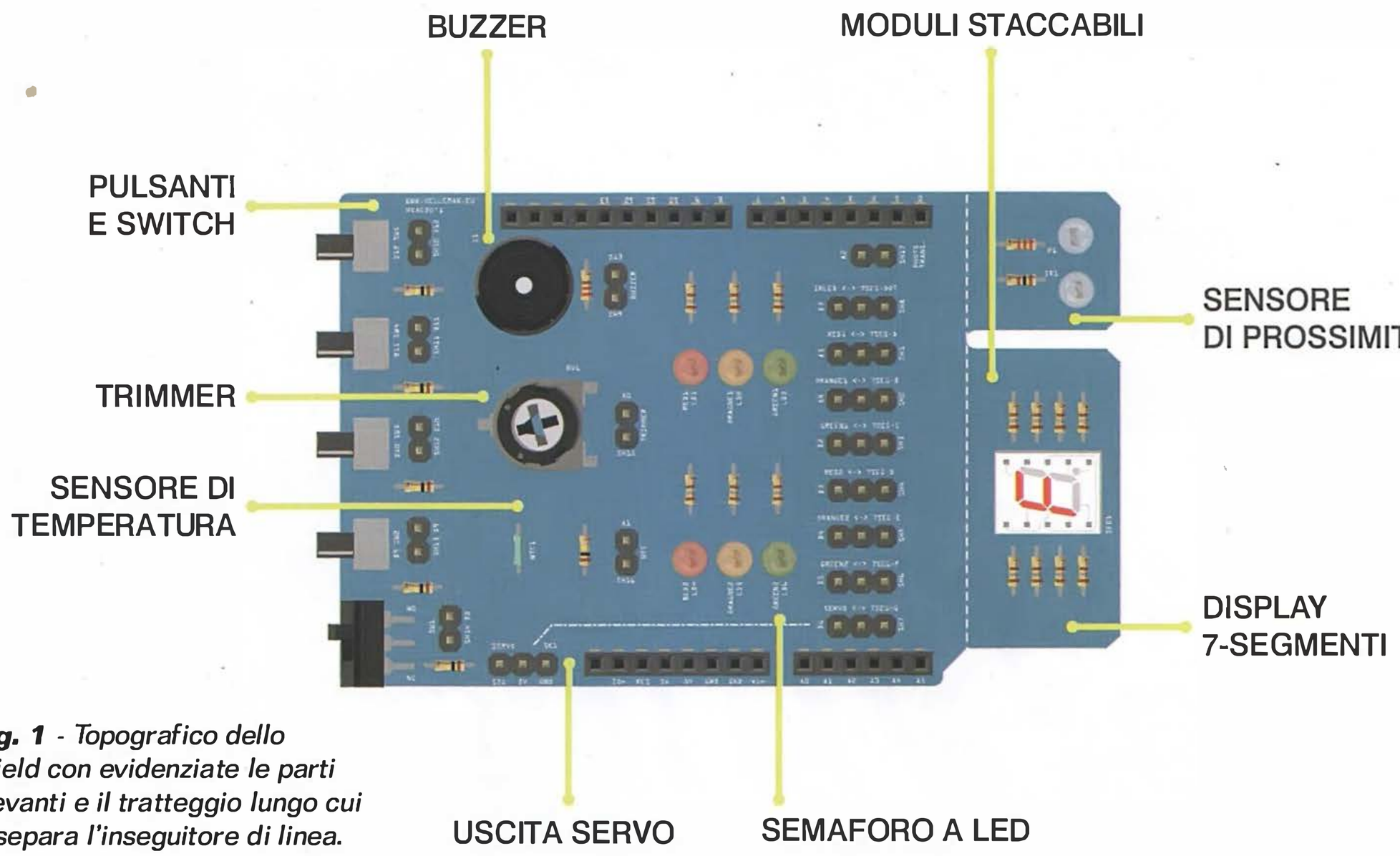


Fig. 1 *- Topografico dello shield con evidenziate le parti rilevanti e il tratteggio lungo cui si separa l'inseguitore di linea.*

riferiche e dispositivi elementari che a seconda dell'esperimento da condurre possono essere inseriti o disinseriti e connessi a uno o più pin di Arduino in base alla necessità del momento; possiamo quindi dire che lo shield è un "mucchietto" ...un paniere di hardware elementare che possiamo utilizzare al bisogno o anche solo per testare qualche sketch dimostrativo.
Il progetto è quindi l'ideale per imparare a programmare i microcontrollori, perché permette di verificare immediatamente il funzionamento di piccoli sketch per la gestione dell'accensione di LED, la generazione di suoni o il gioco elettronico, l'acquisizione di tensioni analogiche e la lettura di temperature, oltre a tanto altro.
Lo shield, chiamato STEM Shield in virtù del suo orientamento ai programmi didattici rientranti nel concetto omonimo, ha un form factor idoneo ad essere applicato alla scheda Arduino Uno e simili, nonché alla Arduino Mega; può quindi essere applicato alle nostre board Fishino, le quali, come ormai saprete, sono compatibili ed anche migliorate. Non va invece bene per l'applicazione diretta sulle più piccole Arduino e Fishino, come le Micro e Nano, ma è idoneo alle schede a 32 bit, come la Fishino 32.

SCHEMA ELETTRICO

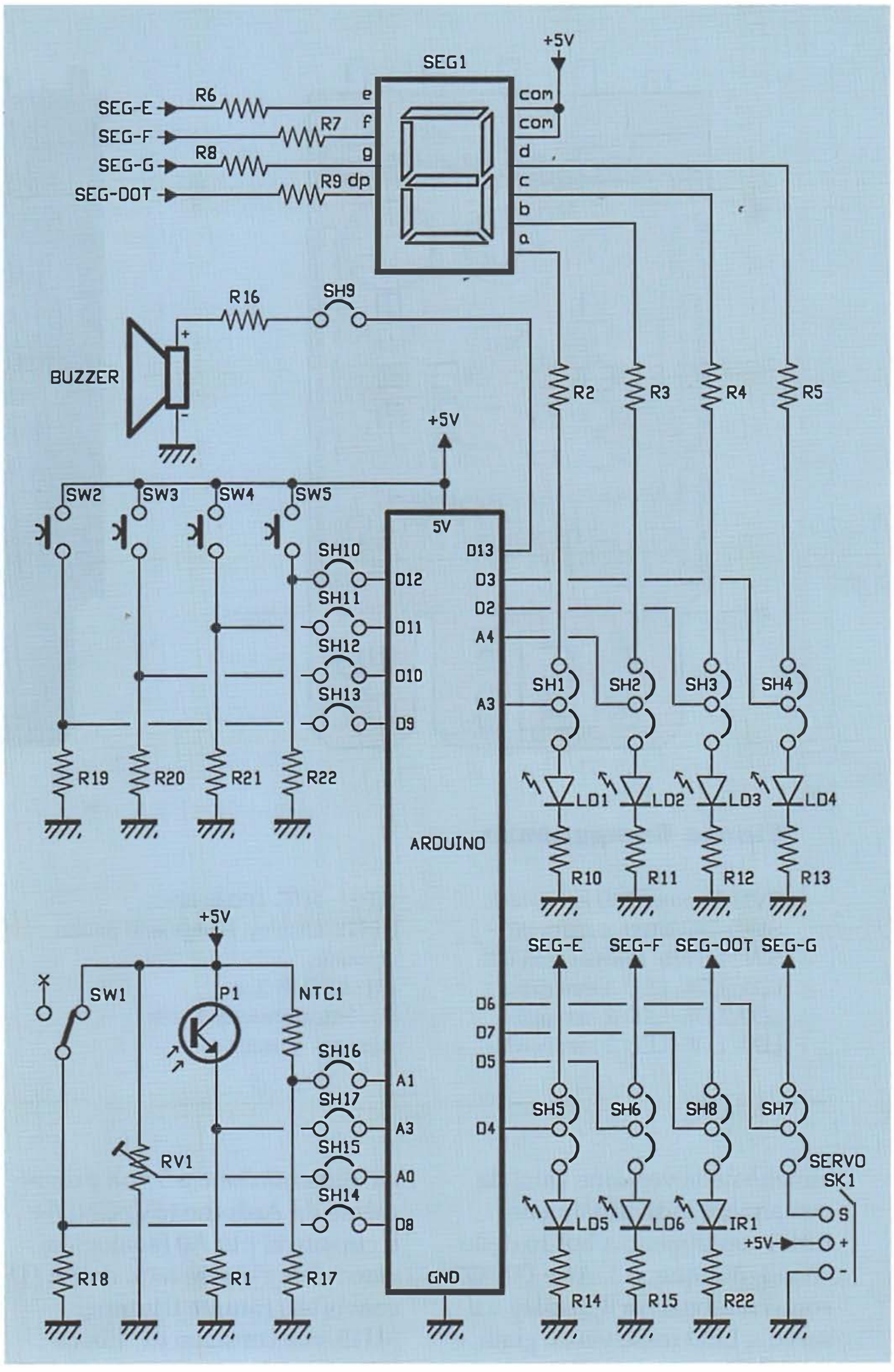
Bene, dopo questo preambolo possiamo andare alla sostanza del progetto e analizzare l'hardware a partire dallo schema elettrico; prima, però, diamo uno sguardo al layout dello shield proposto nella **Fig. 1**. In esso vediamo che lo shield è composto da un corpo principale ed un'estensione composta da due parti separabili laddove non servissero o andassero dislocate lontano; l'hardware nel complesso è composto da:

- sensore di temperatura;
- display 7-segmenti;
- 6 LED simulatori di semaforo;
- 4 pulsanti;
- 1 switch a slitta;
- 1 trimmer;
- 1 cicalino;
- 1 connettore per servo;
- 1 fototransistor all'infrarosso;
- 1 LED all'infrarosso.

Nella sezione staccabile della scheda sono collocati il display (in una metà) e il LED e il fototransistor all'infrarosso (nell'altra) con i quali potete realizzare un sensore di prossimità o un inseguitore di linea (LINE TRACKER), molto apprezzato nelle applicazioni robotiche; non a caso questa parte è staccabile, perché così potete collocarla dove serve: ad esempio nella parte bassa di un robot su ruote. Il resto si trova sullo shield base.
Il simulatore di semaforo è formato da sei LED disposti a terne, ognuna composta da un diodo rosso, uno verde e uno giallo; i pulsanti e il deviatore a slitta servono come dispositivi di input per impartire comandi, mentre i LED possono comunque essere utilizzati disgiuntamente per fornire segnalazioni, al pari del display a LED 7-segmenti che di trova sulla parte staccabile.
Il trimmer consente di testare l'ADC, ovvero di impostare

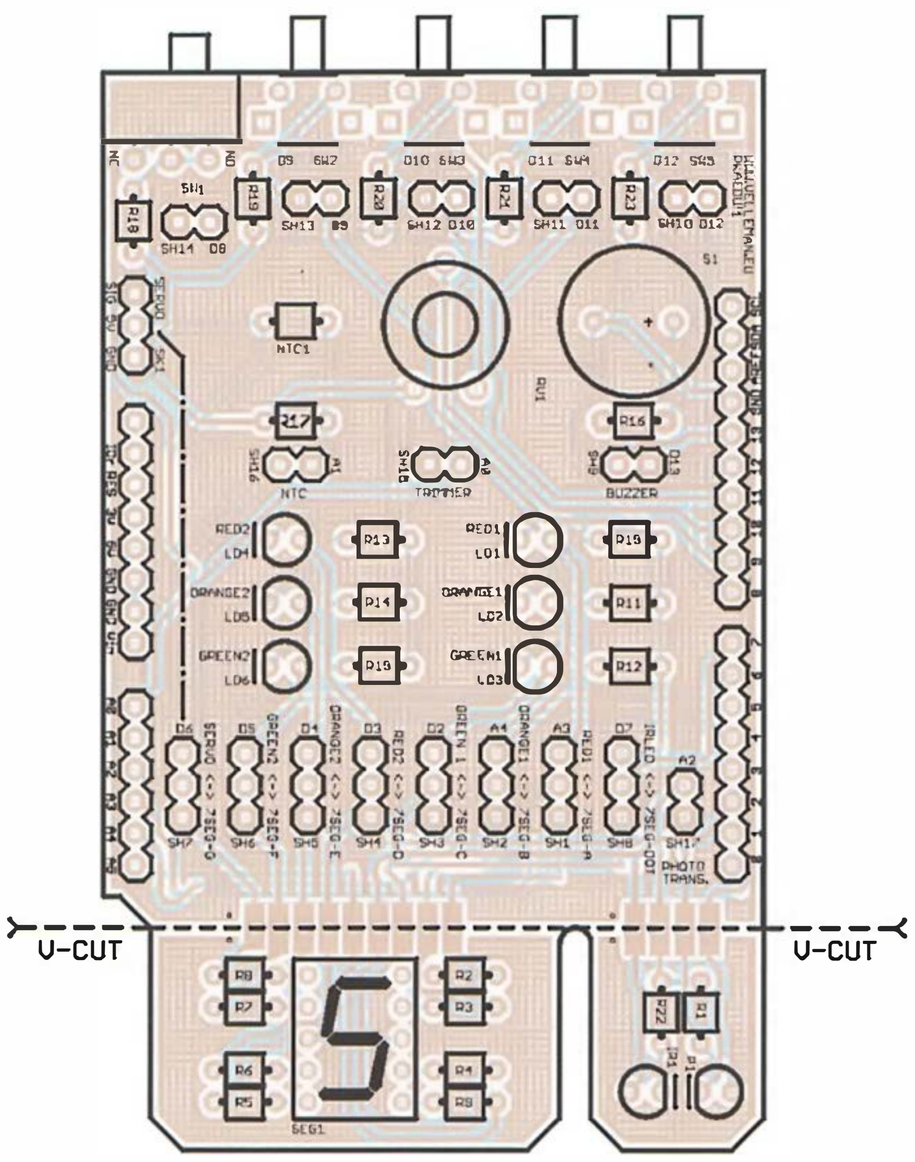
soglie e valori analogici; i jumper ON/OFF servono ad abilitare/disattivare alcuni blocchi dello shield. Il sensore di temperatura è un termistore NTC montato a bordo della scheda.
Dopo questo riepilogo, guardiamo e analizziamo lo schema elettrico, che è realizzato intorno ai connettori standard per gli header Arduino.
Partiamo dai pulsanti, ognuno dei quali è del tipo normalmente aperto ed è collegato da un estremo al positivo di alimentazione terminante sulla linea 5V di Arduino e dall'altro a una linea di Arduino attraverso un jumper che permette di collegarlo e escluderlo, allo scopo di lasciare utilizzare i corrispondenti pin degli header; ciascuna linea di quelle assegnate da SW2÷SW5 è dotata di un resistore di pull-down che tiene a massa l'estremo del rispettivo pulsante. I jumper di inserzione dei pulsanti sono quattro, siglati SH10÷SH13, ossia tanti quanto i pulsanti stessi; con i jumper chiusi ed SH10 a D2. SH11 è collegato a D11 di Arduino, SH12 a D10, SH13 a D9.
Il deviatore viene invece utilizzato come interruttore (un estremo e il centrale sono tra loro cortocircuitati) collegato con l'estremo libero al positivo 5V e il cursore (terminato a massa tramite il resistore di pull-down R18, da 100 kohm) collegato, tramite il jumper SH14, alla linea D8 dell'header Arduino.

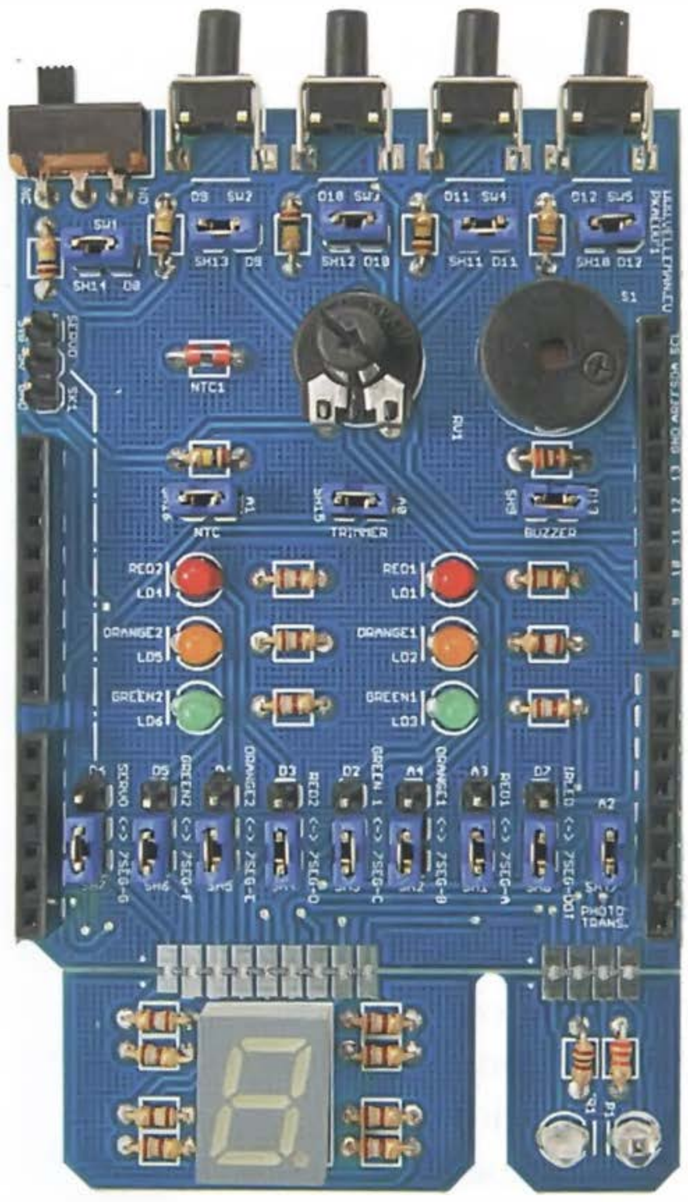
[schema **ELETTRICO**]



Passiamo adesso al cicalino piezoelettrico, che è del tipo senza elettronica e può quindi essere pilotato con segnali di varia frequenza, anche PWM, per fornire semplici note acustiche o suonare brani musicali; il suo polo negativo è fisso a massa, mentre il positivo viene collegato, tramite la resistenza limitatrice della corrente, R16, al pin header D13, tramite il jumper SH9.
Il display 7-segmenti a LED è del tipo ad anodo comune e nel circuito è collegato con l'anodo al positivo 5V di Arduino e i catodi dei 7 segmenti e del punto decimale collegati, tramite jumper a due vie, alle linee A3, A4, D2, D3, D4, D5, D6, D7, cui corrispondono rispettivamente i segmenti A, B, C, D, E, F, G, DP (punto decimale o DOT); i ponticelli sono a due vie perché permettono sia di collegare/scollegare i catodi relativi ai segmenti del display, sia, in opzione, a connettere/sconnettere la linea del servo e i catodi dei 6 LED. Tale soluzione è stata voluta perché i pin di Arduino non sono abbastanza,

[piano di **MONTAGGIO**]





Elenco Componenti:

RV1: Trimmer 100 kohm MO
SW1: Deviatore a slitta 90°
SW2÷SW5: Microswitch 90°
LD1, LD4: LED 3 mm rosso
LD2, LD5: LED 3 mm giallo
LD3, LD6: LED 3 mm verde

NTC1: NTC 100 kohm
SEG1: Display 7 segmenti anodo comune
IR1: LED IR 3 mm
P1: Fototransistor 3 mm
Buzzer: Buzzer

Varie:
- Strip mschio 3 vie (9 pz.)
- Strip maschio 2 vie (9 pz.)
- Jumper (17 pz.)
- Strip M/F 6 vie
- Strip M/F 8 vie (2 pz.)
- Strip M/F 10 vie
- Circuito stampato (87 x 54 mm)

almeno nella versione Uno, da comandare individualmente tutti i componenti a bordo dello shield; dunque, A3, A4 e D2÷D7 sono condivisi tra il display e il servo, i LED rossi, verdi, gialli, e il LED IR montato sulla sezione staccabile dello shield.
I jumper interessati sono da SH1 a SH8 per i segmenti dall'A al punto (DOT) del display; da SH1 a SH6 commutano i LED a luce visibile, SH7 commuta il LED IR ed SH8 il canale di segnale del connettore per il servocomando.
Passiamo adesso alla sezione "sensoristica" con il trimmer, che è collegato con gli estremi all'alimentazione a 5 volt proveniente da Arduino (5V/GND) e il cursore al pin A0 (analogico, giacché dev'essere letto dall'A/D converter) tramite il jumper SH15, che consente di "liberare" A0 laddove serva per altri utilizzi, senza per forza dover rimuovere lo shield, ad esempio perché potrebbe essere necessario mantenere alcune periferiche a bordo dello stesso. La tensione che raggiunge A0 cresce portando il cursore del trimmer verso il 5V e decresce, fino ad annullarsi, ruotandolo verso GND.
Il termistore NTC che fa da sensore di temperatura è collegato a partitore con la resistenza R17 ed il partitore è alimentato con i 5 volt; il punto di unione dei due resistori si collega alla linea A1 (analogica anch'essa e quindi letta dall'ADC) tramite il ponticello SH16 (che permette di liberare il pin di Arduino); al crescere della temperatura la tensione letta diminuisce, mentre cresce al calare della temperatura. Il range è tra circa 1 e 4 volt, ma può variare in base ai valori resistivi che il termistore utilizzato assume in corrispondenza della minima e massima temperatura cui può essere sottoposto.
Chiudiamo la descrizione dello

schema elettrico con il fototransistor siglato P1, che è incapsulato in un contenitore in resina trasparente tipo LED, dotato di due soli piedini che portano all'esterno il collettore e l'emettitore; la base è isolata perché poco rilevante, almeno fin quando il componente non deve trovarsi a commutare dall'interdizione alla conduzione troppo velocemente (allorché la ricombinazione delle cariche nella base rallenterebbe la commutazione). Il fototransistor lavora nella configurazione a emitter-follower ed ha il collettore alimentato dai soliti 5 volt e l'emettitore collegato a massa tramite l'apposito resistore (R1) nonché alla linea analogica A2 di Arduino, attraverso l'immancabile jumper (in questo caso è SH17). Il fototransistor viene impiegato in questa configurazione perché consente di ottenere un segnale analogo al livello di illuminazione ricevuta dalla base, il che è utile proprio per le applicazioni in cui serva determinare il grado di riflessione della luce del LED IR1, come ad esempio nella ricerca di una linea segnata a terra; per questo il segnale sull'emettitore viene letto da una linea di Arduino assegnabile all'ADC interno.

REALIZZAZIONE PRATICA

Completata la descrizione dell'hardware passiamo adesso alle note costruttive: lo shield si realizza su un circuito stampato a doppia faccia ottenibile per fotoincisione dopo aver scaricato dal nostro sito Internet *www.elettronicain.it* le tracce dei due lati rame; incisa e forata la basetta potete montarvi i componenti, che, nell'ottica di rendere la costruzione accessibile a tutti, sono a montaggio tradizionale (THT) e non in SMD, come ormai spesso vedete. Partite disponendo le resistenze e i LED, da disporre come mostrato nel disegno di montaggio visibile in queste pagine, quindi passate al display a 7 segmenti, da orientare come mostrato e da appoggiare con il corpo al circuito stampato. Diversamente è una buona soluzione quella di montare il display su uno zoccolo per circuiti integrati DIP (passo 2,54x7,5 mm) tagliato a misura.
Inserite e saldate tutti i pin-strip a passo 2,54 mm che servono a realizzare i jumper, rammentando che quelli da SH1 a SH8 sono a tre poli, mentre i restanti sono da due poli.
Fatto ciò, inserite e saldate i pulsanti da c.s. con terminali a 90° e il deviatore a slitta, anch'esso con i terminali piegati ad angolo retto. Ora potete montare il trimmer e il cicalino, quest'ultimo da orientare (perché ha una polarità) come mostrato nel piano di montaggio visibile nella pagina accanto.
Montate, infine, gli strip femmina a passo 2,54 mm con i piedini lunghi per l'inserzione negli header di Arduino: ne occorrono due da 8 contatti, uno da 6 ed uno da 10, visto che le connessioni sono standardizzate per Arduino Uno.
Prima di procedere è opportuno richiamare la vostra attenzione sul fatto che, come si vede dalla traccia lato rame (e come proposto dalla **Fig. 2**, che mostra lo shield visto da sotto), la parte staccabile del circuito, ossia l'appendice posteriore riportante LED infrarosso, fototransistor e display a 7 segmenti, pur essendo attraversata da un solco è elettricamente unita dalla metallizzazione dei fori. Infatti per la connessione tra la porzione base e quella staccabile sono previste due file di piazzole rettangolari affacciate le une alle altre e

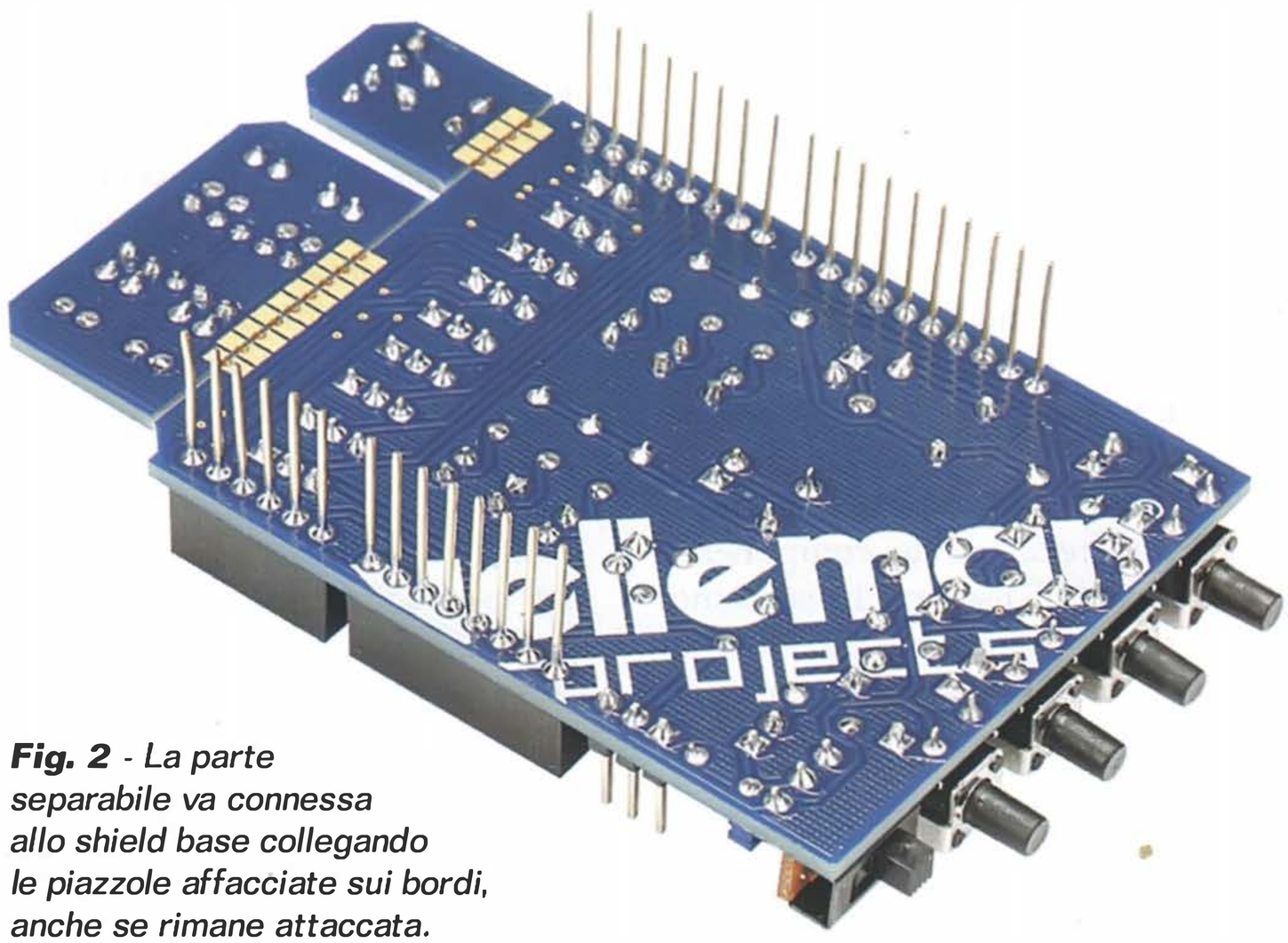
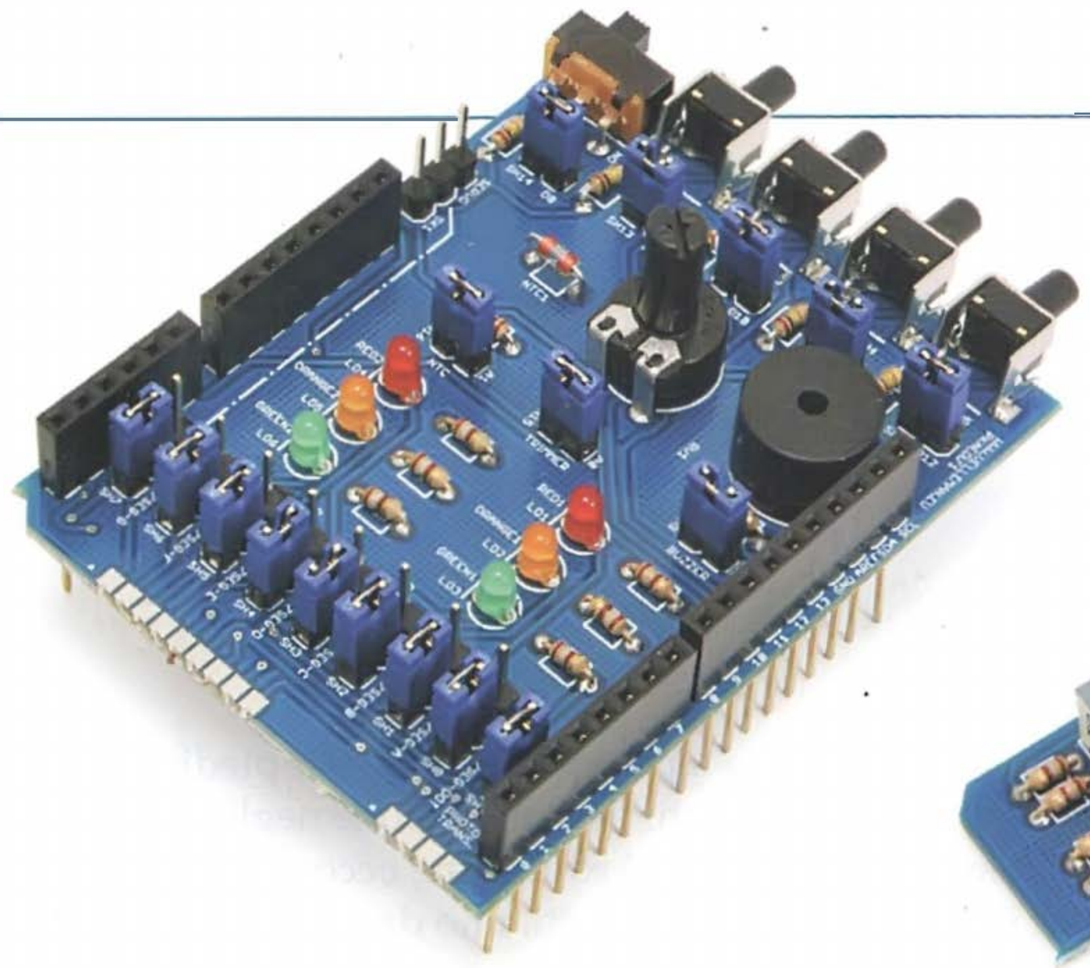
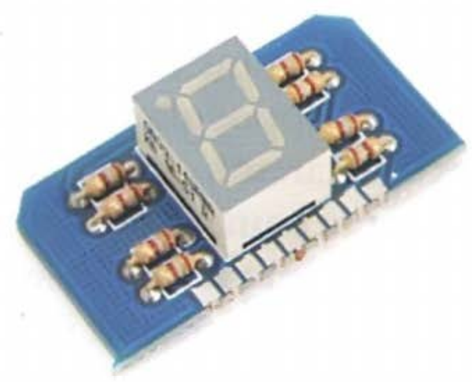


***Fig. 2** - La parte separabile va connessa allo shield base collegando le piazzole affacciate sui bordi, anche se rimane attaccata.*



Le due appendici staccate dallo shield





collegate da una metallizzazione verticale quasi invisibile, perciò se intendete mantenere il circuito stampato tutto intero non serve stagnarle.
Non dimenticate che per una corretta acquisizione della tensione fornita dal fototransistor bisogna che i fili di collegamento, se separate la scheda, non siano più lunghi di un paio di metri, pena un rilevamento affetto da disturbi.
Potete decidere, in base alle vostre esigenze, di staccare solo il modulo LINE TRACKER e lasciare a bordo l'appendice con il display (ricordando comunque di saldarla come appena spiegato) e viceversa.
Completate le saldature, il circuito è pronto all'uso, ossia ad essere applicato ad Arduino.

UN PO' DI ESERCIZIO

Bene, siccome lo shield nasce per fare esercitazioni e sperimentare come si utilizzano le periferiche, vediamo adesso qualche semplice sketch d'esempio che permette di gestire alcuni dei componenti a bordo; gli esempi che seguono sono riferiti ad Arduino Uno, tuttavia modificando la definizione dei pin potete adattarli anche ad altre board.
Il primo ci permette di accendere e spegnere in sequenza i LED LD4, LD5, LD6 (rispettivamente rosso, giallo e verde) a una velocità predefinita e ciclicamente.
Per prima cosa bisogna definire gli I/O di Arduino che userete per alimentare i LED e impostarli come uscite, quindi costruire il loop che attiva le tre porte digitali in uscita.
Lo sketch è riportato nel **Listato 1**, dove utilizziamo una struttura di setup e infine mettiamo la struttura che viene eseguita come programma e ripetuta all'infinito (**void loop()**). Nel setup definiamo le istruzioni *pinMode (pin, modo)*, che servono, prima di utilizzare i pin digitali, a definire per ciascuno se sono nel modo INPUT o OUTPUT. Se non li definiamo con l'istruzione pinMode, questi restano in uno stato ad alta impedenza e quindi inattivi.
Il primo passaggio consiste nel definire i pin utilizzati, operazione, che comunica ad Arduino a quali contatti digitali corrispondono le variabili: stabiliamo che led1 corrisponde al piedino 3 (LD4), led2 al 4 (LD5) e led3 al 5 (LD6).
L'istruzione pinMode nel setup dev'essere impartita contenendo il nome della variabile, quindi la modalità: per esempio *pinMode(led1, OUTPUT)* significa che led1, assegnata in fase di definizione al contatto 11 di Arduino, diverrà un'uscita.
Segue la creazione del loop, ovvero del ciclo che vedrà l'avvicendarsi delle istruzioni di assegnazione dei livelli logici ai piedini specificati; queste istruzioni sono digitalWrite e delay: la prima assegna il livello logico al piedino di uscita digitale e la seconda stabilisce il tempo che deve trascorrere prima di passare alla successiva. Ne deriva che essendo delay il tempo che passa dall'esecuzione di un'istruzione e la successiva e che, essendo le istruzioni in sequenza, ogni stato logico dura il tempo assegnato come parametro a delay. Nel nostro caso il ciclo sarà composto da un'istruzione digitalWrite che assegna lo stato logico alto a una linea, poi da un delay, quindi da una nuova digitalWrite che assegna lo stato basso; il tutto moltiplicato per tre, ovvero la prima volta per led1, la seconda per led2 e la terza per led3, poi il ciclo ricomincia.
Quanto alle istruzioni coinvolte, ricordiamo che la sintassi di digitalWrite è digitalWrite(pin, livello) dove al posto di pin mettete il nome assegnato nella definizione dei pin (per esempio led1) e al posto di livello scrivete se deve assumere LOW o HIGH (sono parole riservate). Ricordate che "pin" può essere una variabile. Per l'istruzione delay, la sintassi prevede invece di specificare tra parentesi il ritardo in millisecondi.
Affinché l'applicazione funzioni, sullo shield i jumper SH4, SH5 ed SH6 dovranno essere chiusi sui LED corrispondenti.

10 note e un trimmer

Passiamo a un secondo esempio dove ci proponiamo di far riprodurre al cicalino (BUZZER) 10 note acustiche la cui frequenza dipende dalla posizione del cursore del trimmer: per prima cosa dobbiamo definire i piedini

Listato 1

```
/*
Gestione uscite di Arduino per accendere in sequenza tre LED
*/
// definizione dei pin
int led1 = 3;
int led2 = 4;
int led3 = 5;
void setup() {
        // impostazione dei pin come uscite
        pinMode(led1, OUTPUT);
        pinMode(led2, OUTPUT);
        pinMode(led3, OUTPUT);
}
void loop() {
        digitalWrite(led1, HIGH); // accendi il led
        delay(1000); // aspetta un secondo
        digitalWrite(led1, LOW); // spegni il led
        delay(1000); // aspetta un secondo
        digitalWrite(led2, HIGH); // accendi il led
        delay(1000); // aspetta un secondo
        digitalWrite(led2, LOW); // spegni il led
        delay(1000); // aspetta un secondo
        digitalWrite(led3, HIGH); // accendi il led
        delay(1000); // aspetta un secondo
        digitalWrite(led3, LOW); // spegni il led
        delay(1000); // aspetta un secondo
}
```

di ingresso e uscita, che in questo caso sono l'ingresso analogico A0 (per questo dovete chiudere il jumper SH15 che collega ad A0 il cursore dell'RV1) e l'uscita digitale 13, che è quella assegnata al cicalino (allo scopo dovete chiudere il jumper SH9). Per generare le note utilizziamo un output digitale che produce impulsi alla frequenza corrispondente a quella delle note. Arduino dovrà per prima cosa leggere il valore di tensione portato da esso ad A0; quest'ultimo verrà letto dall'A/D converter.
Con riferimento al **Listato2**, che riporta il codice, la definizione dei pin è:
int buzzer = 13;

Listato 2

```
/*
Come riprodurre tramite delle note in base al valore letto da un trimmer
*/

// definizione dei pin
int buzzer = 13;
int trimmer = A0;

// definizione variabili
int numNote = 10;
int note[] = {261, 277, 294, 311, 330, 349, 370, 392, 415, 440};  //array frequenze delle note da riprodurre
int valoreletto = 0;
int notascelta = 0;

void setup()
{
  for (int i = 0; i < numNote; i++)      //finché la variabile i non è uguale a numNote, incrementala
  {
    tone(buzzer, note[i]);               //genera sul pin buzzer una frequenza pari al contenuto
                                         //della variabile note[i]
    delay(200);                          //aspetta 200ms
  }
  noTone(buzzer);                        //disabilita l'uscita
}

void loop()
{
  valoreletto = analogRead(trimmer);          //leggi il trimmer
  notascelta = map (valoreletto, 0, 1023, 0, 10);  //scala il valore letto
  if (notascelta==10) {                       //se la nota da riprodurre è la 10 (l'array va da 0 a 9)
    noTone(buzzer);                           //spegni il buzzer
  }
  else                                        //altrimenti
  {
    tone(buzzer, note[notascelta]);           //esegui la nota
  }
  delay(200);                                 //aspetta 200ms
}
```

Listato 3

```
/*
Leggere gli ingressi digitali e accendere un LED.
*/

// definizione dei pin
int led1 = 3;
int button = 9;

void setup() {
  // impostazione pin come uscita
  pinMode(led1, OUTPUT);

  // impostazione pin come ingresso
  pinMode(button, INPUT);
}


void loop() {

  if (digitalRead(button)==1){     //se il pulsante è premuto
     digitalWrite(led1, HIGH);    // accende il led
    }
    delay (500);         // pausa per antirimbalzo
  }
 if (digitalRead(button)==0){     //se il pulsante è rilasciato
     digitalWrite(led1, LOW);    // spegne il led
    }
    delay (500);         // pausa per antirimbalzo
  }
```

int trimmer = A0.

Poi definiamo le variabili:
int numNote = 10;
int note[] = {261, 277, 294, 311, 330, 349, 370, 392, 415, 440};
int valoreletto = 0;
int notascelta = 0;

"numNote" corrisponde al numero della nota da suonare, int note[] è un array contenente la frequenza delle note da riprodurre, mentre "valoreletto" è la lettura dell'A/D converter e "notascelta" la nota da suonare. L'array permette di alleggerire il codice e quindi risparmiare spazio nella memoria di programma del microcontrollore di Arduino, sostituendo la quantità di istruzioni che servono a suonare le note (o comunque a fare uscire da un certo pin digitale di Arduino Uno una frequenza) con una sola int note[] che contenga di volta in volta tra le parentesi quadre il numero della frequenza da riprodurre. Utilizzando l'array possiamo usare un unico ciclo di for che viene ripetuto le volte che serve, semplicemente sostituendo il parametro dell'istruzione, invece di tanti cicli di for quante sono le note. Ogni nota viene definita dalla posizione in cui è stata specificata nell'array, quindi int note[0] significa che viene resa disponibile in output (pin12) la frequenza di 261 Hz, int note[1] che viene riprodotta la frequenza di 277 Hz e così via; int note[9] corrisponde a 440 Hz. Detto questo, andiamo alla definizione del setup: la sintassi di for che utilizziamo è:

for (int i = 0; i < numNote; i++)

indicante che la variabile "i" deve essere incrementata fino a raggiungere il valore della variabile numNote. L'apertura di una nuova parentesi graffa indica che nel ciclo for viene eseguito il comando *tone(buzzer, note[i])*, il quale fa generare un segnale di frequenza pari al contenuto della variabile note[i], dopodiché si aspetta un periodo di 0,2 secondi (200 ms) prima di eseguire altre istruzioni (delay(200)) e successivamente viene sospesa la generazione del segnale *(istruzione noTone(buzzer)*).
Nella sintassi del comando tone inseriamo la linea di Arduino sulla quale mandare la frequenza generata (in questo caso specifichiamo buzzer) e la frequenza, che nello specifico è funzione della variabile note, la quale a sua volta dipende da i.
Dopo il setup abbiamo il loop che verrà eseguito dallo sketch: dato che l'elemento da cui dipende la generazione dei suoni è la variabile (valoreletto) ricavata dalla posizione del cursore del potenziometro, per prima cosa il firmware deve acquisire la tensione su A0; ciò viene ottenuto con l'istruzione *analogRead(trimmer)*.
Subito dopo viene eseguita l'istruzione che divide l'intera lettura dell'A/D converter in dieci passi: questo permette di dividere idealmente la corsa del potenziometro slider in dieci porzioni, ognuna delle quali corrisponde a far suonare una certa nota. Il campo dell'intera corsa è, lo ricordiamo, circa 2,5 volt, diviso in 1.024 porzioni (da 0 a 1.023) all'uscita dell'A/D onverter e a sua volta diviso in 10 dal programma. La ripartizione viene eseguita tramite la funzione *map (valoreletto, 0, 1023, 0, 10)*; la parte tra parentesi dell'istruzione map indica che valoreletto, il cui campo è compreso tra 0 e 1.023, deve essere ricondotto a un valore tra 0 e 10 (11 valori). La funzione permette di rimappare i valori adeguando la risoluzione a 10 bit ai passi richiesti dalla nostra applicazione, sia essa una lettura o una scrittura; nel nostro caso ci permette di adattare la scala dell'ADC ai 10 passi desiderati, ma si potrebbe utilizzare, ad

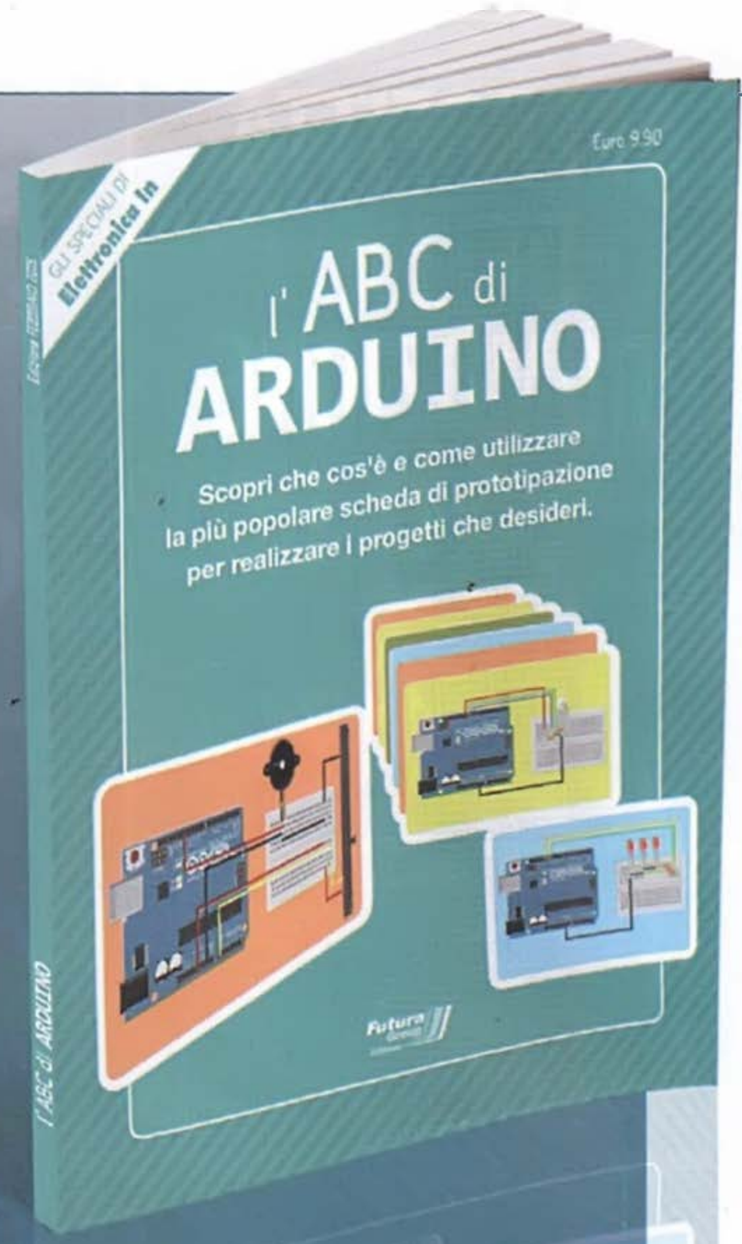
esempio per un'uscita PWM, per trasformare in un valore differente i 255 passi in cui è divisa la variazione 0÷100% del duty-cycle.
La sintassi generale è:
map(variabile, daMin, daMAX, aMin, aMAX). Con essa, il valore della variabile viene convertito dal suo intervallo originario (0, 1.023) i cui valori minimo e massimo sono definiti con daMin e daMAX, a quello di destinazione definito con aMin e aMAX ai due estremi. Si tratta di calcoli effettuati in matematica intera e quindi si applica l'arrotondamento per troncatura dei decimali.
Andiamo oltre e vediamo che dopo la funzione, nello sketch viene eseguita una struttura if/else, per cui se la nota da riprodurre è oltre la 9 (frequenza più alta) si spegne (livello logico zero) il pin 13, altrimenti si esegue l'istruzione *tone(buzzer, note[notascelta])* per suonare la nota corrispondente a valoreletto.
Dopo i soliti 200 ms *(delay(200))* si torna a leggere il valore della tensione fornita dal potenziometro. Notate che il delay così basso (appena 0,2 secondi) è stato voluto per fare in modo che il circuito possa rispondere molto rapidamente alle variazioni di posizione del cursore del trimmer e quindi suonare troppo velocemente.
Notate anche che abbiamo voluto dividere la corsa in un passo in più delle note da suonare per consentirvi di disinserire il tono acustico; infatti per ogni posizione del cursore il circuito suona la nota corrispondente, quindi serve una posizione di spento. Tale posizione corrisponde ad avere il cursore del potenziometro tutto verso l'estremo collegato al positivo 5V (valore 1.023).
L'ultimo esercizio che vi proponiamo coinvolge i pulsanti e consiste semplicemente nel far accendere il LED LD4 quando si preme SW2. Il codice corrispondente è nel **Listato 3**, dove inizialmente impostiamo il pulsante e l'uscita digitale del LED, quindi dopo la fase di setup (dove stabiliamo quali sono i pin di uscita e d'ingresso utilizzati) avviamo il loop, ossia il ciclo per gestire lo stato delle uscite sulla base di quello dei pulsanti; semplicemente, se la linea D9 di Arduino è allo stato 1 il LED viene acceso. Il *delay (500)* viene inserito come antiirimbalzo, per evitare false commutazioni. Rilasciando SW2 (*if (digitalRead(button)==0)*) il LED torna spento; a ciò provvede l'istruzione *digitalWrite(led1, LOW)* seguita dall'immancabile delay per l'antirimbalzo.

CONCLUSIONI

Bene, con questo abbiamo terminato; naturalmente le esercitazioni non si limitano a quelle qui proposte e sul web ne potrete trovare tante altre che utilizzano tutte le periferiche a bordo del nostro shield.
Buona sperimentazione! ■

Lo shield è l'ABC

Per sperimentare con lo STEM Shield cosa c'è di meglio di esercizi già pronti, intuitivi e dall'elevato contenuto didattico? Ebbene, tutto questo lo trovate nel volume l'ABC di Arduino, che dopo un preambolo didattico su Arduino e la sua programmazione, con semplici sketch commentati e spiegati consente di testare la gestione di LED, potenziometro, cicalino, pulsanti, servocomandi ed altre periferiche elementari. Sarà sufficiente, laddove necessario, adattare le definizioni dei pin e gli esercizi generici proposti si adatteranno perfettamente allo shield! Il volume si acquista da Futura Elettronica, ovvero on-line su *www.futurashop.it*.





per il MATERIALE

Il materiale elencato di seguito può essere acquistato presso Futura Elettronica. Lo shield KAEDU educational per Arduino, da montare, (cod. KAEDU) è in vendita al prezzo di Euro 22,00; il libro "l'ABC di ARDUINO" (cod. ABCARDU) è disponibile a Euro 9,90.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775
http://www.futurashop.it

APPROFITTA DELLA PROMOZIONE

€ 399,00

anziché

~~€ 417,00~~

Una "piccola" CNC per i tuoi GRANDI PROGETTI

Acquista tutti i componenti in un unico ordine e potrai avere la nuova CNC completa di parti meccaniche ed elettroniche.

Progettata per uso hobbistico, dispone di una struttura rigida in PVC. Permette di incidere circuiti stampati, effettuare lavorazioni su legno, plastica e alluminio. Dimensioni utili di lavoro 200x180x60 mm.

Tutti i componenti sono acquistabili anche separatamente

SET SCHEDA 3 ASSI CON ACCESSORI



cod. SET3ASSI

€ 75,00

ELETTROMANDRINO 36 VDC - 300 WATT



cod. ELEMANDER11

€ 85,00

MOTORE PASSO-PASSO BIPOLARE - 1,5 A



cod. STEPMOT03

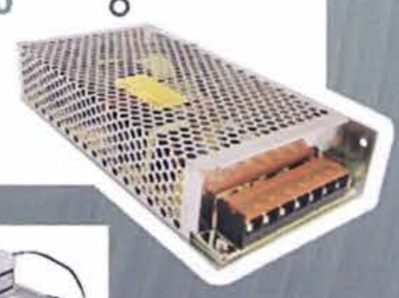
€ 18,00

(sono necessari 3 pezzi)

ALIMENTATORE SWITCHING 24 VDC - 144 W

cod. MW14524

€ 24,00



MECCANICA CNC IN KIT

cod. CNC2018

€ 179,00



N.B. La confezione non comprende i tre motori passo-passo (cod. STEPMOT03), la scheda di controllo (cod. SET3ASSI), l'elettromandrino (cod. ELEMANDER11) e l'alimentatore (cod. MW14524).

Prezzi IVA inclusa.

Istruzioni in ITALIANO!

ACQUISTALA SUBITO !

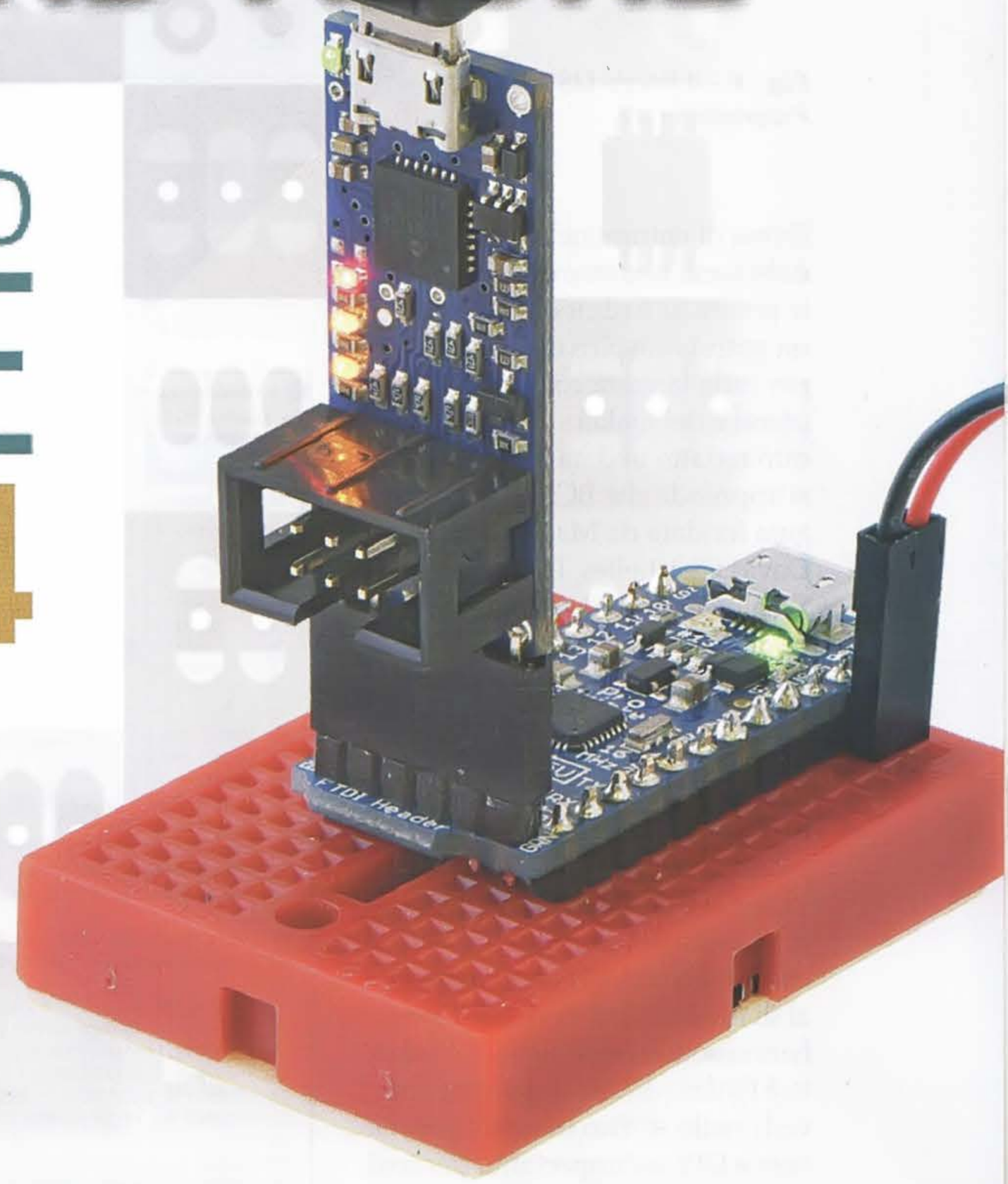
www.futurashop.it/cnc

FUTURA ELETTRONICA®
Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche e vendita on-line su:
www.futurashop.it

PROGRAMMAZIONE AVR STAND-ALONE

ARDUINO IDE 1.8.4



Impariamo a programmare i microcontrollori Atmel della serie AVR utilizzando l'IDE Arduino e il nuovo programmatore Pololu USB AVR v2.

di DIEGO FRASSON

Malgrado l'argomento sia già stato affrontato in passato, in questo articolo torneremo a descrivere come si programmano i microcontrollori Atmel AVR in modalità stand-alone (ossia da soli e non montati in alcuna schèda Arduino...) sfruttando l'ambiente di sviluppo di Arduino; l'occasione ci è propizia per riproporre la trattazione aggiornandola alla luce di tutte le novità che si sono avvicendate dalla prima pubblicazione della materia ad oggi, grazie all'aggiornamento del software IDE (Integrated Development Environment).
Riprendere l'argomento è anche il modo per farvi vedere come utilizzare tool hardware apparsi di recente sul mercato, come ad esempio l'USB AVR Programmer v2 della Pololu, che in questo articolo impiegheremo per programmare il nostro microcontrollore, utilizzando come piattaforma software l'IDE Arduino e come hardware il nostro Personal Computer, cui collegheremo l'USB AVR Programmer tramite un cavo e una porta USB.

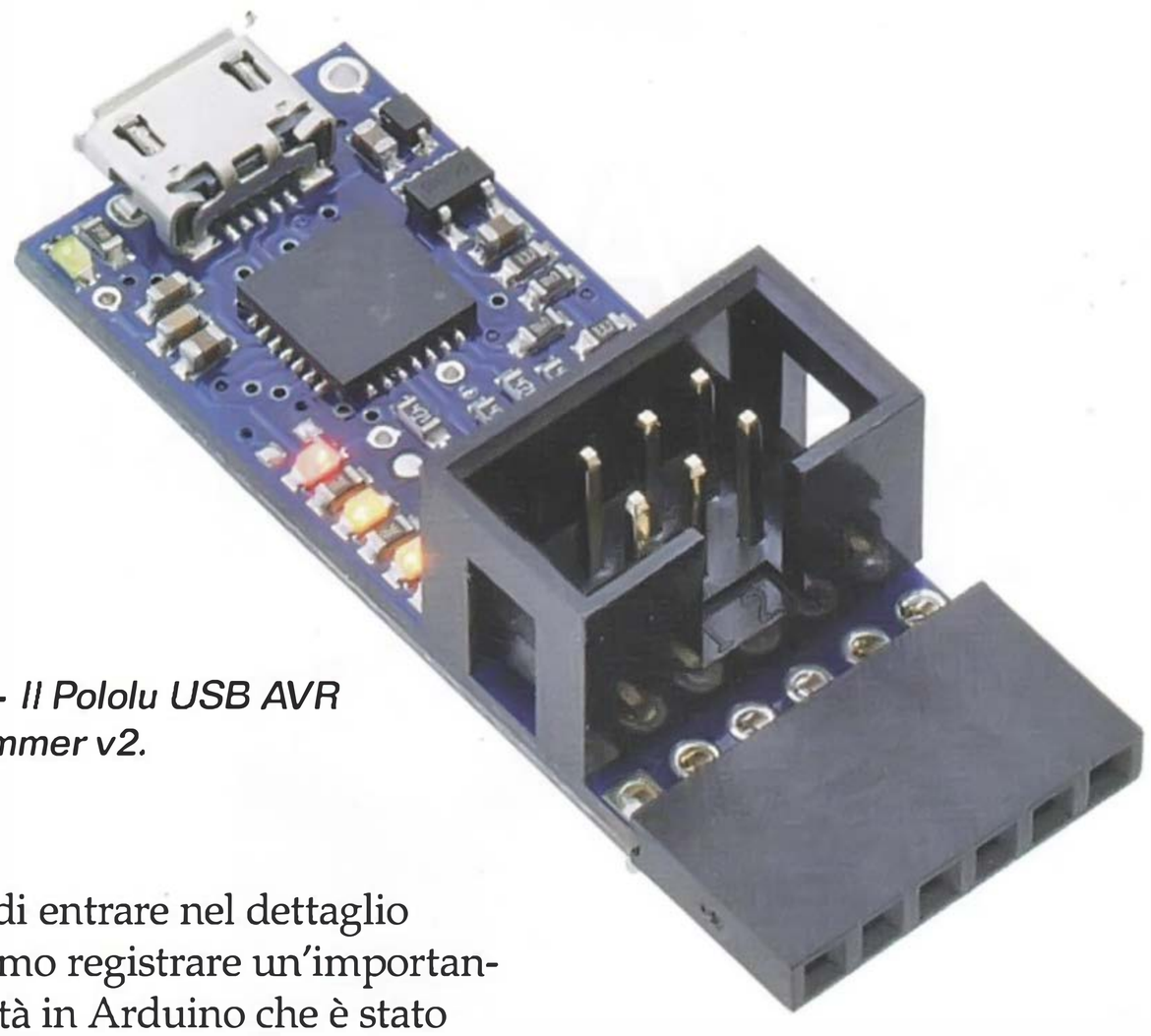


***Fig. 1** - Il Pololu USB AVR Programmer v2.*

Prima di entrare nel dettaglio dobbiamo registrare un'importante novità in Arduino che è stato un grande sospiro di sollievo per tutta la vasta comunità degli utenti e dei makers. Dal comunicato redatto in data 28 luglio 2017 si apprende che BCMI (la compagnia fondata da Massimo Banzi, David Cuartielles, David Mellis e Tom Igoe, cofondatori di Arduino) ha acquisto in quel giorno il 100% della proprietà di Arduino AG la società proprietaria di tutti i marchi Arduino. A seguito di questa acquisizione il Dr. Massimo Banzi è il nuovo Presidente e direttore esecutivo di Arduino. Il Dr. Fabio Violante è il nuovo Amministratore al posto del sig. Federico Musto che perseguirà altre opportunità al di fuori di Arduino AG. Come hanno sottolineato Banzi e Violante è l'inizio di una nuova era che vede nello storico marchio per makers e DIY un'importante punto di riferimento senza più divisioni e liti che hanno compromesso lo sviluppo della piattaforma. Nel momento in cui scriviamo abbiamo inoltre un nuovo ambiente IDE lo 1.8.4 che unifica tutte le schede ed i prodotti creati negli ultimi tempi aprendo inoltre all'integrazione con piattaforme hardware non prodotte direttamente dalla casa madre e dando quindi un'importante porta ed opportunità per chi volesse usare l'IDE di Arduino come piattaforma di sviluppo. Non solo ma alle versioni "tradizionali" si è affiancata anche la versione App adatta ai sistemi con sistema operativo Windows 10. Il nuovo ambiente rende particolarmente semplice aggiungere schede e processori che non sono quelli canonici cui siamo stati abituati dalle board Arduino. Inoltre il nuovo ambiente di sviluppo ci permette di programmare con facilità anche schede self-made, cioè fatte in casa, oppure costruite da produttori diversi da Arduino, come nel nostro caso; non solo: c'è un'importante

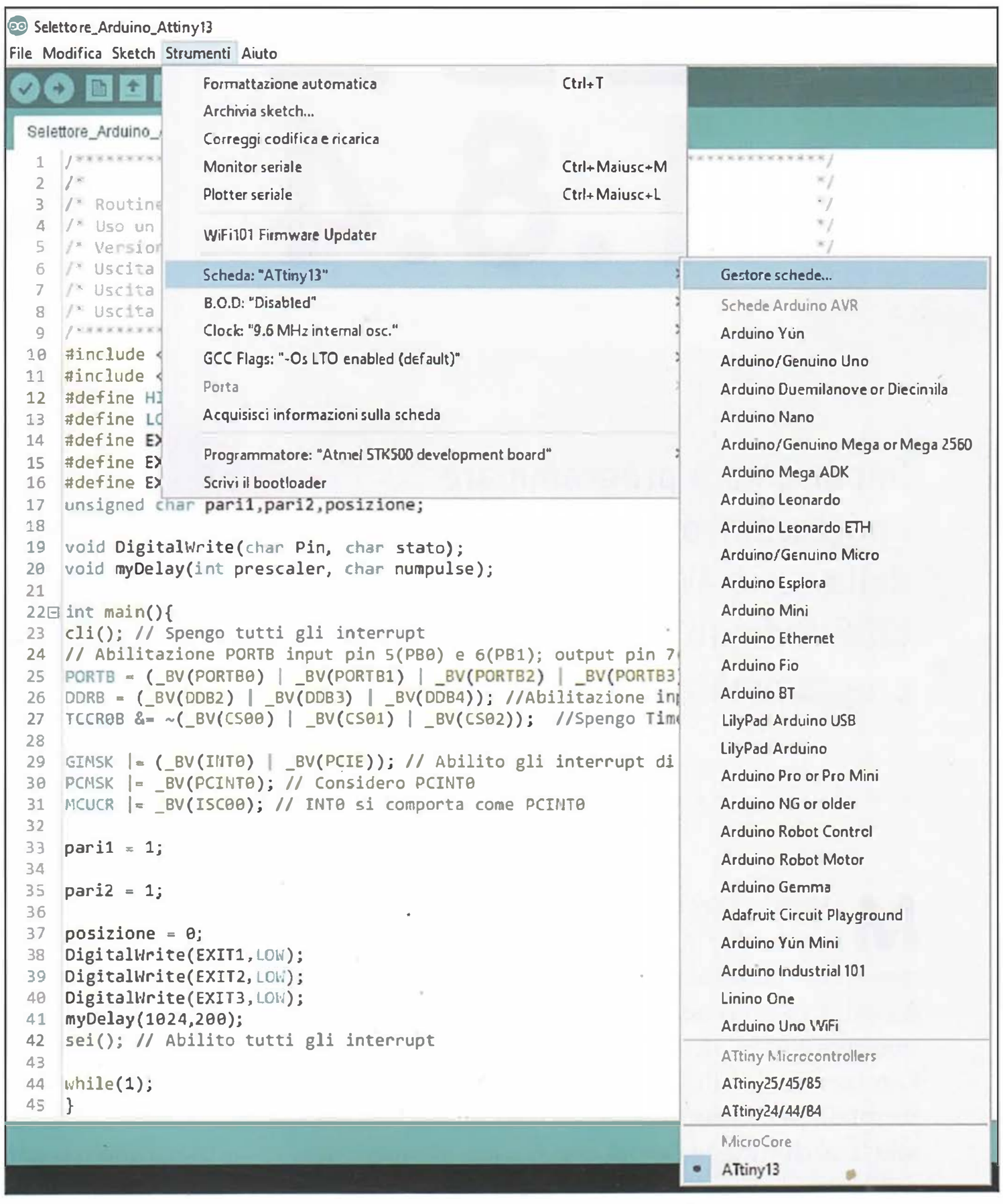


***Fig. 2** - Sottomenu Scheda.*

novità: l'ambiente permette di esportare ora con grande facilità il binario compilato, in modo che lo si possa inviare ad un programmatore hardware o ad un software di programmazione allo scopo di effettuare la produzione in serie.

UTILIZZARE IL NUOVO IDE

Ma vediamo con ordine tutti i passi da svolgere per programmare un AVR in stand-alone tramite la nuova versione dell'IDE Arduino: innanzitutto scarichiamo l'ultima versione disponibile o dal sito *www.arduino.cc* (più precisamente dal link *www.arduino.cc/en/Main/Software*).
Tra le versioni di IDE che si possono scaricare c'è anche la App per Windows10, comoda perché permette di tenere automaticamente aggiornato l'ambiente mano a mano che ci sono dei nuovi rilasci; l'unico neo è che funziona solo su Windows 10 (il funzionamento, lo abbiamo testato, è stabile) e non su Windows 8 o sull'8.1. Una volta scaricato e installato l'ambiente di programmazione Arduino, provvediamo ad installare un programmatore che permette di caricare il nostro programma compilato nel microcontrollore. Scegliamo quindi la nuova versione del programmatore della Pololu, il cui costo è -come la grandezza della scheda- molto ridotto. Sul sito web della Futura Elettronica (*www.futurashop.it*) potrete acquistarlo e scaricare i driver per usare il programmatore in ambiente Windows.
Il programmatore della Pololu (**Fig. 1**), di cui troverete i dettagli in un riquadro in queste pagine, è estremamente comodo; infatti utilizza un cavo USB A/micro B, lo stesso che si usa per un qualunque telefonino/smartphone e viene visto come un programmatore Atmel STK500 e quindi compatibile con tutti gli ambienti di programmazione in uso dall'ambiente Arduino, all'Atmel Studio, all'AVRDudess (che vedremo prossimamente).
A questo punto introduciamo il nuovo sistema di configurazione dei micro/schede di terze parti che si chiama **Gestore Schede**. Il Gestore Schede si trova all'inizio del sottomenu **Scheda** del menu **Strumenti**; come vedete nella **Fig. 2**, in fondo ci sono già due estensioni, *Attiny* e *Microcore*, che servono rispettivamente per gli ATTiny 25/45/85 24/44/84 e l'ATTiny13; prima di essere utilizzate, però, vanno installate.
Quindi, appena si fa clic su Gestore Schede si troveranno già precaricate alcune schede non-core (**Fig. 3**); l'unica cosa installata sono le Arduino AVR Boards, cioè le schede di base (che vengono infatti segnate come INSTALLED). Tutto il resto, no.
Se si fa clic con il pulsante sinistro del mouse sopra le varie bande grigie, la banda selezionata diventa grigio scura e viene attivato il pulsante di installazione a destra ed a sinistra la versione che si vuole installare. Se la scheda è installata apparirà invece il pulsante Rimuovi. Le schede caricate nel gestore schede sono quelle ufficiali, mentre quelle di terze parti si possono caricare sempre dalla stessa scheda, dopo aver impostato una preferenza.
Supponiamo, quindi, di voler caricare le estensioni per gli ATTiny 24/44/84 25/45/85 e quella per l'ATTiny13; per fare questo dobbiamo andare sulle impostazioni del menu **File** e fare clic sui rettangoli sovrapposti a destra a fianco della casella "URL aggiuntive per il Gestore schede" (**Fig. 4**).
Si apre quindi la finestra modale che permette di inserire riga per riga le estensioni e sotto questa finestra appare un link importantissimo: "clicca qui per ottenere una lista di URL per schede non ufficiali".
Facendovi clic si apre nel browser la pagina di GitHub "Unofficial list of 3rd party boards support urls". Ecco: basta fare un copia incolla di quello che interessa (nel nostro caso la MicroCore per l'ATTiny13 e la ATtiny per gli ATTiny 25/45/85 e 24/44/84) in questa finestra, poi cliccare su ok e di nuovo ok anche su quella delle impostazioni; a questo punto in fondo al Gestore Schede appaiono le nostre due nuove

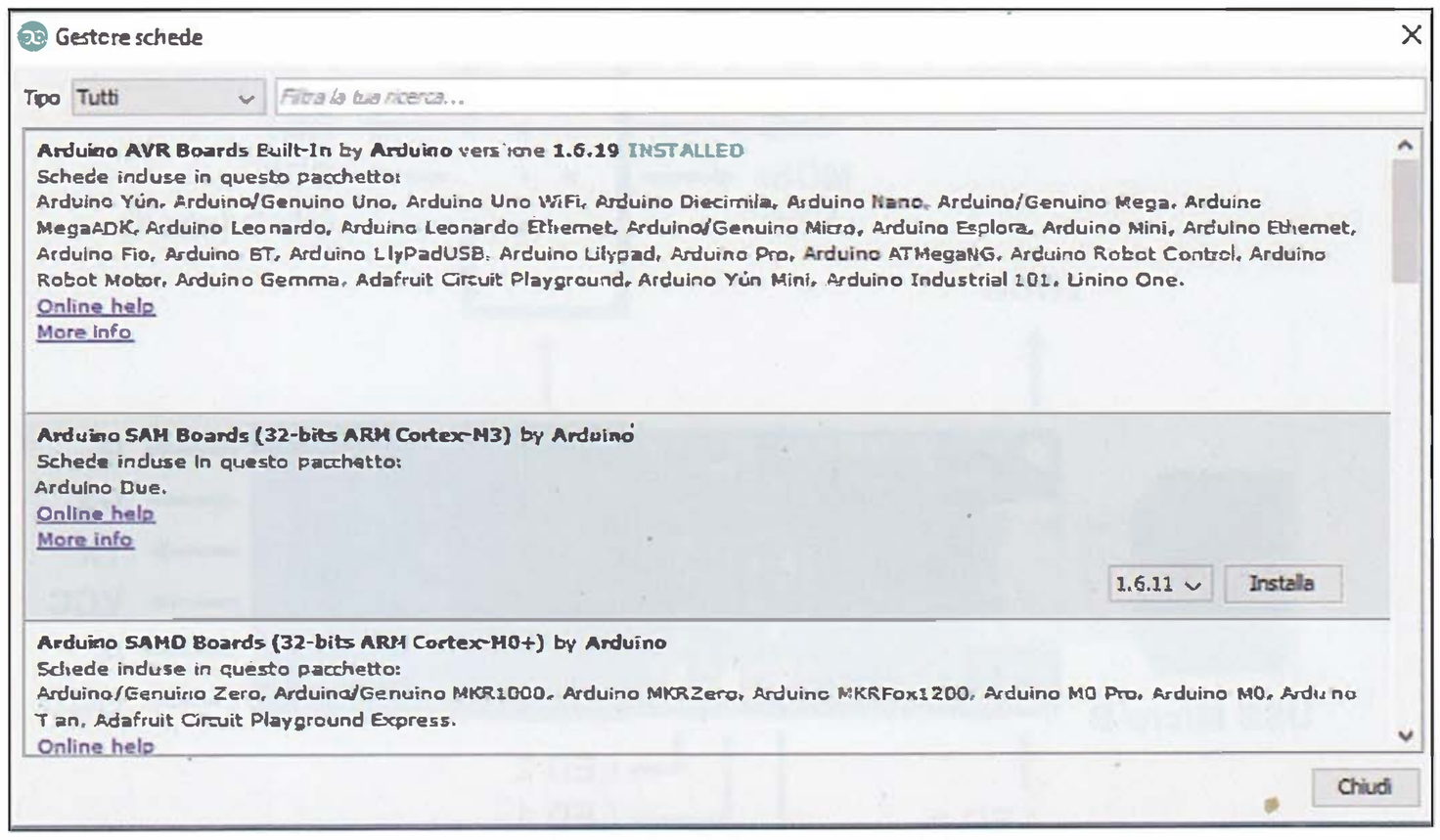


***Fig. 3** - Finestra Gestore Schede.*

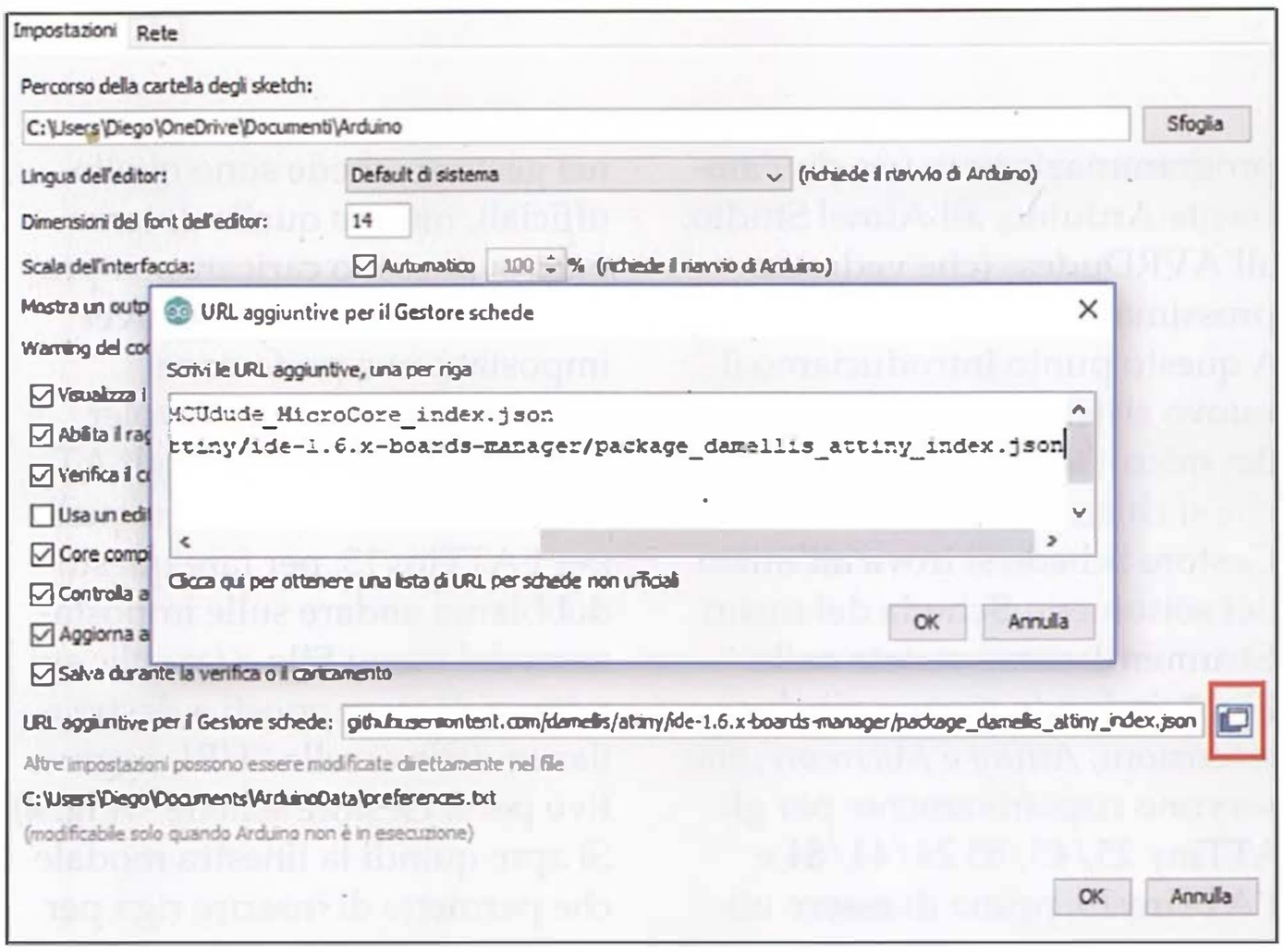


***Fig. 4** - Ricerca schede tramite link web.*

estensioni. Ora basta semplicemente installare il supporto ed il gioco è fatto ed in base alla "Scheda" selezionata si avranno delle opzioni riferite al singolo chip che si vuole usare.
Abbiamo quindi preparato l'ambiente ed ora non ci resta che costruire un programma.
Tuttavia c'è una cosa importantissima da sapere e la si legge su Arduino Playground; tradotta in italiano è la seguente: "L'ambiente di Arduino è basato su microcontrollori Atmel Atmega. Il linguaggio AVR è un ambiente C per programmare i chip Atmel. Molto del codice di Arduino è scritto con le costanti e le funzioni AVR e ci sono molte cose che non sono facili da realizzare con il linguaggio Arduino senza usare il codice AVR. La buona notizia è questa: l'ambiente Arduino usa GCC per compilare il codice e tutte le porte e le variabili registro che si trovano nel datasheet e nei tutorial del codice assembler dell'AVR sono supportate nell'IDE di Arduino". E questa è davvero una bella notizia per chiunque voglia scrivere un programma; una magnifica notizia... Spieghiamoci meglio: normalmente in tutte le pubblicazioni viene scritto che per scrivere uno sketch bisogna utilizzare le due funzioni fondamentali **setup** e **loop**; ebbene, queste due funzioni caricano tutto l'ambiente di Arduino con tutte le sue utilità (o quasi, nel caso dei "piccoli" ATTiny). Tuttavia, se il nostro intento è scrivere un programma stand-alone in C esattamente come faremmo con Atmel Studio, possiamo farlo senza modificare una virgola, a patto di cancellare tutto ciò che viene dall'ambiente Arduino per impostazione predefinita e quindi partire con l'editor completamente in bianco, poi comporre il programma.
Per farvi comprendere il concetto riferitevi al firmware del progetto **Interfaccia con ATTiny per alimentatore** pubblicato nel fascicolo n° 217 (potete scaricare il rispettivo file dalla sezione Download del nostro sito web *www.elettronicain.it*, tra i file del fascicolo n° 217); aprite l'IDE di Arduino, importate il file .ino e qui attivate, dal menu File, l'estensione del MicroCore, quindi

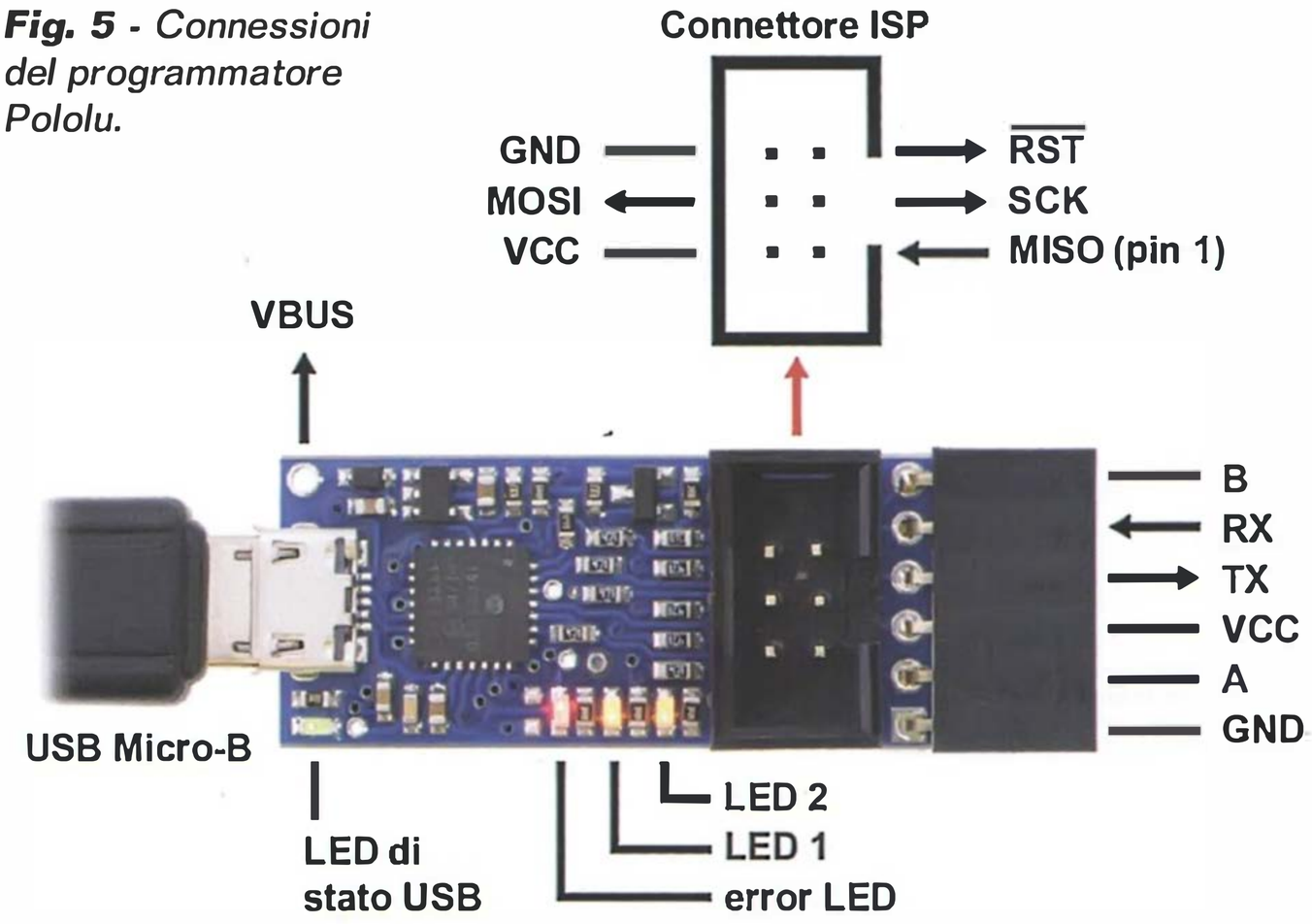


***Fig. 5** - Connessioni del programmatore Pololu.*



Il programmatore USB AVR V2.0 (cod. USBAVRPROGV2) è disponibile presso Futura Elettronica al prezzo di Euro 17,50.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775
http://www.futurashop.it

scegliete la scheda ATtiny13, come spiegato prima. Poi modificate l'opzione del **B.O.D.** impostandola su **Disabled**, copiate il codice cancellando le funzioni setup e loop e fate clic sul pulsante con il segno di "spunta" dell'IDE. Vedrete che il codice si compila senza alcun problema, anche se avete tolto ciò che per il mondo Arduino sarebbe indispensabile. Ecco, quanto accade è la conferma di quanto riportato da AVR Playground.

L'AVR PROGRAMMER ALL'OPERA

A questo punto non ci resta che inserire il codice compilato nella scheda target; per farlo, prendete il vostro programmatore Pololu USB AVR Programmer v2 e configuratelo nell'ambiente Arduino, alla voce Strumenti, come programmatore "Atmel STK500 development board"; ora impostate la porta (normalmente è la COM14) e collegate il cavo (occhio che non ci siano ponticelli che cortocircuitino qualcuno degli ingressi di programmazione a massa) di programmazione al connettore sulla scheda. Alimentate la scheda (il programmatore Pololu per impostazione predefinita non fornisce l'alimentazione al chip; va configurato nel suo programma di setup) e fate clic sul pulsante freccia dell'ambiente Arduino: in qualche secondo si compila e carica.

La **Fig. 5** mostra come si collega il Pololu Usb AVR Programmer v2 al computer e ai pin di programmazione (ICSP nelle schede con micro saldato...) del microcontrollore da programmare.

Ecco fatto: abbiamo appena programmato in C un microcontrollore in modalità stand-alone, senza per forza dover disporre di una scheda Arduino e del relativo bootloader caricato nell'AVR.

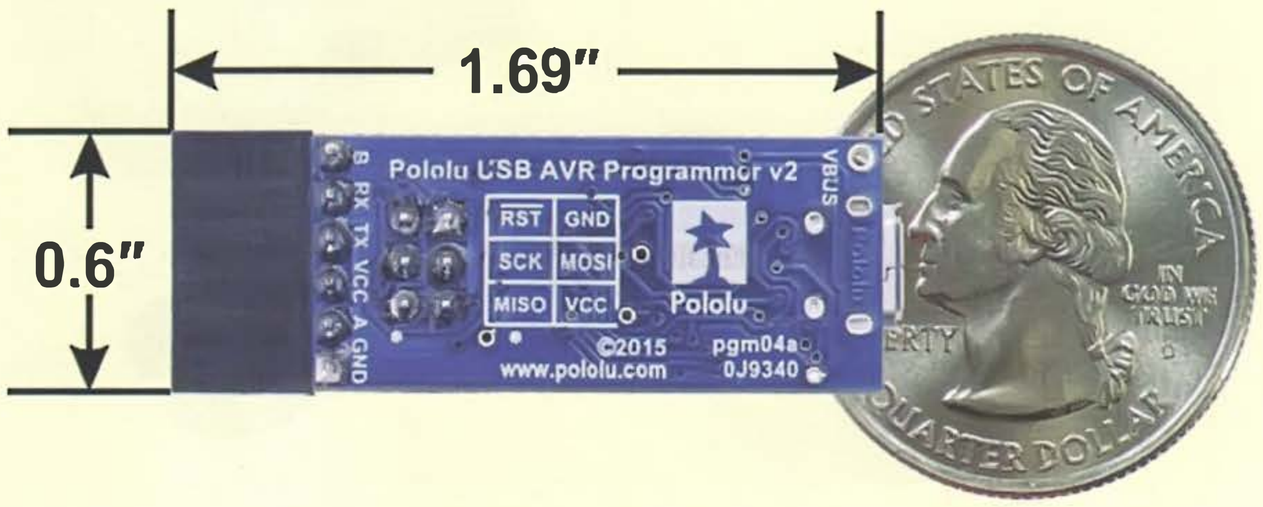
Il programmatore **Pololu**

È adatto ai microcontrollori AVR programmabili tramite interfaccia ISP AVR e può anche essere utilizzato per aggiornare, sostituire o rimuovere il bootloader sulle schede Arduino e sulle schede Pololu A-Star 32U4 compatibili con Arduino. È in grado di funzionare con dispositivi target (il micro o la scheda a microcontrollore da programmare) a 3,3 V e 5 V e può essere configurato per fornire alimentazione al target. Il programmatore si collega alla porta USB del computer tramite un cavo USB A/micro-B e comunica con il software di programmazione, come ad esempio Atmel Studio, AVRDUDE o anche l'IDE Arduino, attraverso una porta COM virtuale utilizzando il protocollo STK500. Si collega al dispositivo target tramite un cavo di programmazione ISP a 6 pin (incluso). Il firmware è aggiornabile. Software di configurazione e driver disponibili per Windows (Vista, 7, 8 e 10), Mac OS X (versione 10.11 e successive) e Linux. Le vecchie versioni di Windows (XP e precedenti) non sono supportate.

Il programmatore raddoppia le porte seriali disponibili via USB, perché una volta installato apre 2 porte COM virtal: una per la comunicazione con il software di programmazione ed un'altra di comunicazione seriale generica. Ciò significa che è possibile programmare un AVR ed eseguire il debug attraverso la porta seriale TTL contemporaneamente senza aprire e chiudere il terminale. Oltre alle linee di trasmissione seriale (TX) e di ricezione (RX), il programmatore consente di utilizzare A e B come linee di handshaking seriale che possono essere configurate usando il nostro software.

I sei piedini del connettore seriale (GND, A, VCC, TX, RX e B) sono disposti similmente alla piedinatura dei comuni cavi FTDI da USB a seriale. Nella configurazione predefinita, il pin B viene utilizzato come uscita DTR (il piedino A è un ingresso inutilizzato) che consente di collegare direttamente il programmatore alle schede Arduino-compatibili e utilizzarlo per caricare programmi tramite un Bootloader seriale. Le principali caratteristiche tecniche del programmatore sono:

- emulazione STK500 attraverso una porta COM virtuale;
- supporta dispositivi a 3,3 V e 5 V;
- incorpora adattatore USB-seriale TTL;
- uscita di clock a 100 kHz;
- pin I/O protetti da una resistenza da 470 Ω;
- dimensioni (mm): 43x15x20;
- peso: 12 grammi.



CONCLUSIONI

In futuro vedremo come farci dare una mano dalle funzioni inserite nel linguaggio di Arduino, ma intanto con le nozioni fornite in questo articolo siamo certi di avervi insegnato come utilizzare un unico ambiente per programmare e caricare il codice nei microcontrollori Atmel AVR. L'occasione è stata buona anche per introdurvi il recentissimo programmatore sviluppato dalla Pololu proprio per i microcontrollori di casa Atmel.

Abbiamo concluso; non ci resta che auguravi buon lavoro!

Tutto è più facile con i nuovi radiocontrolli a 433 MHz!

Controlla facilmente l'irrigazione, le luci di casa, il cancello elettrico, l'impianto d'allarme e molto altro con i nuovi trasmettitori e ricevitori a 1-2-4 canali operanti sulla banda dei 433 MHz. Forniscono un segnale stabile a lunga distanza e un bassissimo consumo di corrente in modalità stand-by. I ricevitori possono essere alimentati a 220VAC o a 12VDC in base al modello.









RICEVITORE

2 CANALI 220VAC
cod. EMY220RX2
€ 16,00

2 CANALI 12VDC
cod. EMY12RX2
€ 16,00

4 CANALI 12VDC
cod. EMY12RX4
€ 19,00

Istruzioni in italiano!

RICEVITORE

1 CANALE 220VAC
cod. EMY220RX1
€ 13,00

1 CANALE 12VDC
cod. EMY12RX1
€ 13,00







TRASMETTITORE

1 CANALE
cod. EMYTX1
€ 11,00



TRASMETTITORE

2 CANALI
cod. EMYTX2 **€ 12,00**

TRASMETTITORE

4 CANALI
cod. EMYTX4 **€ 13,00**

Ai ricevitori va abbinato esclusivamente il trasmettitore con lo stesso numero di canali.

FUTURA ELETTRONICA®
Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Prezzi IVA inclusa.

Caratteristiche tecniche e vendita on-line su: www.futurashop.it

Raspberry Pi

HTML5 web app

Controlliamo Arduino tramite una web app residente su Raspberry Pi, utilizzando jQuery Mobile, PHP5 e MySQL, per gestire un hardware che legge lo stato di 4 sensori e comanda altrettanti relé.



di LUCA BELLAN

Nel fascicolo dello scorso settembre abbiamo presentato ArduTrain WiFi, un suggestivo e tecnologico sistema che permette di controllare e automatizzare plastici LEGO® per mezzo di una web app che funge da pannello di controllo e un hardware composto da Raspberry Pi e Arduino. Visto l'interesse destato dal progetto, specialmente sul piano dello sviluppo della app di controllo, in questo articolo vogliamo estendere il discorso sulla creazione di una web app sia lato client che lato server, per poi fare un approfondimento sull'accoppiata Raspberry Pi-Arduino e sulla comunicazione seriale tra di essi per operare il controllo di un semplice hardware esemplificativo.

Prima di entrare nel dettaglio bisogna distinguere tra una *web app* e un'*app nativa*: la prima è un'applicazione web che non risiede sul dispositivo dell'utente finale, bensì su un server remoto (su Internet o in una rete locale) sgravando l'utente dall'obbligo di installazione di qualsiasi software aggiuntivo; il codice dovrà essere scritto una sola volta e potrà essere eseguito, tramite un browser, su qualsiasi dispositivo, sia esso uno smartphone, un tablet o un Personal Computer.

Se volessimo invece sviluppare un'app nativa per

i tre ambienti più diffusi (Android, iOS e Windows Mobile), dovremmo scrivere tre codici sorgente diversi, con altrettanti ambienti di sviluppo, prestando attenzione ai vari framework disponibili in ogni sistema; oppure potremmo utilizzare un ambiente di sviluppo unico, per esempio Qt (*www.qt.io*), inviandola poi a tre diversi sistemi di approvazione prima di vedere la nostra app pubblicata sui relativi app store.
Lo sviluppo dell'app nativa richiede inoltre molto impegno e risorse ed è anche vincolata agli aggiornamenti che i vari produttori rilasciano per i dispositivi: se dovessero modificare una libreria proprietaria, dovremmo riscrivere il pezzo di codice che vi si interfaccia.
Anche se esistono app native come Blynk (*www.blynk.cc*) che consentono di creare interfacce per controllare i nostri dispositivi IoT, la flessibilità nella creazione dell'interfaccia di controllo viene sacrificata alla semplicità di utilizzo (drag & drop dei controlli e impostazione delle azioni senza scrivere alcuna riga di codice).
Da questa breve, ma doverosa introduzione, deriva che il modo più facile e veloce per realizzare dei piccoli pannelli di controllo per i nostri progetti che girino su qualsiasi dispositivo, è realizzare una web app.
Ci sono molti framework che consentono di creare interfacce web in HTML5, adattive, cross-platform e touch friendly; in questo articolo abbiamo deciso di utilizzare jQuery Mobile (*jquerymobile.com*) perché è molto diffuso tra gli sviluppatori web (quindi è più facile trovare supporto e guide in rete) ed è anche abbastanza semplice da utilizzare per chi ha già qualche nozione base di sviluppo web (HTML, CSS e Javascript).
Come recita la scritta nella home page di questo framework: "*jQuery Mobile is a HTML5-based user interface system designed to make responsive websites and apps that are accessible on all smartphone, tablet and desktop devices.*"; quindi, potremo creare delle interfacce che si adatteranno alle risoluzioni dei vari dispositivi su cui verranno visualizzate: le stesse conterranno dei controlli (pulsanti, caselle di testo, menu a tendina, eccetera) con una risoluzione tale da essere utilizzati con le dita tramite il touch dello schermo, e non solo con il puntatore del mouse sul computer.

L'HARDWARE PER IL TEST

Il progetto di esempio che abbiamo pensato di realizzare consiste in quattro sensori IR (modulini o PIR) e un modulo relé a quattro canali collegati ad Arduino UNO tramite il Sensor Shield; Arduino sarà a sua volta collegato a Raspberry Pi 3 tramite un cavo USB. La web app, che risiederà su Raspberry Pi, ci consentirà di controllare manualmente i relé tramite degli switch ON/OFF, oppure di impostare delle semplici automazioni che verranno trasmesse ad Arduino una volta completate, ad esempio: se il sensore 1 va a livello basso, setta il relé 3 su ON per 10 secondi, poi torna ad OFF.
I componenti per realizzare questo piccolo progetto (facilmente reperibili sul sito web *www.futurashop.it*) sono:

- Raspberry Pi 3;
- microSD 8 GB o superiori;
- alimentatore per Raspberry Pi 5V - 3A;
- Arduino UNO (è possibile utilizzare anche RandA, seguendo questa guida per adattare il software alla nuova versione di Raspbian Jessie: *https://www.open-electronics.org/randa-Raspberry Pi-pi-3-whats-new/*);
- sensor shield;
- modulo relé 4 canali;
- 4 sensori IR di prossimità (oppure 4 sensori PIR);
- 4 prolunghe per servomotori;
- confezione jumper F-F.

Procediamo ora all'assemblaggio dei componenti, riferendoci allo schema dei collegamenti in **Fig. 1**.
Montiamo il Sensor Shield sopra ad Arduino e colleghiamo le prolunghe per servo rispettivamente ai pin 2, 3, 4, 5, facendo attenzione che il cavetto bianco delle prolunghe vada sulla fila di pin S del Sensor Shield.
Colleghiamo all'altro lato delle prolunghe i sensori utilizzando il cavo fornito assieme ad ogni sensore collegando i cavetti rosso e nero ai terminali rosso e nero della prolunga e il cavetto giallo del sensore (DOUT) al terminale bianco.
Utilizzando i jumper F-F procediamo a cablare il modulo relé a 4 canali: i pin G e V della fila 6 andranno collegati rispettivamente su GND e VCC del modulo relé. Invece i canali 1-4 del modulo si connetteranno rispettivamente ai pin S del Sensor Shield 6, 7, 8, 9.

REALIZZIAMO LA WEB APP

Una volta sistemato l'hardware, scarichiamo la versione di Raspbian Jessie Lite più recente (*www.Raspberrypi.org/downloads/raspbian/*), formattiamo la microSD con SDFormatter (*www.sdcard.org/downloads/formatter_4/index.html*) e scriviamo l'immagine su di essa tramite Win32DiskImager (*http://sourceforge.net/projects/win32diskimager/files/latest/down-load*).
Prima di rimuovere la microSD dal computer, salviamoci sopra un file vuoto chiamato "ssh" senza estensione: in questo modo la connessione SSH sarà attiva fin dal primo avvio di Raspberry Pi senza dover collegare monitor, mouse e tastiera per configurarla.
Inseriamo quindi la microSD in

Raspberry Pi, colleghiamo cavo di rete e alimentazione e, attendendo qualche secondo l'avvio del sistema, connettiamoci in SSH dal nostro computer tramite Putty (*https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html*) o MobaXterm (*http://mobaxterm.mobatek.net/download.html*).
Eseguiamo il comando:

sudo raspi-config

per effettuare le configurazioni iniziali del sistema, come l'espansione del file-system e le impostazioni internazionali (lingua, data, nazione del WiFi, ...) e settiamo l'hostname a "webapp". Una volta completata questa procedura non riavviamo, ma apriamo il file per la connessione WiFi con il seguente comando:

sudo nano /etc/wpa_supplicant/wpa_supplicant.conf

Alla fine del file inseriamo le righe seguenti, cambiando i due parametri predefiniti con quelli della nostra rete WiFi:

```
network={
    ssid="nome_rete_wifi"
    psk="password_rete_wifi"
}
```

Chiudiamo e salviamo il file con CTRL+X, poi premendo Y e INVIO: al prossimo riavvio del sistema non occorrerà più il cavo di rete, dato che saremo connessi alla rete WiFi.
Ora passiamo a installare git:

sudo apt-get install git

Posizioniamoci ora nella cartella home dell'utente "pi":

cd /home/pi/

Scarichiamo da GitHub i sorgenti della web app:

git clone https://github.com/open-electronics/webapp.git

Entriamo nella cartella che contiene il file di installazione:

cd webapp/utils/

Settiamo i relativi permessi di esecuzione:

sudo chmod a+x install.sh

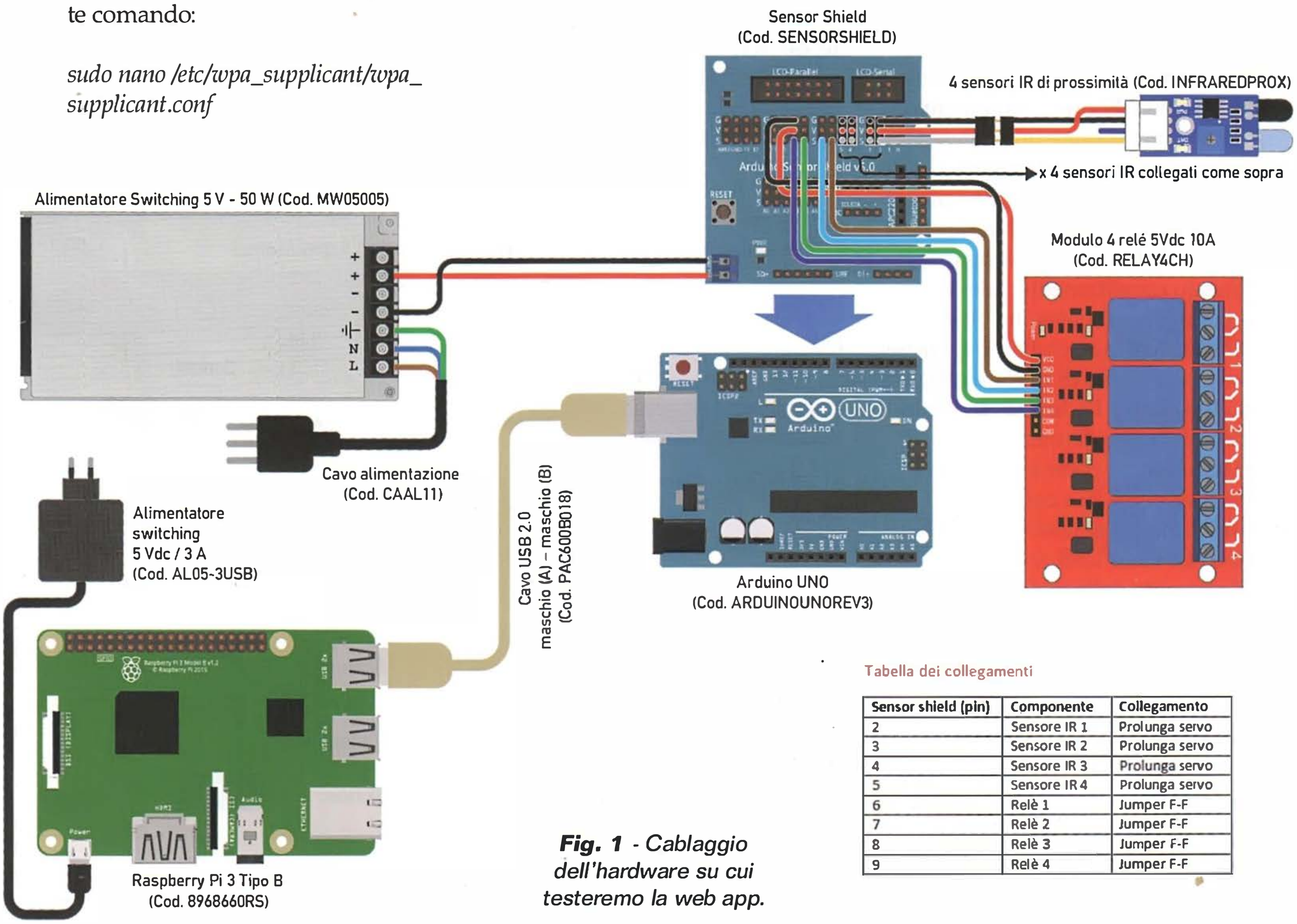


***Fig. 1** - Cablaggio dell'hardware su cui testeremo la web app.*

Tabella dei collegamenti

Sensor shield (pin)	Componente	Collegamento
2	Sensore IR 1	Prolunga servo
3	Sensore IR 2	Prolunga servo
4	Sensore IR 3	Prolunga servo
5	Sensore IR 4	Prolunga servo
6	Relè 1	Jumper F-F
7	Relè 2	Jumper F-F
8	Relè 3	Jumper F-F
9	Relè 4	Jumper F-F

Quindi eseguiamo l'installazione:

sudo bash install.sh

Durante l'esecuzione della procedura verranno lanciati un update ed un upgrade di tutto il software e dei repository del sistema; verranno installati Apache, PHP5 e MySQL (necessari per il funzionamento della web app) e, quando ci verrà richiesto di inserire una password, dovremo inserire "admin" per due volte consecutive.
Verrà poi creato il database della web app con la struttura di base e sarà impostata l'esecuzione automatica ad ogni avvio della procedura PHP che dovrà instaurare la connessione seriale tra la Raspberry Pi e Arduino; successivamente, i sorgenti della web app verranno trasferiti sotto */var/www/html/*.
Alla fine della procedura, il sistema verrà riavviato collegandosi alla rete WiFi precedentemente impostata; tramite un browser, raggiungiamo l'indirizzo *http://webapp/* oppure l'IP della Raspberry Pi e verifichiamo che il menu della web app venga visualizzato correttamente; adesso facciamo clic sul pulsante "System shutdown" e attendiamo qualche secondo prima di poter scollegare la Raspberry Pi dall'alimentazione e dal cavo di rete.
Mettiamo da parte per un istante Raspberry Pi e colleghiamo ora Arduino al nostro computer tramite cavo USB; scarichiamo il sorgente della web app da GitHub sul nostro PC e, tramite l'Arduino IDE (*https://www.arduino.cc/en/main/software*), carichiamo lo sketch "WebappSketch" che possiamo trovare nella cartella "utils".
Adesso connettiamo la Raspberry Pi ad Arduino tramite il cavo USB e diamo alimentazione alla Raspberry Pi stessa, per avviare il nostro sistema.

ANALISI DELLA WEB APP

In seguito all'avvio del sistema, possiamo raggiungere la nostra web app con un browser, come detto in precedenza, tramite l'indirizzo *http://webapp/* oppure con l'indirizzo IP della Raspberry Pi.
Il menu è composto da due pulsanti:

- "Setup", che serve a rinominare tutti i sensori e i relé;
- "Control" (**Fig. 2**), che consente di settare manualmente i relé e di creare delle piccole automazioni; ad esempio, con esso potremo indicare lo stato che deve assumere un sensore per poter attivare un relé per un dato periodo di tempo.

Se un relé dovesse essere associato a un sensore per realizzare un'automazione, non sarebbe più possibile controllarlo manualmente; dovremo quindi disassociarlo per tornare al controllo manuale.
Dalla scheda "Manual" proviamo a settare un relé su ON e poi nuovamente su OFF: dovremo sentire il suono del relé che scatta. Spostiamoci ora nella scheda "Auto" e creiamo una piccola automazione: quando lo stato del primo sensore va a LOW, settiamo il primo relé a ON per 10 secondi.
Lasciamo passare un paio di secondi in modo che la Raspberry Pi trasferisca ad Arduino le istruzioni e, successivamente, passiamo la mano davanti al primo sensore: dovremo sentire il primo relé che va a ON, per poi ritornare a OFF dopo 10 secondi.
Bene, fatto ciò passiamo all'analisi e alla stesura del codice che costituisce la nostra web app; la prima cosa da fare è scaricare Notepad++ (*https://notepad-plus-plus.org/download/v7.4.2.html*) che è un software gratuito e leggero, il quale ci permette di aprire i file sorgente del progetto (che abbiamo scaricato precedentemente da GitHub).
Aprendo la cartella "utils" troviamo lo sketch da caricare su

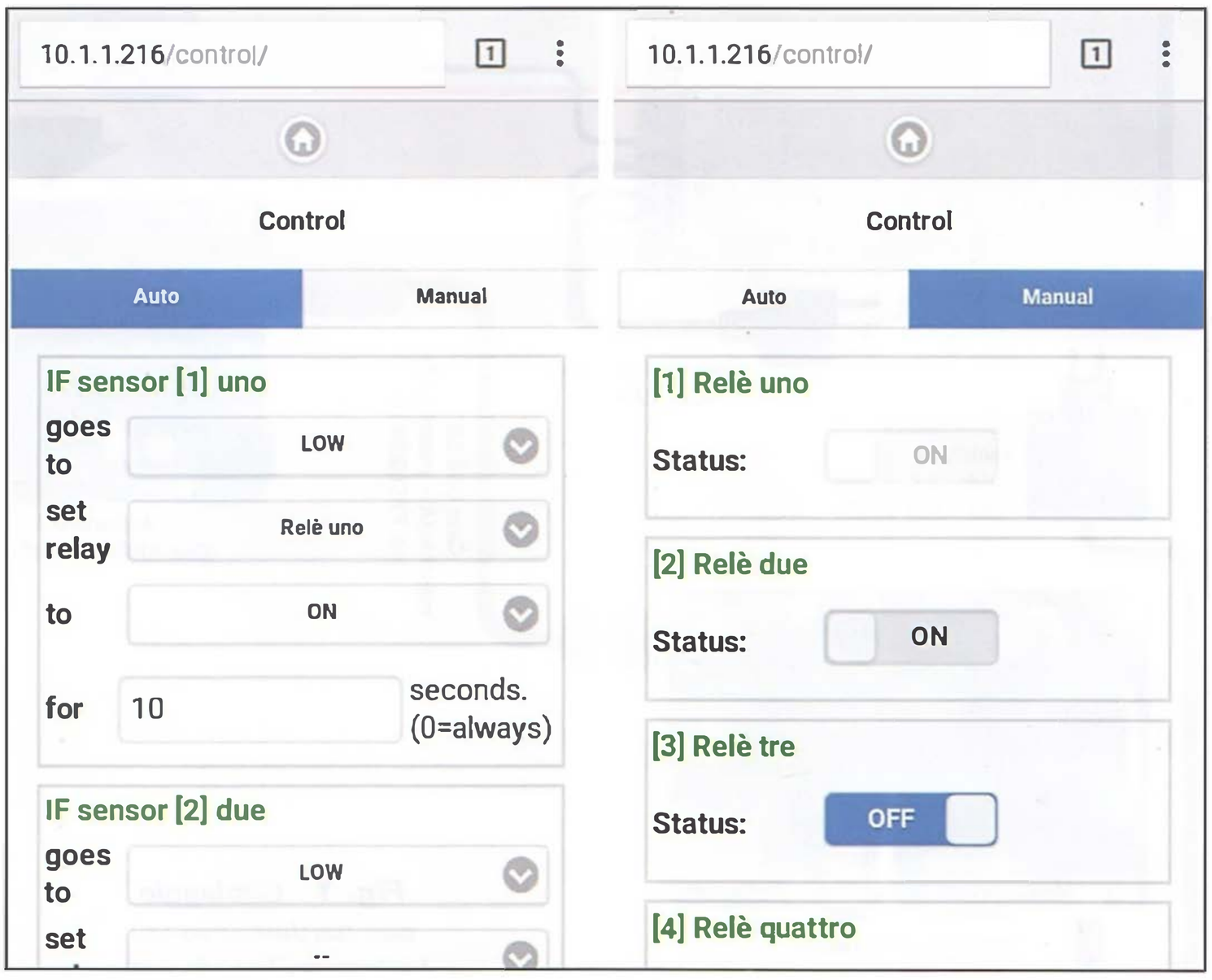


***Fig. 2** - Le due schermate di controllo della web app.*

Tabella: relays

id	name	status
1	Relè uno	0
2	Relè due	1
3	Relè tre	0
4	Relè quattro	0

Tabella: sensors

id	name	eid_relay	target	relay_status	relay_seconds
1	Primo sensore	3	0	1	10
2	Secondo sensore	0	0	0	0
3	Terzo sensore	0	0	0	0
4	Qu rto sensore	0	0	0	0

Fig. 3 *- Le tabelle che compongono il database.*

Arduino, la procedura da lanciare sulla Raspberry Pi per l'installazione di tutto l'ambiente, nonché il file *webapp.sql* che contiene l'installazione del database; il database è composto dalla tabella "relays" che contiene l'identificativo numerico di ogni relé (dall'1 al 4), la rispettiva descrizione e lo stato corrispondente (0 significa OFF e 1 sta per ON).
La tabella "sensors" contiene l'identificativo numerico dei sensori (dall'1 al 4), la loro descrizione, l'eventuale relé associato per creare l'automatismo (0 significa che nessun relé è stato collegato al sensore), il campo target che indica se il sensore si deve attivare (HIGH) o disattivare (LOW) per far partire l'automatismo, lo stato che deve assumere il relé associato e i secondi in cui il relé deve mantenere lo stato indicato prima di tornare allo stato precedente.
Nella **Fig. 3** possiamo notare le due tabelle con il relativo contenuto: nella prima tabella vediamo il relé 2 con lo stato a ON mentre tutti gli altri sono a OFF, mentre nella seconda osserviamo tutti i sensori senza relé associato, tranne il primo, che ha associato il relé 3; in questo caso, quando il sensore andrà nello stato LOW (0), il relé 3 verrà settato su ON (1) per 10 secondi prima di ritornare ad OFF (0).
Nella cartella "common" troviamo tutti quei file e quelle librerie che verranno utilizzati dai vari file che compongono l'applicazione; nel dettaglio, possiamo osservare la cartella di jQuery Mobile per la creazione dell'interfaccia, la classe PHP di connessione al database (che contiene parametri come indirizzo del database, nome del database, utente e password), la classe PHP per la connessione seriale che verrà utilizzata per inviare comandi alla Raspberry Pi e un foglio di stile CSS.
Tornando alla cartella radice "webapp", possiamo trovare il file *background.php* di cui spiegheremo più avanti e il file **index.php** che si occupa di indirizzare l'utente al menu, nel caso in cui avesse digitato nella barra degli indirizzi del proprio browser *http://webapp e non http://webapp/menu*.
Entriamo nella cartella "menu" e apriamo il file *index.php* con Notepad++: questa è la prima pagina della nostra web app; nella parte superiore del codice, tra i tag *<head>* e *</head>*, troviamo il titolo della pagina, il richiamo ai fogli di stile e alle librerie di jQuery Mobile, oltre che una funzione Javascript che invierà, quando invocata, il comando di spegnimento alla nostra Raspberry Pi.
Nella sezione racchiusa tra i tag *<body>* e *</body>* troviamo invece tre link: il primo rimanda alla sezione "Setup", il secondo rimanda alla sezione "Control" e il terzo serve per spegnere Raspberry Pi invocando la funzione Javascript "shutdown".
Come avrete notato, la formattazione dei link non è quella classica (del semplice testo in blu sottolineato), ma risultano dei veri e propri pulsanti: questo è reso possibile grazie alle classi contenute in jQuery Mobile che trasformano gli elementi della pagina web; considerando, ad esempio, il primo link "Setup" possiamo trovare nella proprietà "class":

- ui-btn: trasforma l'elemento in un pulsante;
- ui-icon-gear: inserisce nel pulsante l'icona di un ingranaggio (l'elenco delle icone è disponibile all'indirizzo Internet: *https://api.jquerymobile.com/icons/*);
- ui-btn-icon-right: allinea a destra l'icona contenuta nel pulsante.

Nel file menu_controller.php viene ricevuto il comando "shutdown" inviato dalla funzione Javascript e, tramite un comando Linux, viene spenta la board Raspberry Pi.
Spostiamoci ora nella cartella "setup" e apriamo il file index.php per analizzare una pagina più complessa: iniziando subito dal tag *<body>* possiamo trovare un link contenente l'icona di una casa che riporta al menu (la pagina home) e il titolo della sezione in cui ci troviamo contenuto in un * *; più sotto troviamo l'elemento "tabs" ridisegnato anch'esso da jQuery Mobile.
Quest'ultimo elemento permette la creazione di più sezioni (tabs) nella stessa pagina che verranno visualizzate una alla volta selezionando la relativa intestazione; il * *

```
<div data-role="tabs" id="tabs">
	<div data-role="navbar">
		<ul>
			<li><a class = "ui-btn-active" href="#SensorBox" data-ajax="false">Sensors</a></li>
			<li><a href="#RelayBox" data-ajax="false">Relays</a></li>
		</ul>
	</div>
	<div id="SensorBox" class="ui-body-d ui-content">
		<table class = "Box">
			<tr><th colspan = "2" style = "color: green;">Sensor 1</th></tr>
			<tr><td><b>Name:</b></td><td><input type="text" id = "SensorName_1" value="" maxlength="30"></td></tr>
		</table>
		<table class = "Box">
			<tr><th colspan = "2" style = "color: green;">Sensor 2</th></tr>
			<tr><td><b>Name:</b></td><td><input type="text" id = "SensorName_2" value="" maxlength="30"></td></tr>
		</table>
		<table class = "Box">
			<tr><th colspan = "2" style = "color: green;">Sensor 3</th></tr>
			<tr><td><b>Name:</b></td><td><input type="text" id = "SensorName_3" value="" maxlength="30"></td></tr>
		</table>
		<table class = "Box">
			<tr><th colspan = "2" style = "color: green;">Sensor 4</th></tr>
			<tr><td><b>Name:</b></td><td><input type="text" id = "SensorName_4" value="" maxlength="30"></td></tr>
		</table>
	</div>
	<div id="RelayBox" class="ui-body-d ui-content">
		<table class = "Box">
			<tr><th colspan = "2" style = "color: green;">Relay 1</th></tr>
			<tr><td><b>Name:</b></td><td><input type="text" id = "RelayName_1" value="" maxlength="30"></td></tr>
		</table>
		<table class = "Box">
			<tr><th colspan = "2" style = "color: green;">Relay 2</th></tr>
			<tr><td><b>Name:</b></td><td><input type="text" id = "RelayName_2" value="" maxlength="30"></td></tr>
		</table>
		<table class = "Box">
			<tr><th colspan = "2" style = "color: green;">Relay 3</th></tr>
			<tr><td><b>Name:</b></td><td><input type="text" id = "RelayName_3" value="" maxlength="30"></td></tr>
		</table>
		<table class = "Box">
			<tr><th colspan = "2" style = "color: green;">Relay 4</th></tr>
			<tr><td><b>Name:</b></td><td><input type="text" id = "RelayName_4" value="" maxlength="30"></td></tr>
		</table>
	</div>
</div>
```

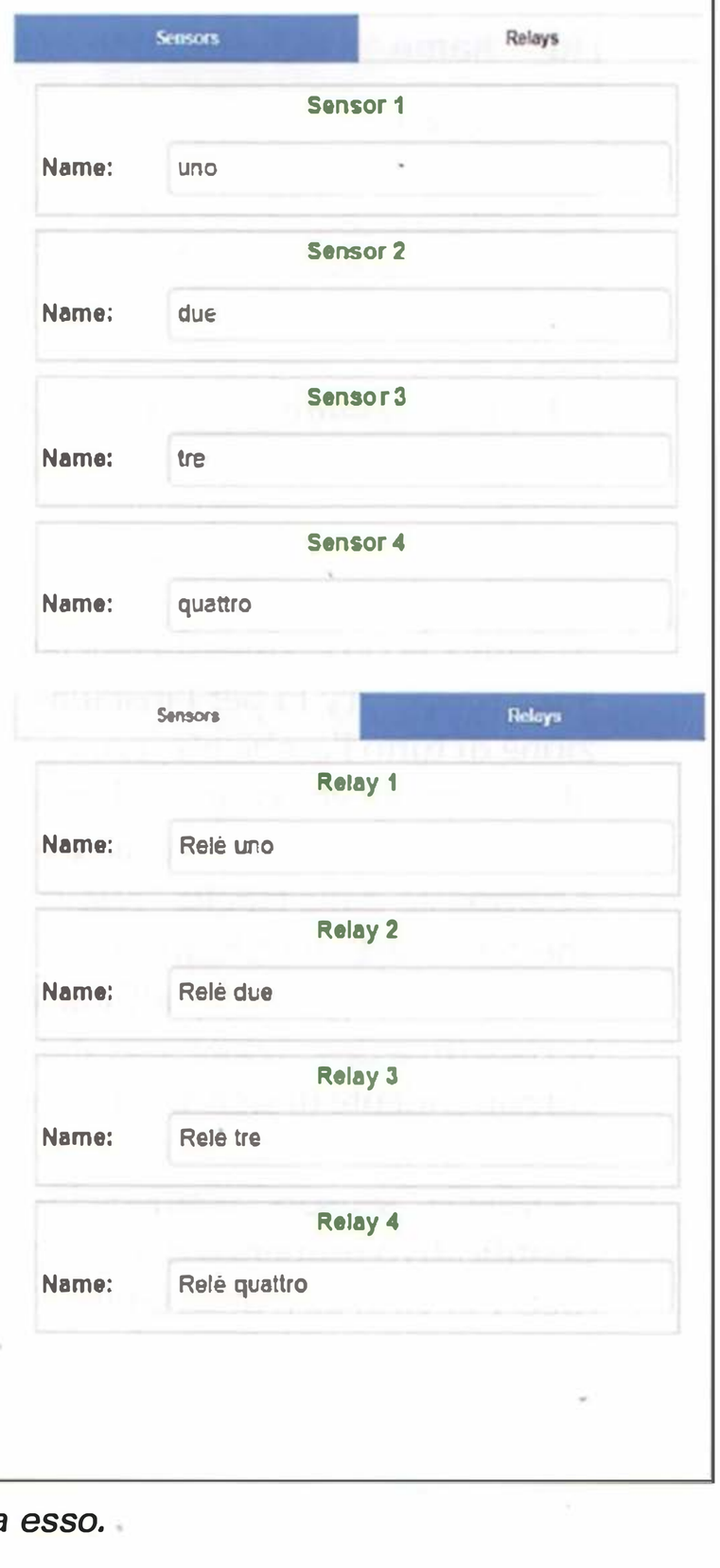


***Fig. 4** - Il codice HTML e la pagina web risultante da esso.*

che ha ruolo di "navbar" (barra di navigazione) contiene le intestazioni di tutte le sezioni: ogni intestazione punta ad un * * specifico dove sono ospitati i contenuti. Nel nostro esempio abbiamo la voce "Sensors" che punta al * * "SensorBox" e la voce Relays che punta al * * "RelayBox"; notiamo che nel link della prima intestazione è stata inserita la classe "ui-btn-active" per attivare il contenuto della prima sezione subito dopo il caricamento della pagina.
Nei due * * successivi (SensorBox e RelayBox) sono elencati i quattro sensori e gli altrettanti relé: a fianco di ognuno di essi è presente una casella di testo nella quale è possibile inserire la descrizione dell'elemento; nella **Fig. 4** possiamo notare le due tab generate dal codice HTML che stiamo analizzando.
Torniamo ora nella parte superiore del codice, tra i tag *<head>* e *</head>*: possiamo notare che le funzioni "loadSensors" e "loadRelays" vengono lanciate non appena la pagina è stata caricata completamente; queste due funzioni eseguono una chiamata asincrona (cioè senza ricaricare la pagina) al file setup_controller.php che risiede sul server, questo file ha il compito di prelevare e restituire alla pagina web l'identificativo e la descrizione di ogni sensore e relé dalle tabelle del database per poi aggiornare le relative caselle di testo che contengono la descrizione degli elementi.
Questo fa in modo che, se avessimo già inserito delle descrizioni, uscendo e rientrando nella pagina queste vengano subito ricaricate e messe al posto giusto.
Durante l'aggiornamento delle caselle di testo, viene associato anche un evento (*focusout*) ad ognuna di esse, che richiama la funzione "updateSensor" oppure la funzione "updateRelay"; in pratica, quando scriviamo dentro a una casella di testo e, successivamente, passiamo a quella dopo o facciamo click su qualsiasi altra parte della pagina web, la casella perde il "focus" e viene richiamata la funzione che preleva la descrizione aggiornata e la invia, sempre tramite chiamata asincrona, al file PHP per l'aggiornamento del database.
Usciamo dalla cartella "setup" per entrare nella cartella "control": iniziamo sempre dal tag *<body>* del

file index.php saltando, per ora, la prima parte del codice che contiene le funzioni Javascript; possiamo notare la navbar che contiene gli elementi "Auto", che punta al <*div*> "SensorBox" e "Manual", che punta al <*div*> "RelayBox".
Il <*div*> "SensorBox" contiene i quattro box dei sensori: ogni box ha una <*select*> (un menu a tendina) per selezionare lo stato target del sensore, una per selezionare il relé di cui si andrà a modificare lo stato, lo stato del relé (ON/OFF) che verrà settato e i secondi in cui il relé manterrà lo stato impostato (0 significa "per sempre").
Tutti gli elementi hanno la funzione "updateSensor" associata ad un loro evento (come "onChange" oppure "onfocusout") che permette di inviare al server le eventuali modifiche che vengono effettuate su di essi.
Il secondo <*div*> contiene invece l'elenco dei quattro relé e, a fianco di ognuno, troviamo una <*select*> che ci permette di impostare manualmente lo stato del relé: questa <select> verrà trasformata da jQuery Mobile in un comodo interruttore che potremo far scorrere a destra o a sinistra per attivare o disattivare il relé; anche questi elementi hanno la funzione Javascript "setRelay" che permette di inviare al server il nuovo stato del relé appena viene rilevata una modifica.

LE FUNZIONI JAVASCRIPT

Ora che abbiamo un'idea della struttura della pagina, possiamo tornare nella parte superiore del codice per analizzare le funzioni Javascript: possiamo notare sempre le due funzioni ("loadSensors" e "loadRelays") che vengono eseguite alla fine del caricamento della pagina ma, questa volta, le analizzeremo separatamente.
La prima funzione ha il compito, tramite una chiamata asincrona al file *control_controller.php*, di popolare tutti gli elementi della sezione "Auto" con i dati salvati provenienti dal database, quindi, ad ogni sensore imposta la descrizione, lo stato target, l'eventuale relé associato, lo stato da impostare al relé e i secondi di mantenimento dello stato.
Con il comando jQuery *$("....").selectmenu("refresh")*; il classico elemento <*select*> "menu a tendina" acquisisce la proprietà adattiva (per essere visualizzato nel migliore dei modi con diverse risoluzioni) e una migliore usabilità sui dispositivi mobili.
La funzione "loadRelays" ha invece il compito di aggiornare il nome e lo stato di tutti i relé recuperando queste informazioni dal server sempre tramite una chiamata POST asincrona: se il relé dovesse essere associato ad un sensore, dovremo impedire la modifica manuale del suo stato, per fare questo controlliamo se la variabile "sensor" ci viene restituita vuota (in questo caso sblocchiamo il controllo manuale) oppure piena (per bloccare il controllo manuale tramite la disabilitazione dell'interruttore).
Come abbiamo visto, jQuery Mobile esegue una trasformazione degli elementi HTML aggiungendo del CSS e delle funzioni Javascript; in **Fig. 5** possiamo confrontare alcuni elementi HTML con e senza l'applicazione di questo framework, ulteriori elementi da posizionare nelle pagine web li possiamo trovare a questo indirizzo: *http://demos.jquerymobile.com/1.4.1/*.

I FILE PHP

Dopo aver analizzato le pagine della web app che verranno eseguite sul dispositivo dell'utente, spostiamoci sul server, quindi sulla Raspberry Pi, per chiarire meglio il funzionamento dei file PHP "*_controller.php"; questi file ricevono chiamate dalle pagine della web app tramite le funzioni Javascript che abbiamo visto precedentemente, ogni chiamata arriva con un comando ben preciso (possiamo notare la variabile PHP "$Command"), a questo punto il codice compie determinate operazioni come l'estrazione dei dati dalle tabelle del database (e la restituzione degli stessi alla pagina web) oppure l'aggiornamento dei dati contenuti nel database. Prendiamo come esempio la pagina setup_controller.php: alla ricezione del comando "loadSensors" viene eseguita una query sulla tabella "sensors" che estrae identificativo e descrizione di ogni sensore; un ciclo *while* scorre tutti i risultati della query e i dati vengono inseriti in un array multidimensionale ($response) che viene poi codificato in JSON (*www.json.org*) per poi essere restituito alla funzione Javascript

Elemento	HTML	HTML + jQuery Mobile
Pulsante	Click Me!	Click Me!
Select	LOW	LOW
Checkbox	ON	Off
Slider		

***Fig. 5** - Alcuni elementi creati da semplice HTML a confronto con gli elementi di jQuery Mobile.*

Listato 1

```
//  PIN
#define PIN_SENS_1 2
#define PIN_SENS_2 3
#define PIN_SENS_3 4
#define PIN_SENS_4 5
#define PIN_RELAY_1 6
#define PIN_RELAY_2 7
#define PIN_RELAY_3 8
#define PIN_RELAY_4 9

//  Serial vars
char inData[100];
char inChar = -1;
byte index = 0;
byte commandPos = -1;
int command[6];

//  Auto vars
byte AutoSensorTarget[4] = {0, 0, 0, 0};
byte AutoRelay[4] = {0, 0, 0, 0};
byte AutoRelayStatus[4] = {0, 0, 0, 0};
int AutoRelaySeconds[4] = {0, 0, 0, 0};
byte AutoTimerActive[4] = {0, 0, 0, 0};
unsigned long AutoTimer[4] = {0, 0, 0, 0};

void setup() {
  //  Start serial
  Serial.begin(9600);
  //  Initialize components
  pinMode(PIN_SENS_1, INPUT_PULLUP);
  pinMode(PIN_SENS_2, INPUT_PULLUP);
  pinMode(PIN_SENS_3, INPUT_PULLUP);
  pinMode(PIN_SENS_4, INPUT_PULLUP);
  pinMode(PIN_RELAY_1, OUTPUT);
  pinMode(PIN_RELAY_2, OUTPUT);
  pinMode(PIN_RELAY_3, OUTPUT);
  pinMode(PIN_RELAY_4, OUTPUT);
}

void loop() {
  checkSerial();
  checkSensors();
  manageTimers();
}

void checkSerial() {
  if(Serial.available() > 0) {
    while (Serial.available() > 0) {
      inChar = Serial.read();
      if(inChar != '@'&& inChar != '#') {
        inData[index] = inChar;
        index++;
      } else if(inChar=='#') {
        commandPos++;
        command[commandPos] = atoi(inData);
        resetSerialString();
      } else if(inChar == '@') {
        switch(command[0]) {
          case 0:
            setRelay(command[1], command[2]);
          break;
          case 1:
            setSensor(command[1], command[2], command[3], command[4], command[5]);
          break;
        }
        commandPos = -1;
        resetSerialString();
      }
    }
  }
}

void checkSensors() {
  if(digitalRead(PIN_SENS_1) == AutoSensorTarget[0] && AutoTimerActive[0]
== 0 && AutoRelay[0] != 0) {
    setRelay(AutoRelay[0], AutoRelayStatus[0]);
    if(AutoRelaySeconds[0] != 0) {
      AutoTimer[0] = millis();
      AutoTimerActive[0] = 1;
    }
  }
```

della pagina web che ha effettuato la chiamata.
Alla ricezione del comando "updateSensor" viene eseguita una query di update alla tabella "sensors" per aggiornare la descrizione di un sensore, che restituisce l'identificativo dello stesso e la nuova descrizione da sostituire a quella già presente nella tabella. Il file PHP *background.php* viene lanciato all'avvio del sistema tramite il demone *systemd*; questo file si occupa di instaurare la connessione seriale tra Raspberry Pi e Arduino tramite una porta, che solitamente è la "/dev/ttyACM0". Dopo aver aperto la connessione seriale tramite l'apposita libreria, viene creato un ciclo infinito tramite l'istruzione "while(1 == 1) {"; in questo ciclo vengono letti continuamente tutti i campi delle tabelle "relays" e "sensors": se anche solo un campo dovesse cambiare, verrebbe immediatamente inviato il comando via seriale ad Arduino per aggiornare lo stato dei componenti collegati ad esso. Per esempio, quando viene rilevato un cambiamento nello stato di un relé, il nuovo stato viene subito inviato ad Arduino tramite la funzione PHP "sendMessage" assieme all'identificativo del relé in questione; Arduino riceverà e processerà il comando ed infine setterà fisicamente il relé con il nuovo stato.
La stessa cosa avviene per i sensori: se dovesse cambiare uno dei valori (target del sensore, relé associato, stato da settare al sensore e secondi di durata) la procedura invierebbe subito un comando in seriale con tutte le nuove informazioni per aggiornare le variabili dello sketch che gira su Arduino.

LO SKETCH DI ARDUINO

Passiamo adesso all'ultima parte, ossia il codice Arduino, che trovate nel **Listato 1**. Premettiamo che per mettere in comunicazione tra loro

Raspberry Pi e Arduino abbiamo istituito un protocollo molto semplice; il formato dei comandi per settare i relé è il seguente:

0#[identificativo relé]#[stato relé]#@.

Il primo campo è sempre fisso a 0 e identifica il tipo "relé", l'identificativo va da 1 a 4 mentre lo stato può essere 0 (OFF) oppure 1 (ON). Il formato dei comandi per associare un relé ad un sensore è invece il seguente:

1#[identificativo sensore]#[identificativo relé associato]#[stato target del sensore]#[stato da impostare al relé]#[secondi di mantenimento dello stato]#@;

in questo caso il primo campo è sempre fisso a 1 per identificare il tipo "sensore", il secondo e il terzo campo possono assumere i valori dall'1 al 4 e rappresentano rispettivamente il sensore e il relé associato, il quarto campo è lo stato in cui il sensore farà partire l'automazione (0 LOW, 1 HIGH), il quinto campo contiene lo stato il relé e l'ultimo campo contiene i secondi in cui il relé manterrà lo stato impostato.
Come separatore per ogni campo abbiamo utilizzato il cancelletto ("#"): in questo modo lo sketch riconosce la fine del contenuto di un campo e trasferisce il valore ricevuto in una variabile specifica; come terminatore di comando abbiamo utilizzato una chiocciola ("@") che serve per fare eseguire il comando quando tutte le variabili sono popolate.
Analizziamo ora nel dettaglio il funzionamento dello sketch di Arduino aprendo il file *WebappSketch.ino*, riportato nel **Listato 1**, contenente il codice che riceve ed esegue i comandi di Raspberry Pi tramite un protocollo ad hoc).

Listato 1 (segue)

```
  if(digitalRead(PIN_SENS_2) == AutoSensorTarget[1] && AutoTimerActive[1]
== 0 && AutoRelay[1] != 0) {
    setRelay(AutoRelay[1], AutoRelayStatus[1]);
    if(AutoRelaySeconds[1] != 0) {
      AutoTimer[1] = millis();
      AutoTimerActive[1] = 1;
    }
  }
  if(digitalRead(PIN_SENS_3) == AutoSensorTarget[2] && AutoTimerActive[2]
== 0 && AutoRelay[2] != 0) {
    setRelay(AutoRelay[2], AutoRelayStatus[2]);
    if(AutoRelaySeconds[2] != 0) {
      AutoTimer[2] = millis();
      AutoTimerActive[2] = 1;
    }
  }
  if(digitalRead(PIN_SENS_4) == AutoSensorTarget[3] && AutoTimerActive[3]
== 0 && AutoRelay[3] != 0) {
    setRelay(AutoRelay[3], AutoRelayStatus[3]);
    if(AutoRelaySeconds[3] != 0) {
      AutoTimer[3] = millis();
      AutoTimerActive[3] = 1;
    }
  }
}

void manageTimers() {
  for(int i=0;i<4;i++) {
    if(AutoTimerActive[i] == 1) {
      if(millis() >= AutoTimer[i] + (AutoRelaySeconds[i] * 1000) ) {
        AutoTimer[i] = 0;
        AutoTimerActive[i] = 0;
         if(AutoRelayStatus[i] == 1) { setRelay(AutoRelay[i], 0); } else {
setRelay(AutoRelay[i], 1); }
      }
    }
  }
}

void setRelay(int address, int value) {
  switch(address) {
    case 1:
      digitalWrite(PIN_RELAY_1, value);
    break;
    case 2:
      digitalWrite(PIN_RELAY_2, value);
    break;
    case 3:
      digitalWrite(PIN_RELAY_3, value);
    break;
    case 4:
      digitalWrite(PIN_RELAY_4, value);
    break;
  }
}

void setSensor(int sensor, int relay, int target, int stat, int seconds) {
  sensor -= 1;
  AutoSensorTarget[sensor] = target;
  AutoRelay[sensor] = relay;
  AutoRelayStatus[sensor] = stat;
  AutoRelaySeconds[sensor] = seconds;
}

void resetSerialString() {
  index = 0;
  for(int i=0;i<100;i++) {inData[i] = '\0';}
}
```

Dopo la definizione dei pin e della modalità di ciascuno di essi (nella funzione "setup"), nel loop possiamo notare tre funzioni di seguito descritte.

- **checkSerial**: controlla costantemente la ricezione di nuovi comandi provenienti da Raspberry Pi tramite la porta seriale, appena riceve una stringa incomincia ad accodare, carattere per carattere, fino a quando non trova un carattere separatore; in questo caso "scarica" la stringa

14ª FIERA REGIONALE DELL'ELETTRONICA SCANDIANO
(REGGIO EMILIA)
21/22 OTTOBRE
CENTRO FIERISTICO 2017



MERCATINO DELLE PULCI RADIOAMATORIALI
TELEFONIA - COMPONENTISTICA
COMPUTER - HI-FI CAR - RADIANTISMO
CB e OM - VIDEOREGISTRAZIONE
ORARI
Sabato 21 Ottobre 9 - 18,30
Domenica 22 Ottobre 9 - 18

che si è formata in una variabile seguendo un ordine ben preciso. Alla ricezione del carattere terminatore esegue il comando controllando il tipo (il primo campo); se è un relé chiama la funzione "setRelay" che si occupa fisicamente di modificare lo stato del relé, se invece è un sensore, chiama la funzione "setSensor" che aggiorna gli array dedicati ai sensori con le impostazioni contenute nel comando.

- **checkSensors**: verifica l'attivazione di ciascun sensore in base allo stato target impostato; se un sensore dovesse attivarsi verrebbero effettuati diversi controlli e verrebbe infine fatto partire il timer per l'automazione di accensione e di spegnimento del relé associato;
- **manageTimers**: questa funzione verifica lo stato dei timer di ciascun sensore tramite la funzione *millis()*; nel caso in cui un timer fosse attivo e i secondi impostati fossero trascorsi, la procedura invertirebbe lo stato del relé e spegnerebbe il timer.

CONCLUSIONI

Quello che abbiamo voluto presentarvi in queste pagine è un piccolo progetto che può essere modificato a vostro piacimento per controllare diversi componenti elettronici connessi ad Arduino e Raspberry Pi tramite una web app; anche il database è stato strutturato per rendere facile l'inserimento di altri oggetti da controllare. Per personalizzarlo occorre solamente inserire nuovi record, ricordandosi di mantenere l'identificativo univoco. Buon lavoro!



per il MATERIALE

Il materiale per realizzare questo progetto si può acquistare presso Futura Elettronica. Il Sensor Shield per Arduino (cod. SENSORSHIELD) costa Euro 11,00, il modulo 4 relè 5VDC/10A (cod. RELAY4CH) è disponibile a Euro 13,00, la board Arduino UNO REV3 (cod. ARDUINOUNOREV3) al prezzo di Euro 24,50, la board Raspberry Pi 3 Tipo B (cod. 8968660RS) a Euro 44,00, il sensore IR di prossimità (cod. INFRAREDPROX) costa Euro 5,00. L' alimentatore switching 5VDC/50W (cod. MW05005) al prezzo di Euro 15,00, l'alimentatore micro USB 5V/3A (cod. AL05-3USB) a Euro 10,50, il cordone alimentatore 10A/250V 3PO (cod. CAAL/11) è disponibile a Euro 2,90 e il cavo USB 2.0 M(A)/M(A) da 1,8 mt (cod. PAC600B018) al prezzo di Euro 2,20.

Il materiale va richiesto a: Futura Elettronica, Via Adige 11, 21013 Gallarate (VA) - Tel: 0331-799775 - www.futurashop.it

DAI CREATORI DELLA

3DRAG

La stampante 3D più versatile al mondo ad un prezzo imbattibile!

Cod. 3DRAG/K (in kit)

€ 499,00

IVA inclusa

Versione montata
cod. 3DRAG/M
€ 699,00 IVA inclusa.

Versione V1.2

E CON LA STAMPANTE 3DRAG PUOI ANCHE...

REALIZZARE CIRCUITI STAMPATI MEDIANTE INCISIONE

Minitrapano professionale
con 34 accessori
Cod. PROXXON1 - € 114,00



STAMPARE CON IL CIOCCOLATO

Set estrusore Choco
3DRAG - in kit
Cod. 3DCHOCO - € 178,00

VASTA GAMMA DI ACCESSORI!

SCOPRILI TUTTI SU
www.futurashop.it

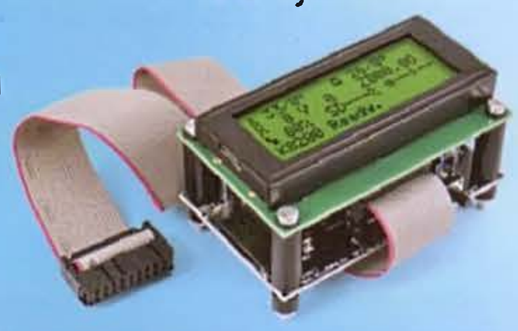
Illuminatore a LED
Cod. VM8202
€ 21,00



Nozzle in ottone con foro Ø 0,3 mm
Cod. NOZZLE03
€ 9,90
disponibile anche
con foro Ø 0,5 e 0,8 mm



Display LCD per stampa autonoma
Cod. VM8201
€ 64,00



Vasto assortimento di PLA ed ABS in bobine da 1KG



Piatto di stampa in vetro temperato
Cod. GP8200
€ 9,50



FUTURA ELETTRONICA®
www.futurashop.it

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche di questi prodotti
e acquisti on-line su *www.futurashop.it*

3D4040: STAMPA IN GRANDE!

di GABRIELE DAGHETTA

Capace di produrre stampe di 400 x 400 x 400 mm, integra tutte le funzionalità della 3Drag compresa l'hackerabilità caratteristica del controller basato su Arduino.

Quando realizzammo la nostra prima stampante 3D, che battezzammo 3Drag, eravamo agli inizi della nostra esperienza di progettisti di macchine per stampa additiva in tecnologia FDM, tuttavia il prodotto che ne scaturì fu molto apprezzato dal pubblico e lo è ancor oggi. Quella macchina, basata su un controller con hardware Arduino e derivato dal Sanguinololu del progetto RepRap, poi evolutosi nel nostro controller, sempre vitalizzato da un firmware derivato dal Marlin (il cuore del progetto RepRap) è ancora oggi un punto di riferimento e viene adottata sia da Maker e hobbisti, sia da professionisti che desiderano ottenere modelli e riproduzioni di oggetti in PLA o ABS.

La nostra 3Drag è limitata alla stampa di oggetti di 200x200x200mm al massimo, il che, per alcune applicazioni (ad esempio plastici di case) non è l'ideale; non è un caso che con il progetto 3Drag Big abbiamo aumentato la larghezza "stirando" il telaio originale della 3Drag. Ma siccome puntavamo in alto, non ci siamo fermati lì e abbiamo pensato a come ottenere una macchina in grado di raddoppia-



CARATTERISTICHE TECNICHE

- **Dimensioni di stampa: 400x400x400 mm**
- **Materiali utilizzabili: ABS e PLA**
- **Movimentazione: tecnologia Core XY**
- **Diametro filamento: 1,75 mm**
- **Diametro ugello fornito: 0,4 mm**
- **Diametro ugelli opzionali: da 0,3 mm a 0,8 mm**
- **Velocità di stampa massima: 350 mm/s**
- **Piano di stampa fisso: vetro temprato da 6 mm**
- **5 motori passo-passo (200 step)**
- **Struttura: profilati in alluminio**
- **Risoluzione meccanica asse Z: 0,625 micron**
- **Risoluzione meccanica assi X e Y: 0,0125 mm (12 micron)**
- **Ingombro: 690x630x610 mm**
- **Firmware: Marlin personalizzato per 3D4040**
- **Software: compatibile con software OpenSource (consigliato RepetierHost)**
- **Sistemi operativi supportati da RepetierHost: Windows – Mac – Linux**

re la "cubatura" stampabile dalla 3Drag; il frutto del nostro lavoro è la nuova stampante 3D4040, che vi presentiamo in queste pagine e che ha richiesto un lungo sviluppo e numerosi test al banco, prima di "vedere la luce".

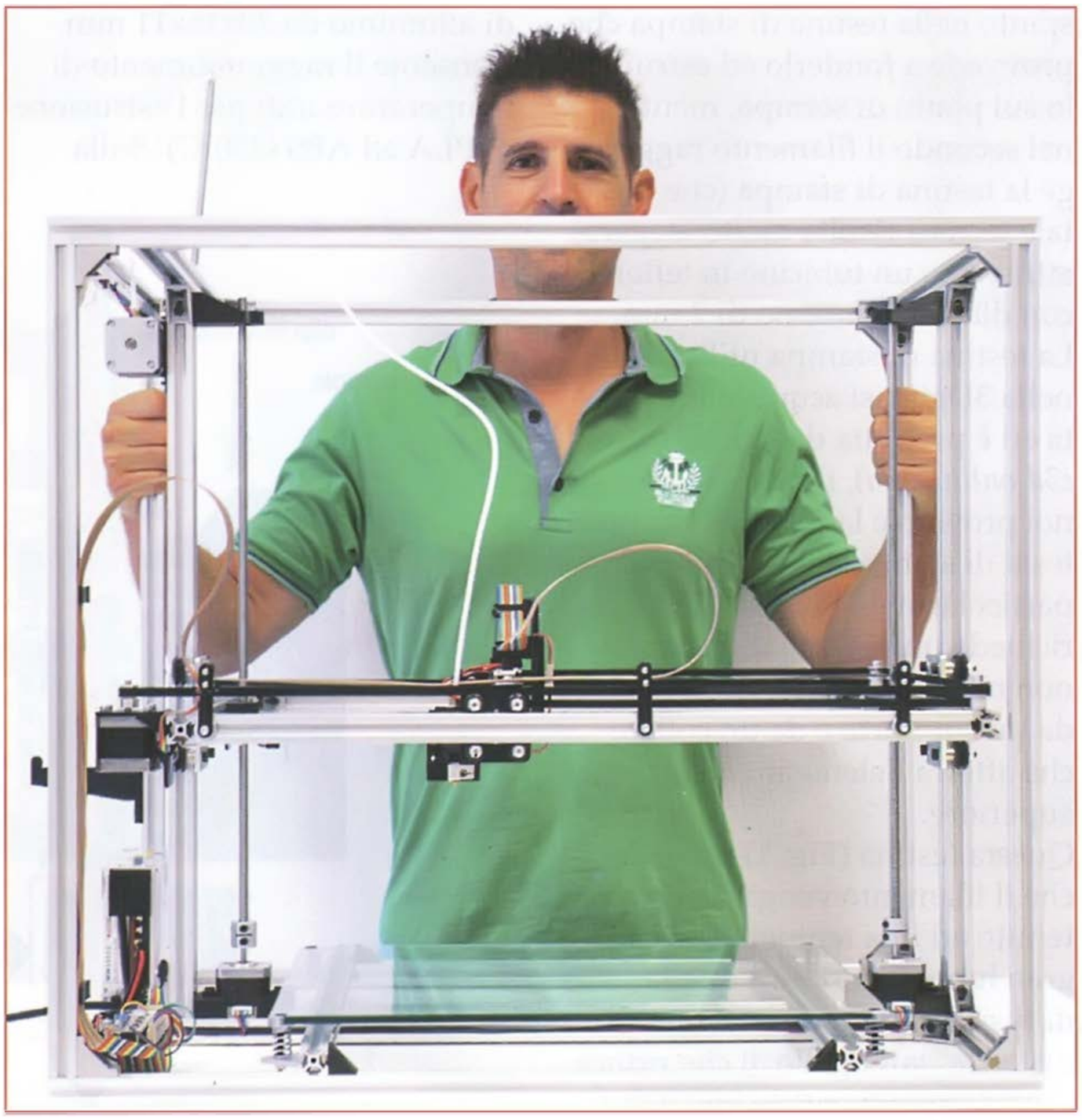
DA 3DRAG A 3D4040

Raddoppiare le dimensioni di stampa non è cosa semplice come si potrebbe pensare, perché, sebbene la soluzione più intuitiva che verrebbe in mente sarebbe raddoppiare le tre dimensioni della 3Drag originale, in realtà non è stato possibile. Infatti, se è vero che nel progetto 3Drag Big abbiamo semplicemente ingrandito la base del telaio consentendo un allungamento del lato in larghezza ma senza toccare profondità e altezza, quello è quanto si poteva fare. Tentando di estendere anche la profondità ci si scontrava con vari problemi e limiti della meccanica della 3Drag, nata per il formato 200x200x200 mm e non idonea a reggere le sollecitazioni causate dalla stampa di pezzi di grandi dimensioni; idem per la struttura a "patibolo" che reggeva la testa e l'azionamento dell'asse Z.
Va poi detto che utilizzare la filosofia adottata per realizzare la 3Drag avrebbe imposto una struttura di dimensioni almeno doppie, quindi con un piatto da 40 cm di lato avremmo ottenuto un ingombro di almeno 80 cm su ogni lato; il motivo è che nella 3Drag la testa sale e scende soltanto. La testa fissa sul piano orizzontale implica che la parte più estrema del pezzo sia spostata rispetto alla testina (che sta al centro della macchina) di una distanza pari alla massima larghezza sull'asse X e parimenti sull'asse Y (in profondità): una cosa inaccettabile, sia per ingombro, sia per costo e peso che ne sarebbero derivati.
Inoltre la movimentazione di un piatto di tali dimensioni, peraltro realizzato in vetro per ragioni di rigidità non avrebbe molto senso per ovvi motivi (masse e inerzie elevatissime).
Per questi motivi abbiamo pensato di passare a una meccanica con piatto fisso in orizzontale, il cui ingombro non supera di molto la dimensione del piatto stesso, rendendo invece mobile la testa di stampa sugli assi X e Y, dato che la testa ha una massa ridotta e tale che l'inerzia, anche alle più elevate velocità di stampa, è poco rilevante. In definitiva, abbiamo adottato un approccio simile a quello della 3D Vertex.
Tornando alla 3Drag, va detto che il piatto scorre su bàrre rettificate, soluzione più che valida per escursioni limitate a 20 cm, ma improponibile per dimensioni maggiori perché a parità di diametro di barre si avrebbero delle flessioni inaccettabili. L'adozione di guide lineari su binari (ad esempio come quelle della hiwin, le cui caratteristiche si possono consultare sul web alla pagina *www.hiwin.de/it/Prodotti/Guide_Lineari/4263*) è stata scartata a priori a causa dell'elevato costo della guida. La soluzione più sensata è quella di sfruttare le cave del profilato utilizzato per costruire il telaio, come guida per ciascuno degli assi. Per questo motivo la struttura della 3D4040 è stata realizzata interamente con profili in alluminio definiti V-slot che aggiungono una nuova funzionalità rispetto ai tradizionali T slot; grazie alla scanalatura a V, infatti, consentono di realizzare sistemi di movimento di tipo lineare che fanno uso prevalentemente di apposite pulegge di scorrimento (V-Wheel) generalmente realizzate in POM (Poliossimetilene) caratterizzate da una considerevole precisione e lunga durata.
Questi profili, che abbiamo unito tra di loro mediante angolari in alluminio pressofuso, hanno permesso di realizzare una struttura portante robusta ma nel contempo leggera e funzionale (oltre che economica...) che non necessita di tiranti o di piastre di irrigidimento malgrado le dimensioni del telaio siano di tutto rispetto: ben 690x630x610 mm.
Il telaio dell'asse Z si muove in verticale grazie a due motori passo-passo collegati in parallelo allo stesso driver (asse Z – codice Futura Elettronica 3DDRIVER) che a loro volta azionano due viti

trapezie ad un principio (in acciaio inox) con passo 2 mm; tale sottosistema è composto dalla cornice che permette il movimento XY della testina. In sostanza la testina, tramite l'asse Z viene allontanata o avvicinata al piatto di stampa, che nel nostro caso rimane solidale con la struttura. Questa soluzione permette di semplificare sensibilmente il telaio, ma soprattutto consente di mantenere basso il baricentro dell'intera struttura, cosa utilissima dato che il piatto, in vetro temprato da 6 mm di spessore, pesa circa 3kg!
Il telaio dell'asse Z scorre verticalmente grazie a due carrelli laterali dotati di pulegge che scorrono nelle cave dei montanti, composti ognuno da un profilato in alluminio con sezione di 20x40 mm ciascuno.
Il sistema utilizzato per movimentare l'estrusore viene chiamato **core XY**; potete trovare maggiori dettagli sul web, alla pagina *http://reprap.org/wiki/CoreXY/it*. Il meccanismo che ne è alla base dispone di due motori che agiscono contemporaneamente sul posizionamento XY della testina di stampa (mediante due distinte cinghie di tipo GT2 da 6 mm con passo di 2 mm); l'utilizzo di due cinghie anziché una molto lunga riduce gli errori di posizionamento, perché un'unica cinghia lunga è più soggetta a deformazione rispetto a due cinghie distinte (le deformazioni naturali interesseranno in maniera uguale entrambe le cinghie pertanto non vi sarà disparità tra le due). Unico requisito è che entrambe le cinghie devono avere il medesimo tensionamento.
Grazie a questa tecnologia e al sistema di alimentazione dove il motore che spinge il filo si trova sul telaio e il filo raggiunge la testa di stampa attraverso un bowden, ci sono minori masse in movimento, il che permette di raggiungere maggiori velocità di stampa con un'elevata precisione.
Aver affidato lo spostamento XY alla testa di stampa permette, rispetto al sistema cartesiano Z-head (quello della 3Drag, per intenderci), oltre che di stampare più alla svelta e ottenere maggior precisione (dovuta ad una minor inerzia), di mantenere una struttura del telaio che risulti più ordinata.
Altra differenza tra la 3D4040 e la 3Drag è che in quest'ultima il motore Y è montato sul carrello X mentre nella 3D4040 i motori dell'asse X e Y sono entrambi fermi e fissati alla cornice del carrello asse Z. Nella 3Drag, invece, è il piatto viene mosso, il che, a causa dello spostamento di masse non indifferenti, può provocare perdite di passo e soprattutto vibrazioni in presenza di oggetti molto alti, il che va a ripercuotersi sul grado di finitura della parte alta dell'oggetto stesso.
Nella 3D4040 la risoluzione, ossia il minimo spostamento ottenibile sull'asse Z per ogni singolo micro-step, è pari a 0,000625 mm; tale misura è determinata da:

Z = passo vite/3200

dove passo vite è il passo della barra filettata che fa da traino, azionata dal motore passo-passo, nel nostro caso pari a 2 mm. Parliamo di micro-step perché, come nella 3Drag, anche in questa macchina è possibile impostare il controllo a frazioni di passo, fino a 1/16, quindi, essendo i motori stepper i soliti NEMA 17 da 200 step/giro, otteniamo un numero massimo di micro-step pari a 200x16=3.200.



ESTRUSORE

La 3D4040 presenta innovazioni anche nell'estrusore, che qui è basato su una leggerissima testina di stampa E3D V6 per filamenti da 1,75 mm, che consente di ridurre considerevolmente le masse in movimento, quindi di stampare più veloce. Inoltre l'estrusore supporta ugelli (nozzle) di vario diametro: da 0,3 mm per chi necessita di stampe molto dettagliate, fino a 0,8 mm per chi vuole ridurre i tempi di stampa.
L'alleggerimento della testa è ottenuto, come già detto, dal fatto che mentre l'estrusore della 3Drag comprendeva anche il motore di trascinamento del filamento di stampa, qui la testina di stampa è separata dal meccanismo di alimentazione, che è invece opportunamente fissato al telaio della macchina come è stato nella 3D Vertex. Nel primo caso il filo viene direttamente spinto nella testina di stampa che provvede a fonderlo ed estruderlo sul piatto di stampa, mentre nel secondo il filamento raggiunge la testina di stampa (che con tale tecnica risulta molto leggera) attraverso un tubicino in teflon con diametro interno di 2 mm.
La testina di stampa utilizzata nella 3Dgiga si acquista già pronta ed è prodotta dalla E3D (*http://e3d-online.com*); il modello da noi provato è la E3D V6. Questa testa di stampa ha dimensioni particolarmente contenute, però richiede un sistema di fissaggio non molto semplice, composto da un supporto e da un collare che afferra l'elemento nella parte superiore;
Questa testina (**Fig. 1**) prevede che il filamento venga mantenuto ad una temperatura di gran lunga inferiore a quella di fusione fino a quasi 2 mm di distanza dall'ugello, il che riduce considerevolmente le inevitabili deformazioni che provocherebbero rallentamenti o addirittura ostruzioni all'interno del passaggio. La strozzatura situata nella parte bassa (che unisce il blocchetto riscaldatore al resto del corpo) rappresenta la soluzione vincente per ottenere il dovuto taglio termico, impedendo che parte del calore generato dalla cartuccia riscaldante venga dissipato da altre parti che non siano l'ugello vero e proprio consentendo un'elevata reattività e resa con minimi sprechi di energia.
Il raffreddamento del corpo in alluminio della testina è garantito dal flusso d'aria generato da una ventola 40x40 mm (a 12 Vcc) applicata direttamente su di esso mediante apposita clip (disegnata in 3D e stampata in PLA).
L'elemento riscaldante è una cartuccia cilindrica da 40 W di potenza (Ø 6x21 mm) alimentata a 12 Vcc inserita in un blocchetto di alluminio da 20x16x11 mm.
Consente il raggiungimento di temperature utili per l'estrusione di PLA ed ABS (250°C). Sulla testina può essere montato un ugello, realizzato in ottone, con foro che può variare da 0,3 a 0,8 mm. Il controllo della temperatura è affidato ad un termistore NTC. Per l'impostazione del PID si utilizza una funzione presente in Marlin (quello da noi utilizzato è la versione: 1.1.0-RC8) che consente di effettuare l'autotuning (Info per PID Autotune http://reprap.org/wiki/PID_Tuning) e di determinare i tre valori da assegnare ai parametri Kp, Ki e kd.
La stessa procedura viene utilizzata per il piatto riscaldato.
Per il piatto della stampante è stato scelto il vetro sulla base di una considerazione: viste le grandi dimensioni, per evitare che fletta durante la stampa, è necessario realizzarlo con un materiale ad elevata rigidità, ma anche a basso costo. Il vetro temprato da 6 mm si è rivelato un ottimo compromesso e ha fornito risultati soddisfacenti anche quando sottoposto a riscaldamento.
Utilizzando del metallo sarebbe

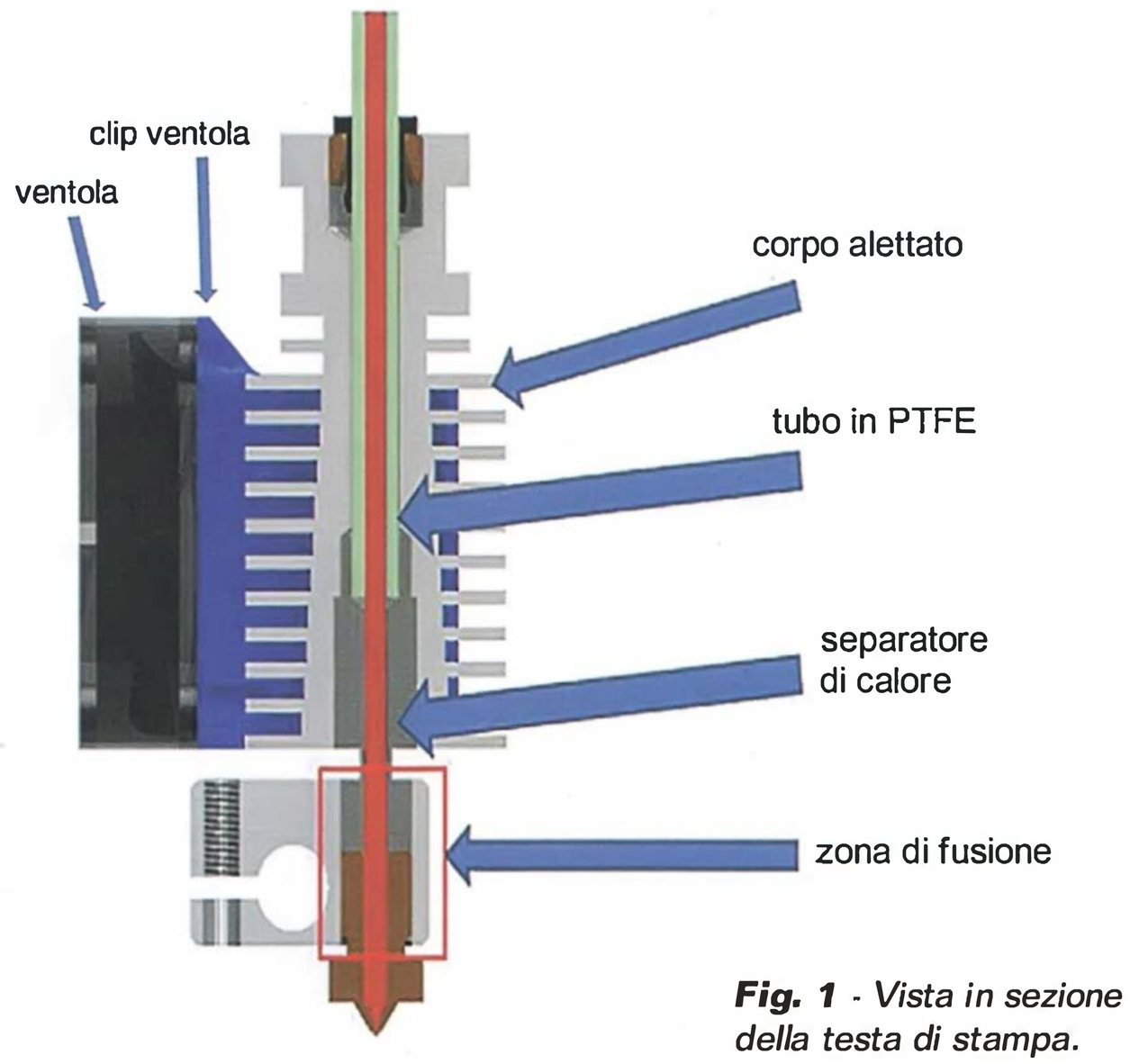


***Fig. 1** - Vista in sezione della testa di stampa.*

stato necessario qualcosa di più pesante, almeno per ottenere la stessa rigidità.

L'ELETTRONICA

Il controller della stampante 3D4040 è il medesimo della 3Drag (3DCONTR-DRIVER commercializzato da Futura Elettronica, su *www.elettronicain.it*) ma richiede l'utilizzo di un SSR (relé allo stato solido) per la gestione dell'elemento riscaldante del piatto, onde evitare surriscaldamenti delle piste e del connettore di alimentazione posto sulla scheda; infatti qui, vista la superficie di base quadruplicata, occorre un riscaldatore molto più potente, che assorbe una corrente di valore tale da mettere a dura prova il MOSFET (che non ha dissipatore) e il circuito stampato (nonché il connettore) di alimentazione della scheda controller. La potenza di quello testato supera i 200 watt!

Per quanto riguarda il cablaggio, è sostanzialmente identico a quello della 3Drag con il controller di ultima generazione, ma in più ha un secondo motore stepper Nema 17 sull'asse Z, dato che il piatto viene sollevato insieme da due barre filettate azionate ognuna da un proprio motore (i due sono sincronizzati perché collegati in parallelo).

La scheda controller, meglio descritta nell'articolo dedicatole nel fascicolo n° 175 di Elettronica In, è basata su un microcontrollore ATmega 2560 e condivide buona parte dell'hardware della RAMPS (RepRap Arduino Mega Pololu Shield). Il processore ATmega2560 dispone di ben 256 kB di memoria di programma e lavora con un clock di 16 MHz, quindi è potente quanto basta e può ospitare il firmware Marlin da noi modificato per governare la stampante 3D4040.

La nostra scheda può essere programmata direttamente dall'IDE Arduino e dispone, allo scopo, di una connessione USB per collegarla al computer tramite un normale cavo miniUSB; la stessa porta permette il controllo da PC durante le stampe.

Il circuito monta quattro moduli driver per motori passo-passo (3DDRIVER), dato che a bordo macchina abbiamo un motore per ciascuno degli assi X e Y, due in parallelo per lo Z ed un quinto motore per comandare l'ingranaggio che spinge il filo di materiale plastico, attraverso il bowden, dentro l'estrusore; la scheda ha inoltre tre MOSFET che ci permettono di pilotare due riscaldatori (uno è quello del piatto e l'altro quello che scalda l'ugello dell'estrusore) ed una ventola a bassa tensione, oltre ad un converter USB/seriale per interfacciare l'ATmega con il computer e ad uno stadio di alimentazione, che completa l'insieme.

I driver per i motori passo-passo sono i moduli prodotti dalla Futura Elettronica (cod. 3DDRIVER); ciascuno è sostanzialmente un integrato A4988 della Allegro, molto versatile perché può essere impostato per definire sia la direzione di rotazione dell'albero del motore, sia il numero di gradi che il rotore del motore deve compiere alla ricezione di ogni comando; in altre parole, possiamo decidere se quando gli forniamo un impulso di comando, il modulo deve far ruotare l'albero di uno step alla volta, oppure di 1/2, 1/4, 1/8 o 1/16, in base all'accuratezza che si desidera ottenere. Supporta, perciò, la modalità microstep.

Il motore si muove, ad ogni impulso che l'ATmega invia su /STEP, in base all'impostazione di MS1, MS2, MS3; nel nostro caso, avendo impostato il massimo fattore di divisione, per un passo completo occorrono 16 impulsi, quindi con i nostri NEMA 17 un giro completo richiede al microcontrollore 3.200 impulsi.

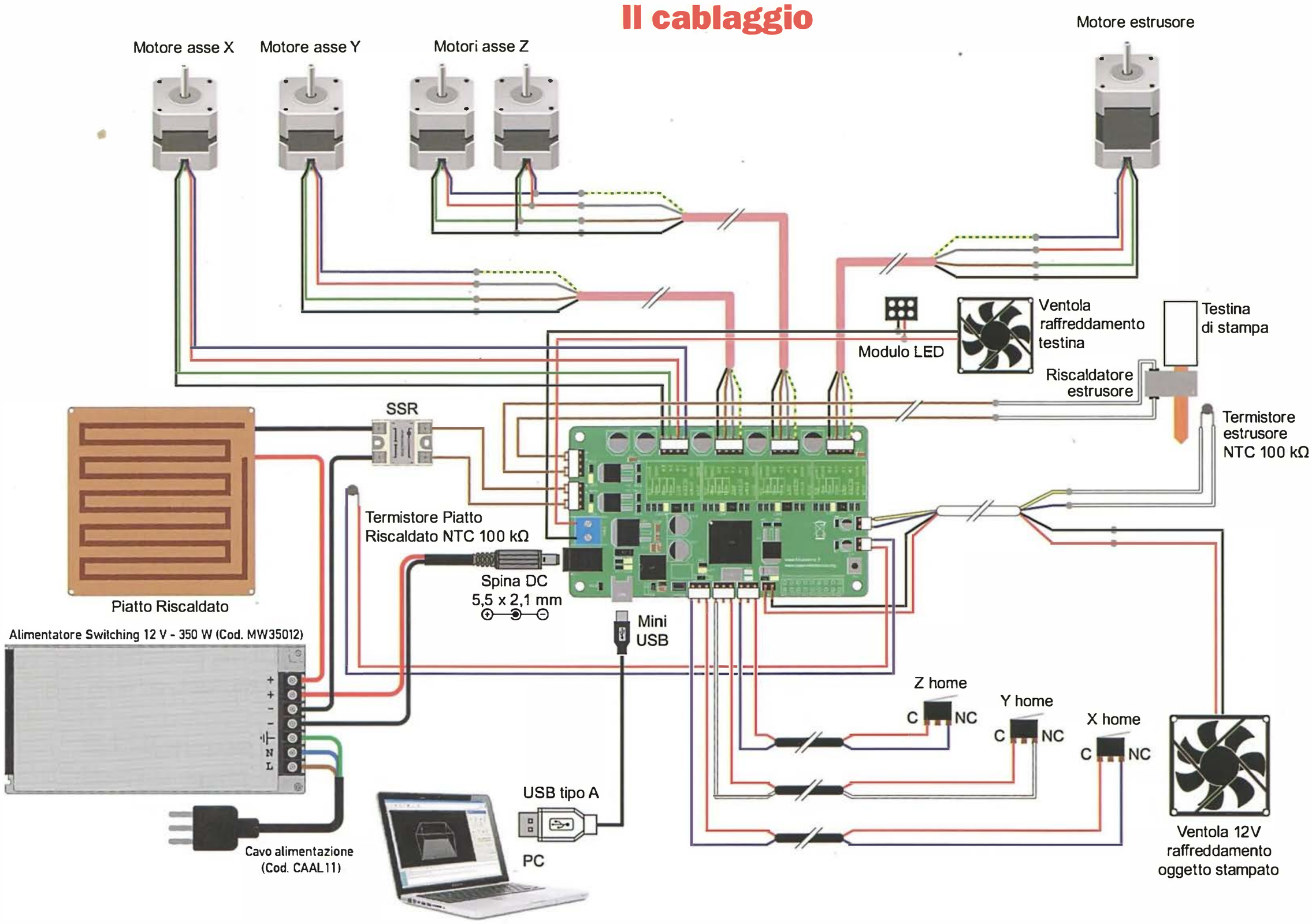
Notate che la linea /STEP di ciascun controller è stata collegata ad un transistor NPN, che serve come amplificatore di corrente per pilotare un LED, il quale pulserà analogamente alla corrispondente linea di comando del microcontrollore in modo da farci monitorare quello che accade. Così, nel caso durante il test (o l'uso) un motore non giri malgrado il relativo LED pulsi, vuol dire che il problema è all'interno del driver o nel motore, ovvero nei cablaggi.

Resta inteso che siccome il LED pulsa alla stessa frequenza degli impulsi di comando, potremo vederlo lampeggiare solo quando il motore corrispondente girerà a velocità molto ridotta, dato che già a 25 Hz l'occhio umano vede il diodo sempre illuminato.

Passiamo adesso ai driver a MOSFET, utilizzati per alimentare i riscaldatori e la ventola: sono tutti dei BUK6215-75C, capaci di erogare una corrente di drain che raggiunge 57 A e sopportare una Vds in stato di interdizione, pari a 75 V; la loro bassissima Rdson (15 milliohm max a 15 A di Id) consente di minimizzare la potenza dissipata e perciò di saldarli direttamente ad una pista dello stampato (che funge da dissipatore).

Nella scheda, i MOSFET non prevedono dissipatore perché vengono utilizzati a correnti molto ridotte (siamo nell'ordine dei 2÷2,5 ampere per il riscaldatore dell'estrusore, che va collegato all'uscita HEATER1). Siccome la scheda nasce per la 3Drag, il MOSFET che comanda il piatto riscaldato è anch'esso privo di

Il cablaggio



dissipatore, quindi non può commutare la corrente diretta al riscaldatore del piatto in vetro della nuova stampante; dunque, al connettore HEATER2 colleghiamo un relé allo stato solido (SSR) che a sua volta alimenterà il piatto, come mostra lo schema di cablaggio visibile in queste pagine.

Il micro legge le sonde di temperatura che riportano il feedback sull'attività dei riscaldatori; queste sonde sono termistori NTC da collegare ai contatti THERM1 e THERM2. Più esattamente, a THERM1 va collegato il sensore interno all'estrusore, mentre a THERM2 va collegato l'NTC che rileva la temperatura raggiunta dall'eventuale piatto riscaldato. Le sonde consentono di stabilizzare la temperatura di estrusore e piatto riscaldato; senza questa retroazione, anche ammesso di lavorare a corrente costante e di supporre che a quella corrisponde una certa temperatura, le variazioni ambientali e operative potrebbero alterare la temperatura alla quale viene portato il materiale plastico da estrudere, con evidenti problemi di deposizione e di composizione dell'oggetto in stampa. Rilevando la temperatura dell'estrusore attraverso la tensione presente ai capi THERM1, il microcontrollore verifica il dato con quello impostato: se la temperatura eccede il valore settato, riduce la corrente nel filamento riscaldatore, mentre in caso contrario la aumenta. Analogo è il discorso per il riscaldatore del piatto riscaldante, che comunque è un accessorio e la sua assenza non pregiudica il funzionamento della macchina; infatti il firmware base ha disattivato il controllo del piatto e della temperatura raggiunta, che servirebbe, in abbinamento al client di stampa, a far avviare la stampa stessa solo una volta che il piatto ha raggiunto la temperatura di regime. Notate che i termistori NTC utilizzati per rilevare la temperatura dell'estrusore e del piatto riscaldato sono entrambi da 100 kohm a 25 °C.

I transistor vengono pilotati tutti con un segnale PWM, così da variare la potenza ceduta agli utilizzatori e controllare il calore generato dalle resistenze dei riscaldatori e la velocità della ventola del raffreddatore del pezzo, però con bassissime perdite, dato che i MOSFET lavorano in modo on-off. Anche l'uscita che comanda il relé allo stato solido, cui ci siamo affidati per commutare

l'alimentazione del riscaldatore del piatto di stampa, produce un segnale di comando PWM a bassissima frequenza.
Ciascuna delle uscite riservate ai riscaldatori è provvista di un LED che pulsa insieme al relativo segnale PWM da circa 4 Hz, permettendo di verificare visivamente lo stato di funzionamento; il PWM della ventola di raffreddamento è invece a frequenza più alta e il relativo LED sembrerà sempre acceso.
La scheda controller legge lo stato degli switch finecorsa sui tre assi di spostamento; tutte le relative linee del microcontrollore sono provviste di pull-up interno, attivato dal firmware. I finecorsa sono interruttori o commutatori che, opportunamente posizionati, vengono azionati dall'arrivo a fondo corsa del piatto o del sostegno della testa di stampa; nel nostro caso, vengono situati sul telaio. Quando un finecorsa si aziona, il microcontrollore ne riceve informazione sotto forma di stato logico e provvede ad arrestare il movimento corrispondente e a far rientrare il carrello o rialzare la testa di stampa.
La nostra scheda dispone di tre ingressi per i finecorsa: XSTOP che corrisponde a YSTOP che coincide con ZSTOP.
Per ciascun finecorsa sono stati previsti tre contatti: +, S e -.
Questa soluzione è stata scelta per poter collegare sia finecorsa di tipo elettromeccanico (microswitch e vari) sia rilevatori ottici, composti da un fotodiodo e un LED ad infrarossi affacciati tra loro in una cavità: quando il carrello arriva, una linguetta fissata ad esso passa nella cavità e blocca la luce che dovrebbe raggiungere il fotodiodo, determinando l'attivazione del finecorsa.
Nel nostro caso, siccome utilizziamo degli switch finecorsa li colleghiamo ognuno tra S e – (massa). così la condizione di arrivo a fine corsa coincide con il livello alto sul rispettivo input.
Per la comunicazione con il computer, la scheda controller utilizza un convertitore USB/seriale FT232RL della FTDI, contenente la logica che occorre a trasformare i dati dal formato seriale TTL all'USB, smistando i dati in arrivo da TXD ed RXD sui piedini DP e DM.

REALIZZAZIONE PRATICA

Passiamo adesso alla realizzazione della stampante, che consta di una meccanica e di un'elettronica; per prima bisogna assemblare la meccanica, composta da telaio e parti in movimento: quindi struttura portante, carrelli per il movimento degli assi Z e X, Y, piatto, gruppo testa, movimentazione. Tutte le fasi dell'assemblaggio sono doviziosamente descritte nelle istruzioni di montaggio scaricabili dal nostro sito *www.elettronicain.it* insieme agli altri file del progetto. Qui di seguito le vediamo in sintesi.
Per prima cosa dovete mettere insieme lo scheletro del telaio, unendo tra loro due profilati in alluminio 20x20 da 690 mm e due da 588 mm; l'assemblaggio di questi e di tutti gli altri profilati utilizzati nella stampante avverrà mediante angolari in alluminio pressofuso (**Fig. 2**) e bulloni M5x10 TCEI più dadi TM5. Tale soluzione conferisce rigidità e stabilità al telaio della macchina.
Nel mezzo collocate un ulteriore profilato da 690 mm ottenendo la situazione di **Fig. 3**.
Montate in verticale, agli angoli della base del telaio, quattro profilati 20x20 da 570 mm, che costituiranno i montanti del telaio. Poi dovete fissare verticalmente un ulteriore profilato 20x40 da 570, poi prendere due profilati in alluminio 20x20 da 690mm e due da 588mm (identici a quelli utilizzati per la base) e fissarli sulla parte superiore del telaio utilizzando tre angolari per ciascuno vertice. Prendete l'ultimo profilato 20x40 da 570mm e fissatelo al centro del profilo 20x20 da 588 mm; a questo punto posizionate la "T" così ottenuta nel telaio, ottenendo quanto mostrato nella **Fig.** 4.
Ora è il momento di realizzare il sistema composto dal telaio dell'asse Z e relativi carrelli laterali che servono a consentirne lo spostamento; ciascuno ha come base una piastra in alluminio e andrà applicato al montante dedicato, vincolandolo con delle pulegge a V bloccate da viti M5x30 e dadi.
Adesso preparate la meccanica che permette lo scorrimento in orizzontae della testa di stampa, ossia il meccanismo X-Y: si basa su un profilato in alluminio 20x20 da 564mm con applicati alle estremità i carrelli visibili nel dettaglio della **Fig. 5**. Il sistema siffatto consente lo spostamento sull'asse X e nella **Fig. 5** appare completo e montato nel telaio asse Z.



***Fig. 2** - Gli angolari del telaio.*

Non resta che montare il telaio X-Y (ossia quello che scorre sull'asse Z) sui carrelli Z preventivamente applicati ai montanti laterali che fanno da guida, ottenendo quanto mostrato nella **Fig. 6**, nel cui dettaglio vedete un carrello Z dal lato esterno.
Ora è il momento di realizzare la movimentazione degli assi: per prima cosa appoggiate due staffe motore (**Fig. 7**) sul lato destro della base e posizionate tra di esse un motore NEMA17 da 1,5A (corpo corto). Fate altrettanto sul lato sinistro.
Avvitate (dall'alto verso il basso) una vite trapezia Ø 8 mm passo 2 mm da 500 mm fino a farla fuoriuscire dalla rispettiva chiocciola (quella gialla in **Fig. 7**) per 4-5 cm; applicate sull'estremità inferiore della barra filettata un giunto in alluminio e bloccatelo, quindi avvitate le barre fino ad inserire completamente l'albero di ciascun motore nel relativo giunto. Prendete i due supporti per cuscinetto Z ed inserite nella relativa sede un cuscinetto, bloccandoli poi al telaio allo stesso livello dell'angolare superiore.
Ora preparate il carrello portatesta (consentirà il movimento sull'asse Y): prendete l'apposita piastra quadrata in alluminio anodizzato nero e collocatela sulla guida del carrello X con le relative pulegge, quindi avvitatevi una piastra in alluminio appositamente sagomata per il fissaggio delle cinghie. Completate il carrello Y montando la seconda piastra per fissaggio cinghie.
Posizionate sul telaio X-Y i due motori NEMA17 per il comando dei movimenti sugli assi X e Y, quindi dotateli di pulegge dentate. Prendete la cinghia dentata e dividetela in due spezzoni di uguale lunghezza (circa 2,5 m ciascuno), poi applicatela nell'or-

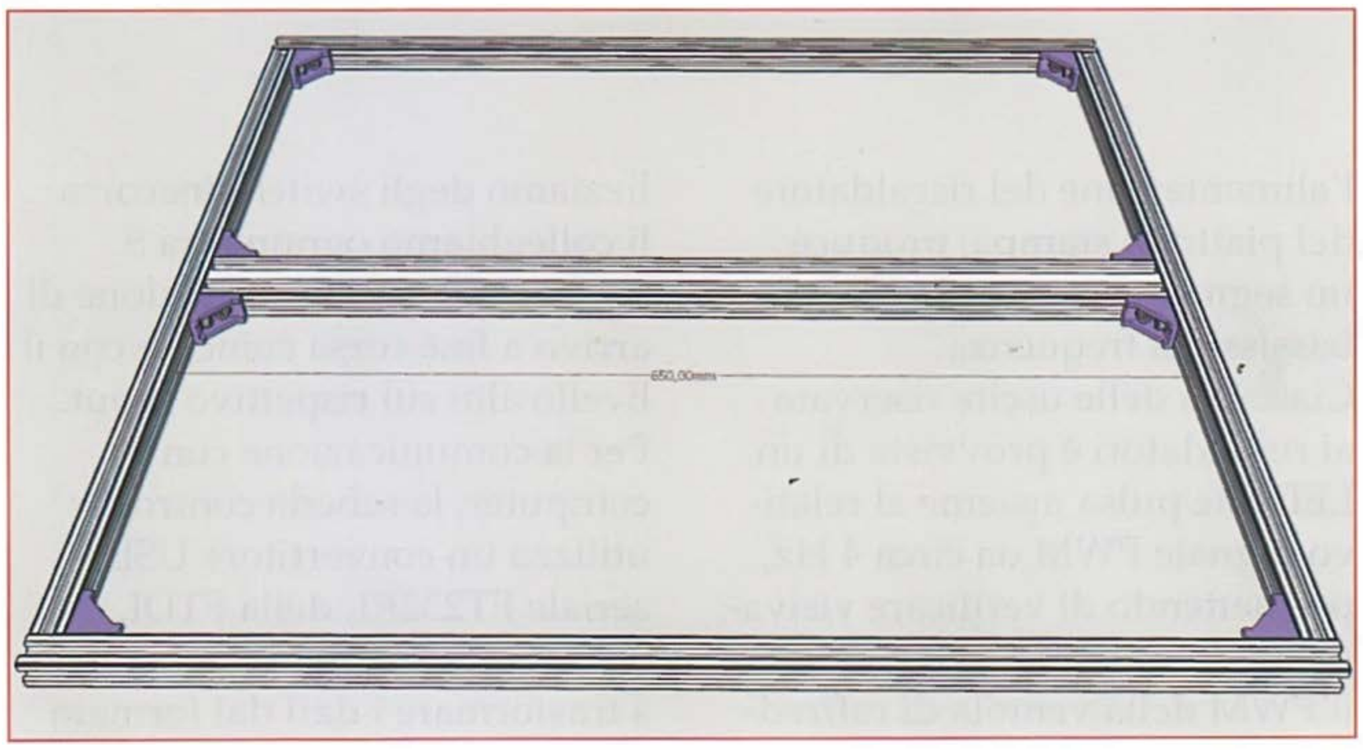


Fig. 3 *- Base del telaio con montato il profilato centrale di rinforzo.*

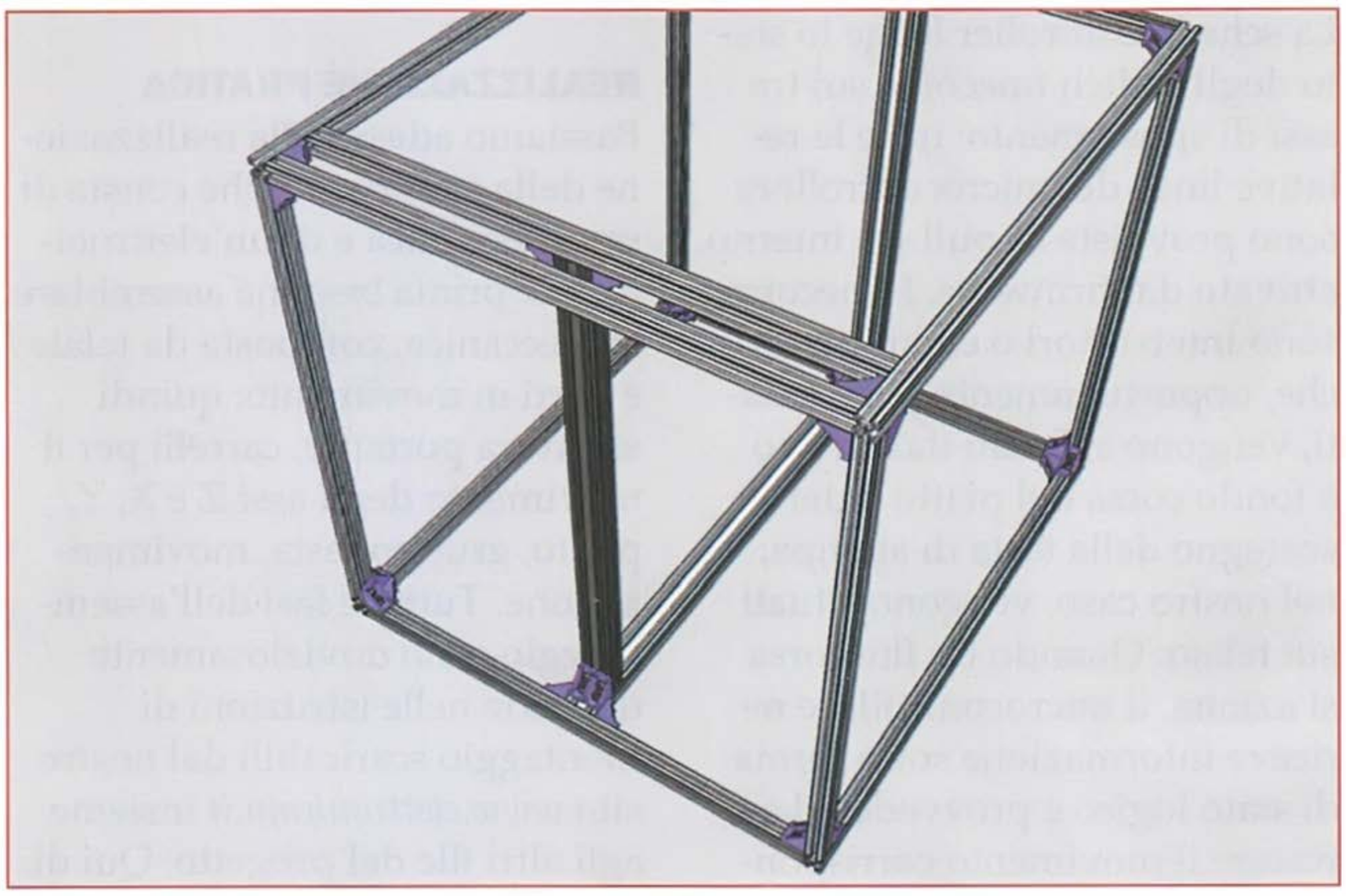


Fig. 4 *- Il telaio completato.*

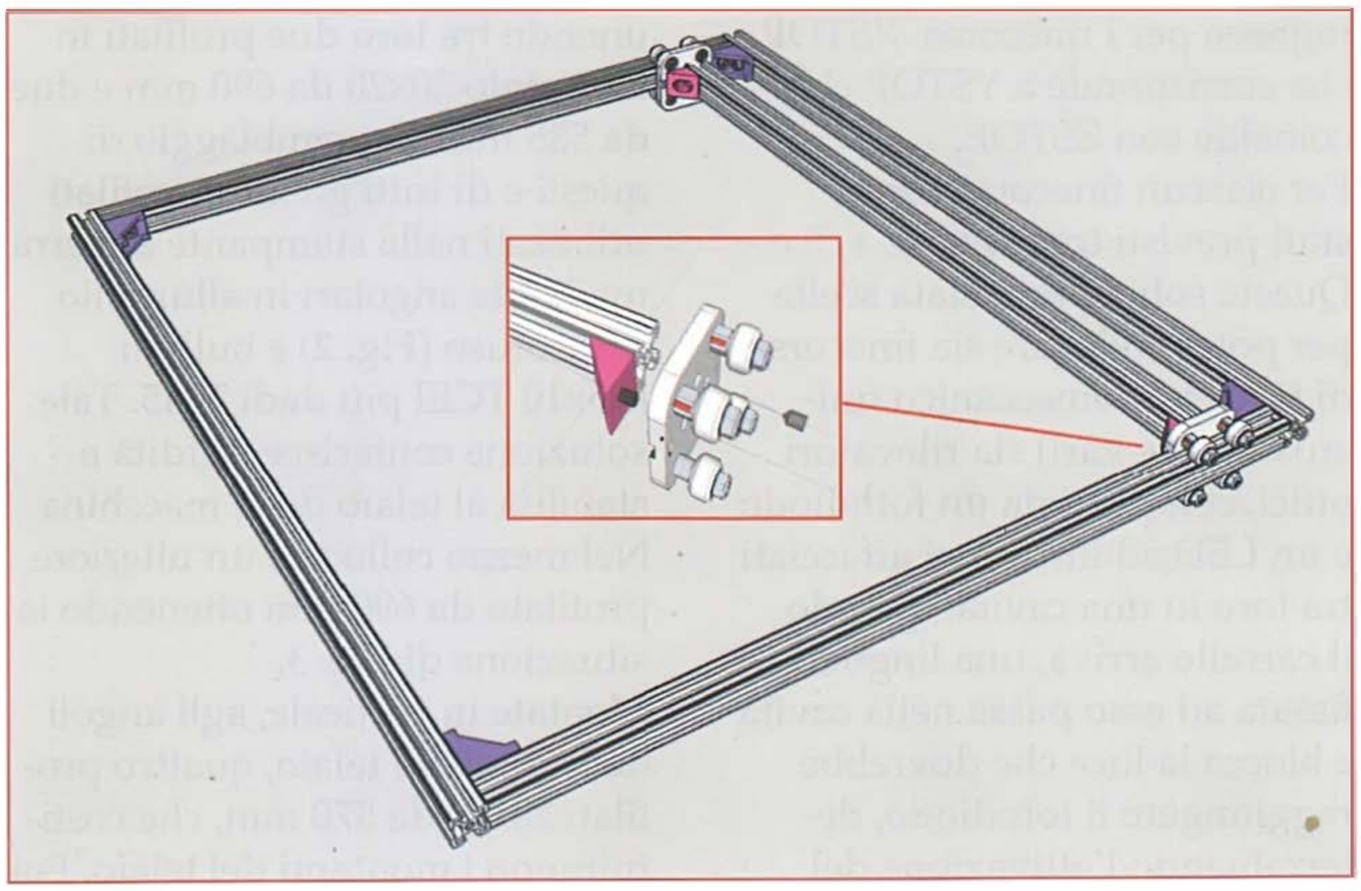


Fig. 5 *- Assemblagio della cornice mobile portatesta con il profilato dotato, alle estremità, dei carrelli per il movimento sull'asse Y.*

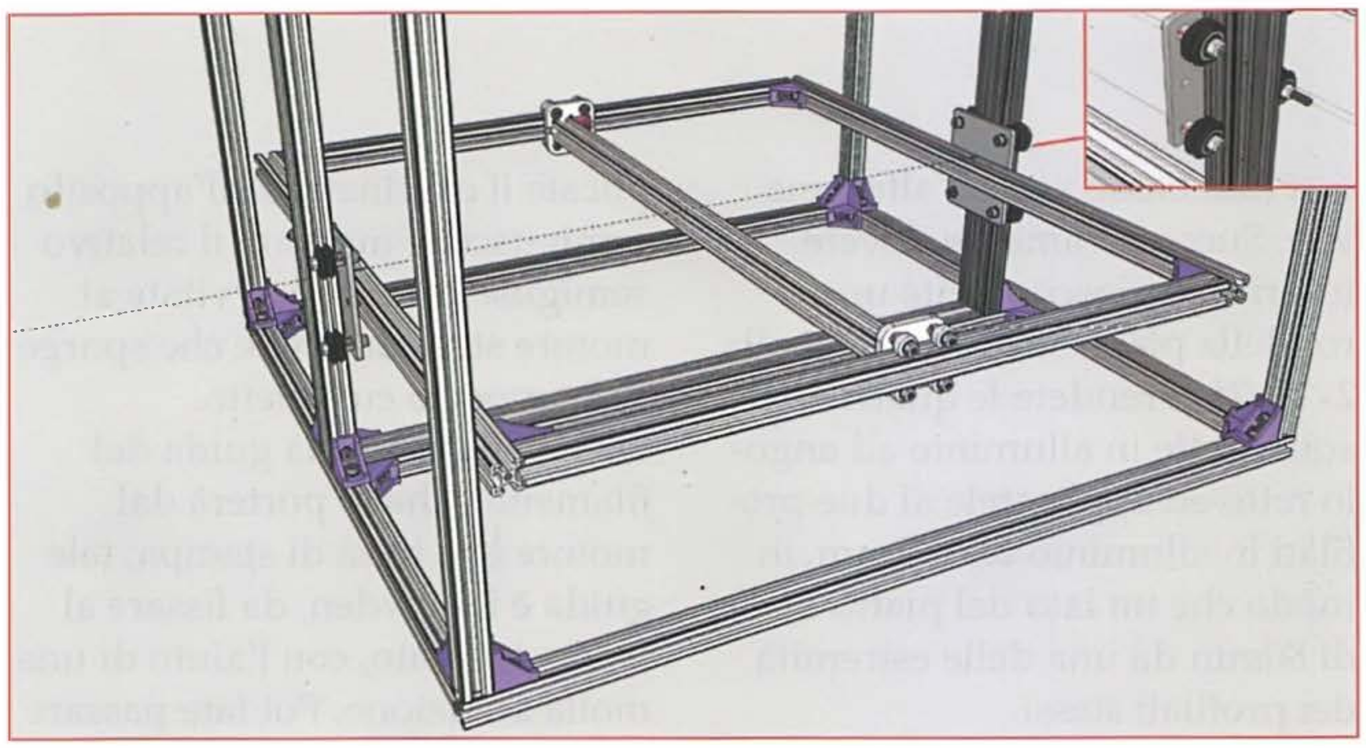


***Fig. 6** - Il telaio X-Y inserito in quello della macchina coi suoi carrelli asse Z.*

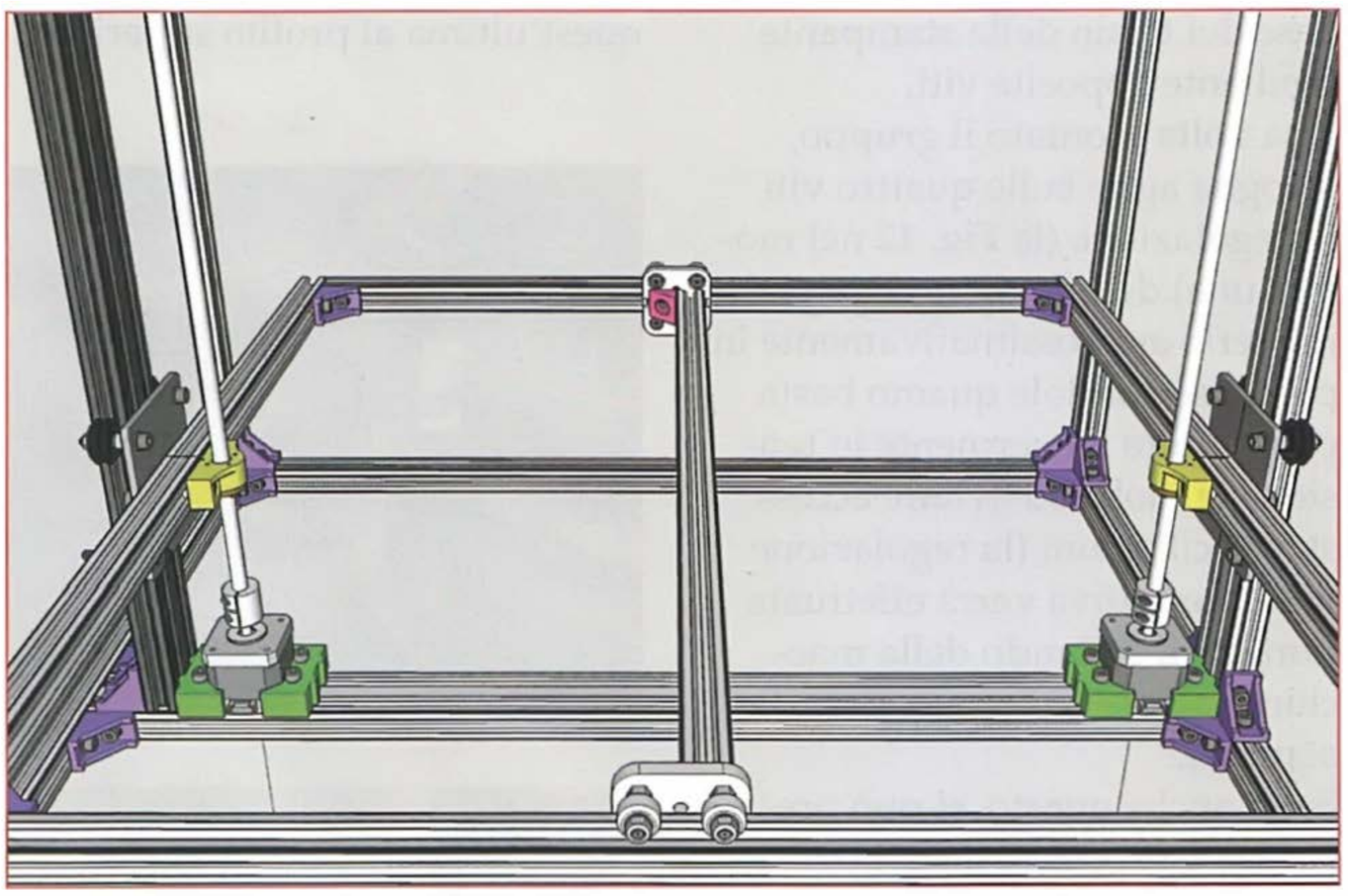


***Fig. 7** - Assemblaggio dell'azionamento asse Z.*

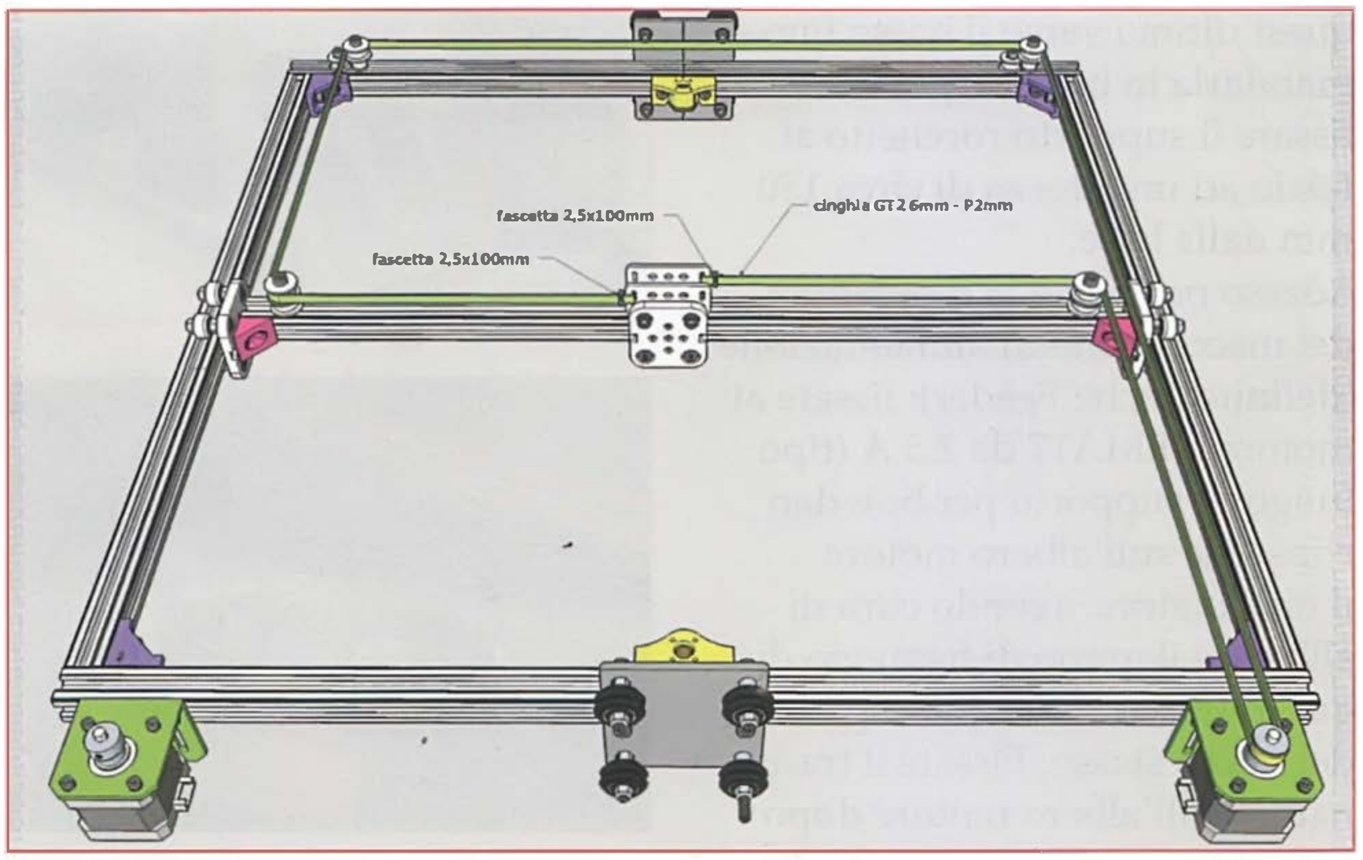


***Fig. 8** - Cornice porta testa con montati gli stepper-motor e le cinghie per il movimento X-Y.*

dine sulla puleggia del motore destro, sulla puleggia centrale e successivamente fissatela alla piastra anteriore del carrello Y mediante una fascetta. Stendete il resto della cinghia sulle altre pulegge fino a raggiungere l'altra piastra dove andrà fissata sempre con una fascetta. Effettuare la stessa operazione con l'altra cinghia. Il gruppo di movimentazione risulterà come in **Fig.** 8.
Passiamo adesso alla testa di stampa, che andrà applicata al carrello portatesta (**Fig. 9**) applicando sul corpo alettato della testina la clip per ventola (**Fig. 10**); i fili dovranno passare come in **Fig. 14** ed essere bloccati con una fascetta. Fatto ciò, sulla stessa clip bisognerà montare una ventola 40x40x10 mm (la ventola andrà orientata in modo che la direzione dell'aria sarà verso la testina). Applicate anche il convogliatore in plastica all'estremità inferiore del condotto aria (orientandolo come in **Fig. 10** in basso) mediante viti autofilettanti. Poi fissate al convogliatore una seconda ventola 40x40 (che deve soffiare verso di esso).
A questo punto dovete montare i fine-corsa a microswitch, i primi due dei quali vanno lavorati accorciando la linguetta fino a 33 mm, poi piegandola leggermente all'estremità verso il corpo. Con una punta da trapano da 3 mm bisogna allargare i fori di fissaggio così da montarli con viti 3MA. Fissate quindi il microswitch asse Y al profilato, con una vite M3 in modo che scatti quando il carrello Y si trova a circa 42 mm di distanza dalla piastra del carrello posto sull'estremità sinistra. Fissare il microswitch X al profilato in modo che scatti quando il carrello X si trova a circa 85mm di distanza dal bordo interno del profilo su cui sono montati i motori. Prendete il microswitch

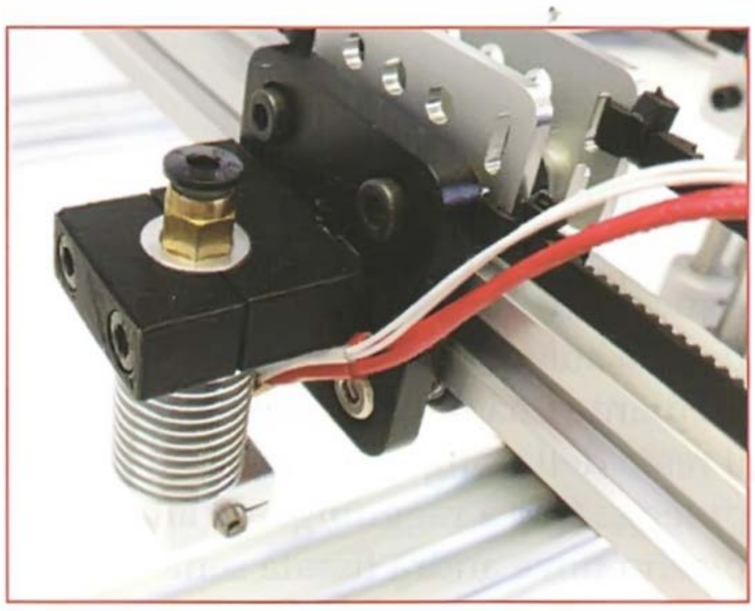


***Fig. 9** - La testina montata sul suo carrello.*

dell'asse Z e accorciate la linguetta a 23 mm. Fissatelo al profilato in modo che i relativi contatti distino circa 2 mm dall'angolare collocato sotto.
Il posizionamento degli switch finecorsa è mostrato nella **Fig. 11**.
Passiamo adesso al piatto di stampa: disponetelo (maneggiandolo con molta cura!) con la svasatura dei fori rivolta verso il basso ed inserire in ciascuno di essi (dal basso verso l'alto) una vite. Successivamente dovete inserire su ciascuna vite una rondella piana 5x20 ed una molla 2x15x26. Prendete le quattro squadrette in alluminio ad angolo retto ed applicatele ai due profilati in alluminio da 627mm, in modo che un lato del piatto disti di 80mm da una delle estremità dei profilati stessi.
Il blocco composto dal piatto montato sui due profilati in alluminio va ora applicato alla base del telaio della stampante mediante apposite viti.
Una volta montato il gruppo, bisogna agire sulle quattro viti di regolazione (la **Fig. 12** nel mostra una) del piatto in vetro per metterlo approssimativamente in piano, serrandole quanto basta per mettere leggermente in tensione le molle ed evitare eccessive oscillazioni (la regolazione fine e definitiva verrà effettuata durante il collaudo della macchina in base a come avviene la stampa).
Fatto anche questo, si può preparare il supporto della bobina: inserite il braccio di supporto nell'apposita fessura della relativa staffa, quindi far scorrere quest'ultima verso il basso fino a mandarla in battuta; poi bisogna fissare il supporto rocchetto al telaio ad un'altezza di circa 150 mm dalla base.
Adesso passiamo al montaggio del meccanismo di alimentazione (definito anche Feeder): fissate al motore NEMA17 da 2,5 A (tipo lungo) il supporto per bowden e inserite sull'albero motore il trascinatore, avendo cura di allineare il grano di fissaggio di quest'ultimo con la parte piana del perno stesso. Fissate il trascinatore sull'albero motore dopo aver allineato la sua gola dentata con i fori guida del filamento presenti su supporto bowden. Applicate il cuscinetto sull'apposito semiguscio e montare il relativo semiguscio, quindi avvitate al motore stepper la vite che sporge dal supporto cuscinetto.
Sistemate adesso la guida del filamento, che lo porterà dal motore alla testa di stampa; tale guida è il bowden, da fissare al telaio, inarcato, con l'aiuto di una molla a trazione. Poi fate passare il tubetto in PTFE che fuoriesce dalla testina nell'occhiello superiore della molla e quindi fissate quest'ultima al profilo superiore

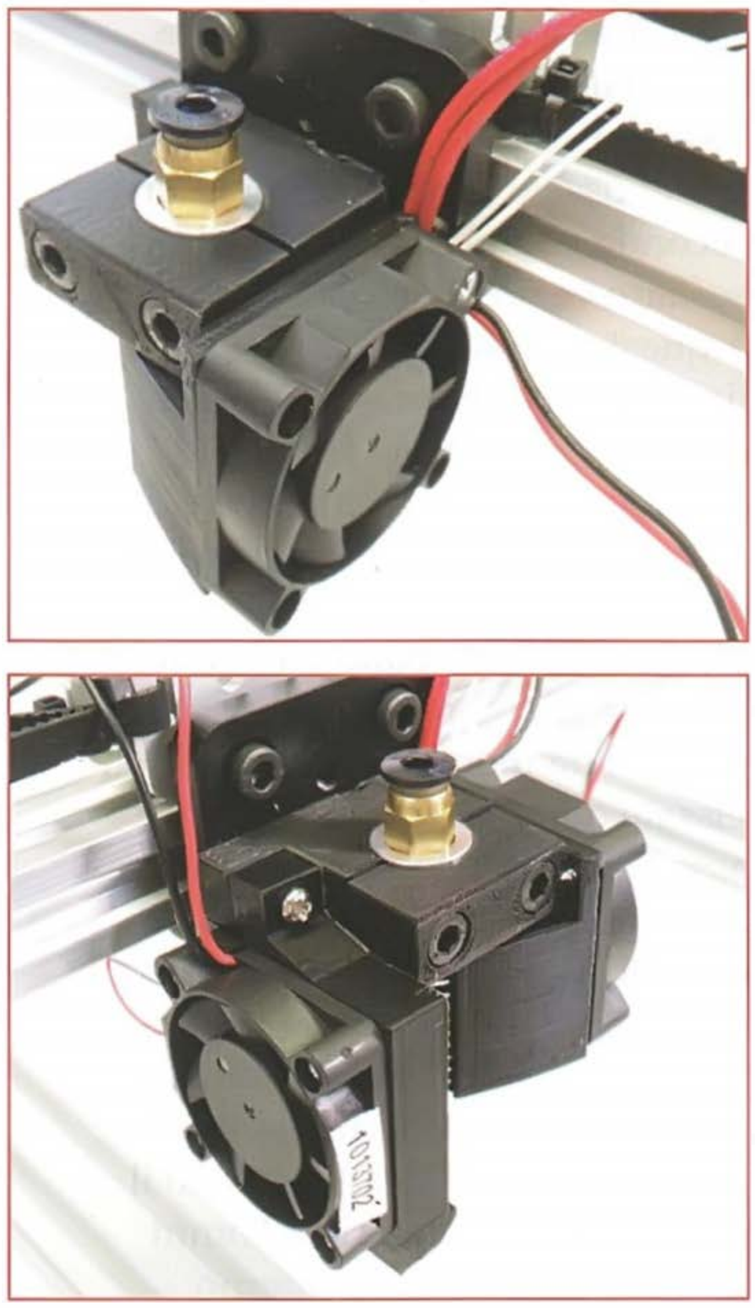
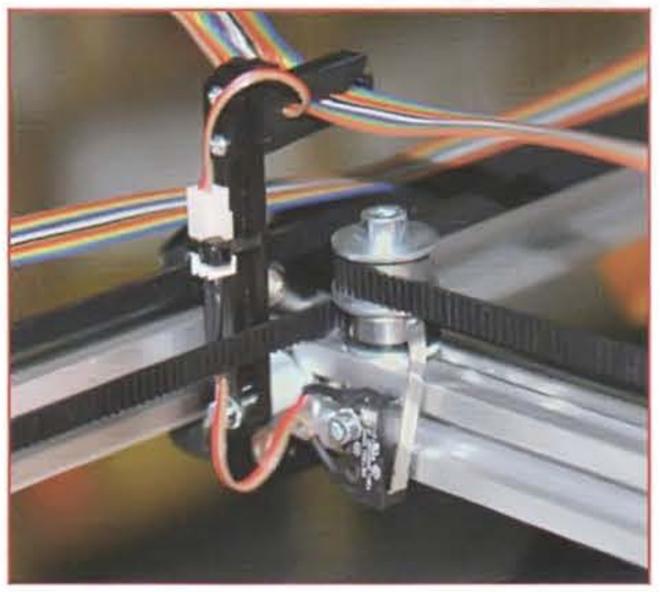
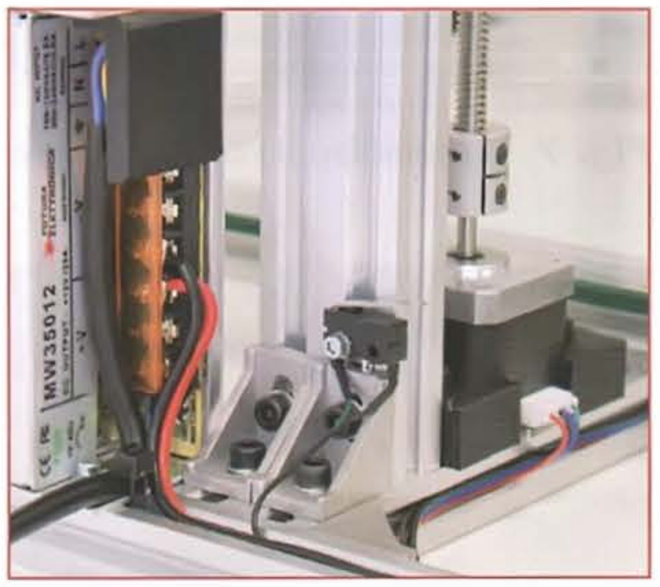


***Fig. 10** - Ventole e convogliatore montati sulla testa di stampa.*





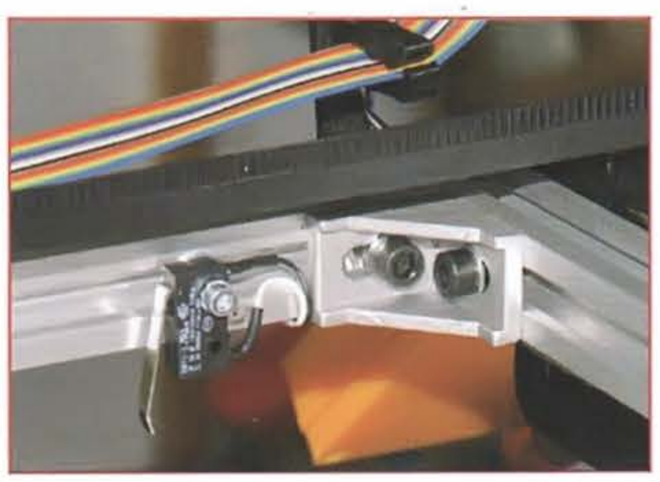
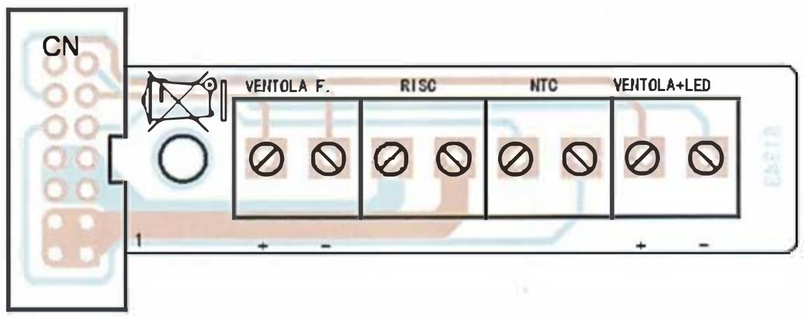


***Fig. 11** Posizione dei microswitch asse Z (sinistra), asse X (sotto) e asse Y sopra).*





Pin connettore CN	Morsetto
1, 2, 3, 4	RISC
5, 6, 7, 8	RISC
9	NTC
10	NTC
11	+ ventola
12	- ventola
13	- LED
14	+ LED

del telaio allineandola con la cava sinistra del montante.
Fissate il bowden sulla guida del carrello Z (nella cava di sinistra) tramite le viti e infine inserite nel raccordo il tubetto in PTFE proveniente dalla testina di stampa (spingerlo bene fino in fondo per bloccarlo in sede). Il gruppo di alimentazione dovrà essere come mostrato nella **Fig. 13** (a pagina seguente).

A questo punto la meccanica è stata completata e potete passare all'elettronica, che consta della scheda controller e del PCB per i cablaggi della testa di stampa, perché in questa stampante abbiamo previsto, per rendere il cablaggio più ordinato, di collegare i cavi della testina e quelli delle ventole a una basetta, sulla quale sono presenti delle morsettiere previste allo scopo e sul lato corto della quale abbiamo un connettore flat. La basetta, come mostrato nella traccia riportata nell'apposito riquadro e nella tabella annessa in cima a questa pagina, realizza le connessioni tra il connettore e le morsettiere. Bisogna quindi fissare tale basetta, tramite l'apposito supporto, al profilo centrale del telaio portatesta come, visibile nella **Fig. 14**, la quale riporta il cablaggio della testa di stampa.
La scheda controller viene fornita già montata e pronta con il bootloader e il firmware caricati, tuttavia è possibile autocostruirla seguendo le indicazioni fornite nell'articolo dedicatole, nel fascicolo n° 175; la scheda acquistata da Futura Elettronica è già programmata. Notate che il firmware di base è scritto per il funzionamento senza riscaldatore del piatto di stampa Z che è sempre attivo, grazie all'istruzione #define DISABLE_Z settata a "false", per bloccare i due motori durante il funzionamento e non farli muovere, onde evitare rotazioni involontarie, dovute a vibrazioni.
Quanto al cablaggio, dopo averla bloccata con dei distanziali sul telaio della macchina, ricordate di connettere i fili dei motori passo-passo ai connettori corrispondenti utilizzando i cavi flat già pronti. La corrispondenza fra motori e connettori è la seguente: il motore dell'asse X va su X-MOTOR, quello dell'asse Y va collegato ad Y-MOTOR, quello che alza e abbassa la testa di stampa va su Z-MOTOR, mentre quello che fa avanzare il filo nell'estrusore va su E-MOTOR. Per i finecorsa, quelli degli assi X ed Y vanno rispettivamente ad XSTOP e YSTOP, mentre quello del carrello della testa di stampa

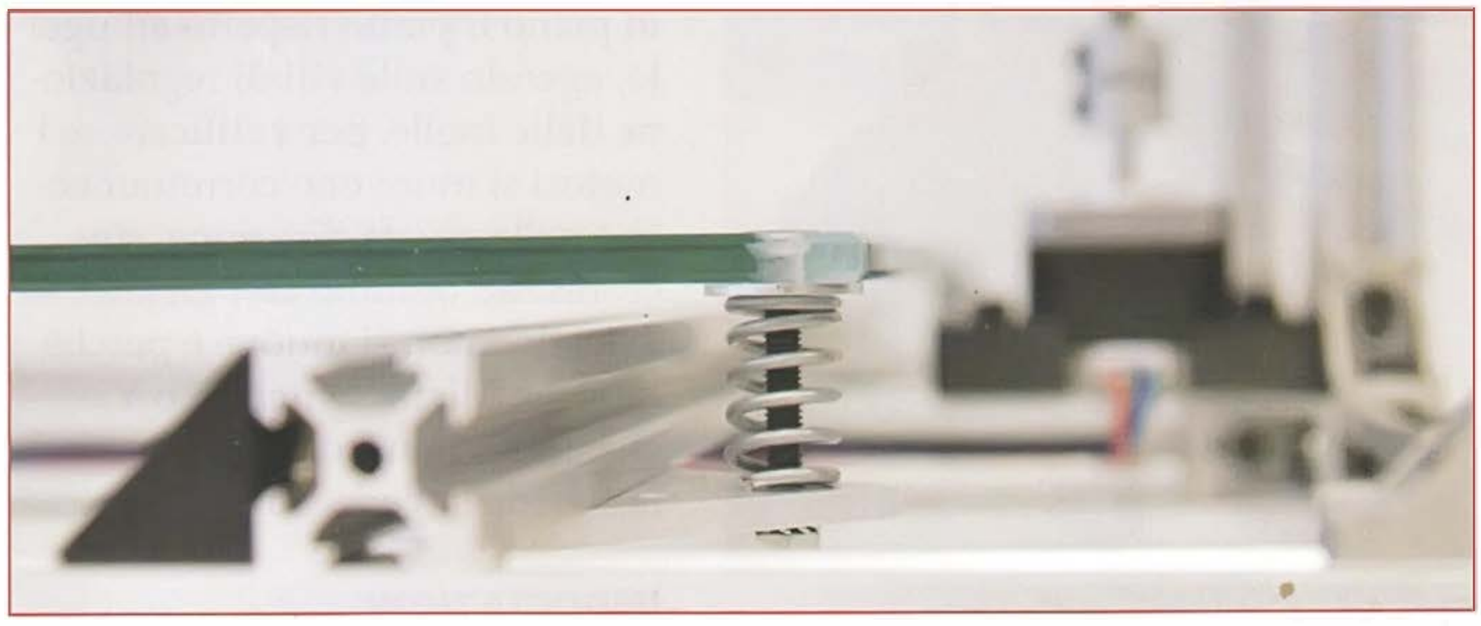


***Fig. 12** - Sospensione e vite di regolazione del piatto su un angolo.*

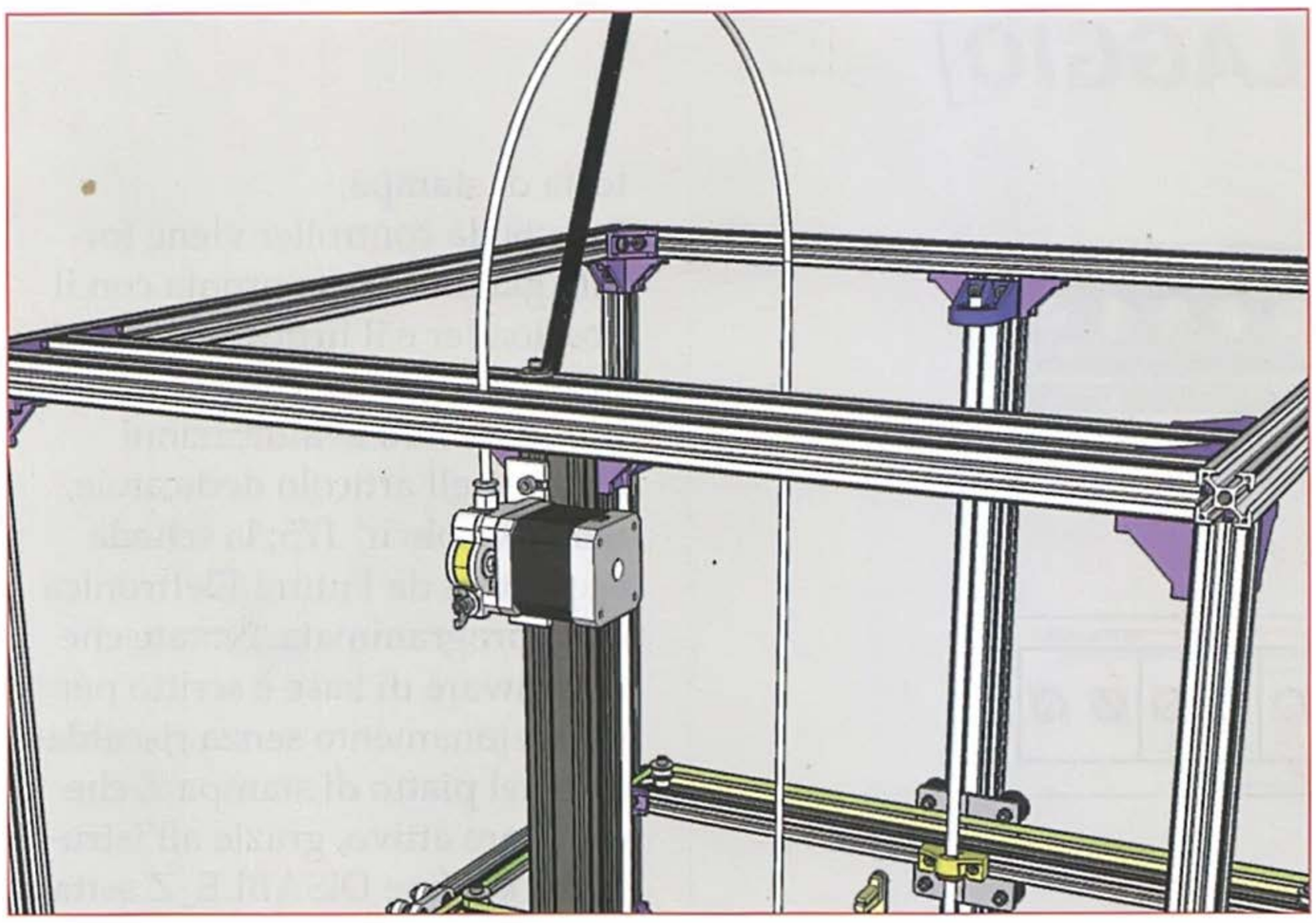


Fig. 13 *- Il sistema di alimentazione del filamento.*

va connesso a ZSTOP.
Il riscaldatore dell'estrusore deve essere collegato, senza badare alla polarità, ai contatti HEATER1, mentre l'NTC corrispondente va a THERM1.
Fissate in fondo al fianco sinistro del telaio l'alimentatore da rete e collegate alla morsettiera il cavo di rete in questo modo:

- cavetto **GIALLO/VERDE** –> morsetto di terra;
- cavetto **BLU** –> morsetto **N**;
- cavetto **MARRONE** –> morsetto **L**;

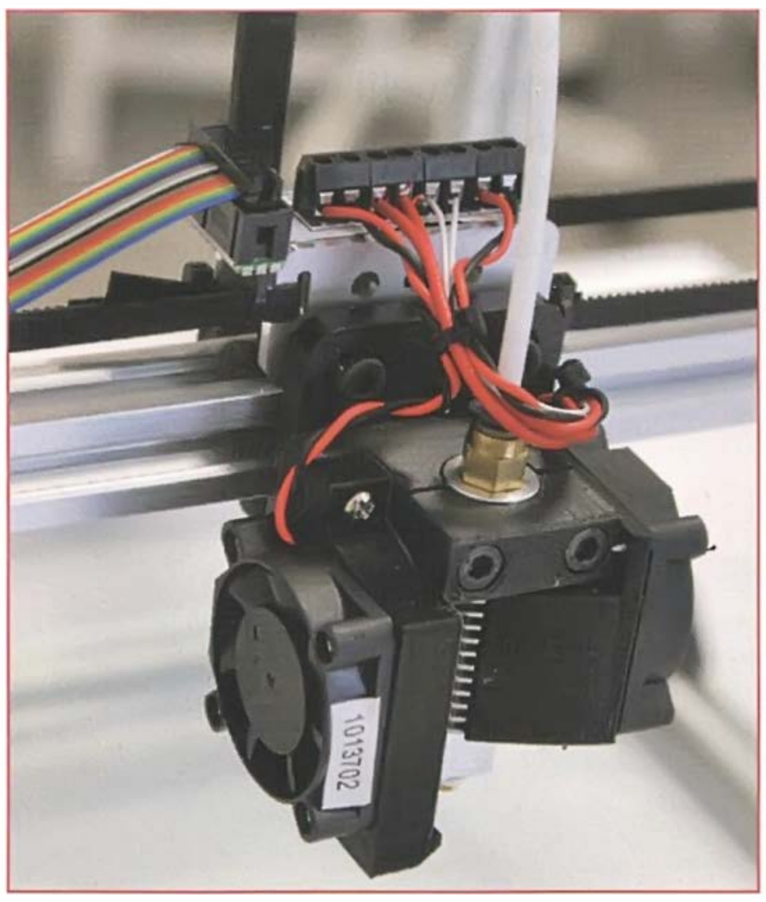


Fig. 14 *- Cablaggio della testa con la scheda a morsetti.*

Connettete poi l'uscita dell'alimentatore a un cavetto rosso/nero che all'estremità riporti un plug da inserire nella scheda controller; il filo rosso va al morsetto +V e il nero al morsetto -V.
Il cavo di rete conviene fissarlo alla parte inferiore del modulo di alimentazione mediante una fascetta 2,5x200, che servirà a proteggerlo dagli strappi accidentali; i morsetti sottoposti alla tensione di rete è utile proteggerli con un coperchietto in plastica (specifico per quel tipo di alimentatore) che previene il contatto accidentale.
Fatto ciò, la stampante è pronta: collegata la scheda controller al computer mediante un cavo USB A/mini USB potete avviare la macchina con una stampa di test, dopo aver messo perfettamente in piano il piatto rispetto all'ugello, agendo sulle viti di regolazione delle molle, per verificare se i motori si muovono correttamente e nella giusta direzione, sincronizzati ognuno con gli altri; se la testina non si muove è perchè qualche Micro Switch (X o Y o entrambi) non è collegato, quindi la scheda la vede a finecorsa.

IMPOSTAZIONI

Per ottenere il funzionamento ottimale bisogna tarare i driver della scheda; la procedura è questa: a stampante alimentata, prendete un multimetro impostato a 2 Vcc fondo-scala e un cacciavite piatto piccolo in ceramica (è possibile utilizzare un cacciavite metallico ma bisogna stare attenti a toccare solamente il trimmer, onde evitare cortocircuiti).
Tenete il puntale nero del tester sulla massa della scheda e il rosso sulla piazzola posizionata sopra al chip del primo driver, quindi regolate il trimmer fino a leggere nel multimetro 0,41V.
Fate lo stesso con gli altri driver, ricordando che quello dell'asse Z va tarato a 0,65V perché deve alimentare due motori in parallelo.

CONCLUSIONI

Bene, con questo abbiamo concluso, almeno per ora. Nella seconda e ultima puntata vi proporremo alcuni add-on, tra cui l'installazione (e personalizzazione dell'attività) del riscaldatore per il piatto di stampa e di un pannello di controllo a display per l'utilizzo della macchina in modo autonomo. Vedremo anche come si può ritoccare la modalità micro-step predefinita e intervenire sul firmware della scheda controller.



per il MATERIALE

La stampante in KIT 3D 40X40X40 cm è in vendita presso Futura Elettronica al prezzo di euro 1199,00 Euro. Il prezzo si intende IVA compresa.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775
http://www.futurashop.it

Concepito per la serie di telecontrolli di Futura Elettronica, consente di mantenere il circuito in funzione, grazie a una batteria agli ioni di litio, in assenza dell'alimentazione principale.

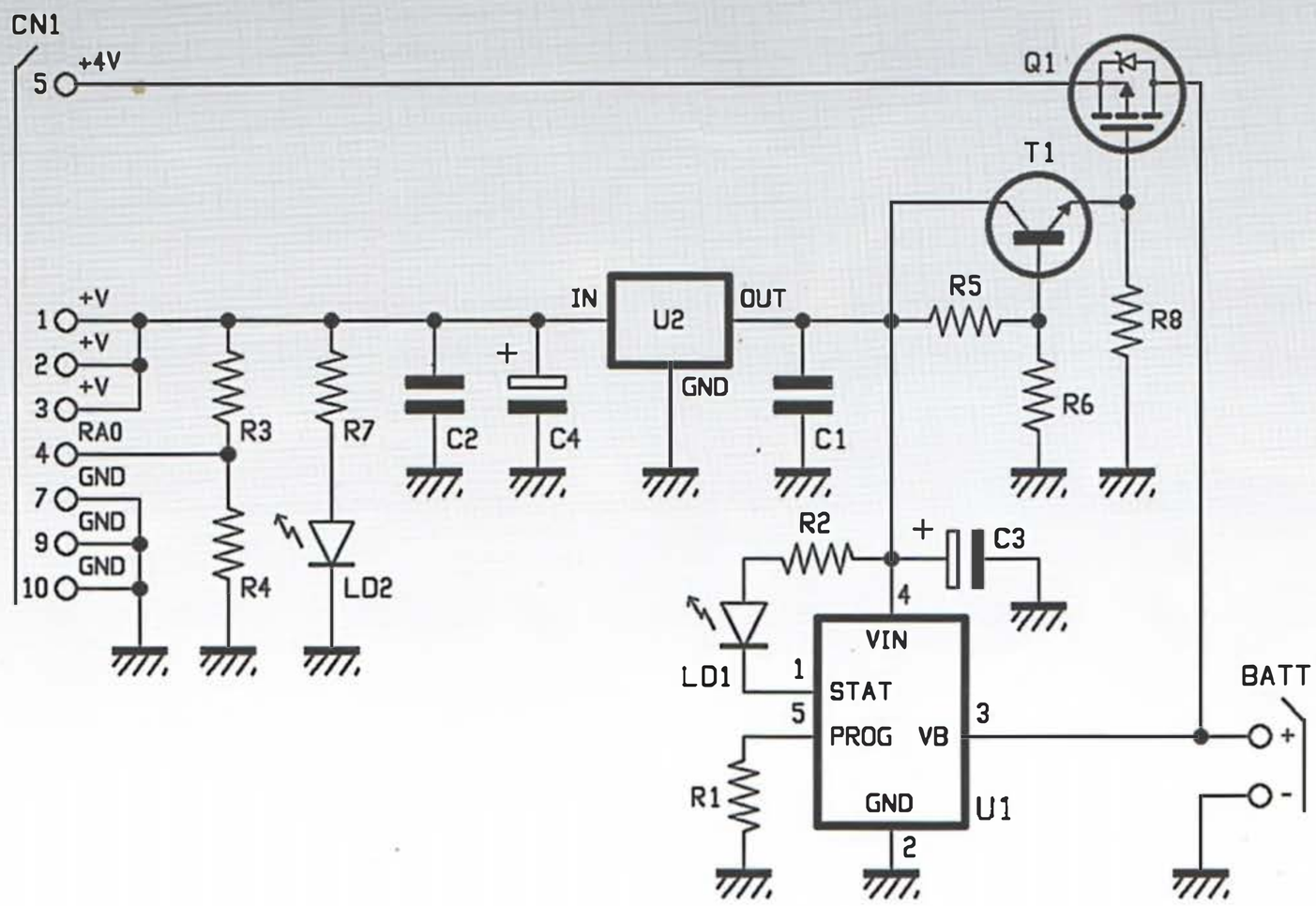


GRUPPO DI CONTINUITA' PER TELECONTROLLI GSM

di ALESSANDRO SOTTOCORNOLA

Sicuramente ricorderete la nostra fortunata serie di telecontrolli GSM siglata TDG, composta da una scheda universale che poteva essere più o meno popolata di componenti specifici in base a una delle quattro applicazioni realizzabili; si trattava di un progetto modulare nel quale era altresì prevista la possibilità di sostituire il modulo GSM/GPRS per superare le problematiche di obsolescenza dei componenti o di approvvigionamento che negli anni potevano verificarsi, allo scopo di mantenere lo stesso apparato installato. Essendo alimentati tramite un alimentatore da rete, i TDG avevano il limite di non funzionare in caso di blackout, sebbene nel firmware avevamo previsto la funzione di ripristino automatico e salvataggio delle impostazioni di sistema in memoria permanente. Nel tempo ci avete fatto notare che rendere funzionanti i telecontrolli prescindendo dalle vicende della rete elettrica sarebbe stato un apprezzato add-on,

[schema **ELETTRICO** gruppo di continuità]



per questo ci siamo messi all'opera e abbiamo realizzato qualcosa che potremmo considerare un gruppo di continuità. Il circuito, semplicissimo, consta di un caricabatteria che mantiene carica una cella agli ioni di litio, la quale all'occorrenza alimenta il TDG del caso, portando l'alimentazione dritta nel cuore del sistema.

SCHEMA ELETTRICO

Scopriamo il funzionamento del gruppo di continuità riferendoci allo schema elettrico che pubblichiamo in queste pagine, insieme allo schema elettrico dei telecontrolli della serie TDG; tale schema ci serve per comprendere il significato delle connessioni riportate sul connettore CN1, che funge sia da collegamento elettrico tra i due circuiti stampati, sia da sostegno meccanico per il PCB del gruppo di continuità. Infatti CN1 del gruppo di continuità si introduce nel connettore EXP dei TDG (**Fig. 1**), che al tempo in cui presentammo la serie di telecontrolli avevamo volutamente trascurato, omettendolo nello schema elettrico e non prevedendone il montaggio perché riservato a futuri sviluppi. Dunque dallo schema vediamo che CN1 riporta 10 contatti, disposti su due file a passo 2,54 mm da 5 poli ciascuna; i contatti 7, 9 e 10 portano la massa comune (e il negativo di alimentazione) mentre 1, 2 e 3 fanno arrivare al circuito la stessa tensione prelevata dal punto +V del telecontrollo, che corrisponde all'anodo del diodo di protezione dall'inversione di polarità posto sulla linea positiva di ingresso dopo il fusibile. Dai contatti 1, 2, 3, preleviamo quindi l'alimentazione principale che farà funzionare il resto del circuito e, quando presente, farà accendere il LED di segnalazione LD2 (preceduto dalla resistenza che ne limita la corrente a valori non distruttivi); con tale tensione, ben filtrata dai condensatori C2 e C4, viene alimentato il regolatore a tre terminali U2 (si tratta di un LD1117-50, in versione SMD), il quale fornisce dalla propria uscita una componente continua e ben stabilizzata al valore di 5 volt, la quale, filtrata dal condensatore C1 e dall'elettrolitico C3, alimenta lo stadio caricabatteria.

Quest'ultimo fa capo a un integrato specifico per la carica di un singolo elemento Li-ion da 3,6-3,7 volt (4,2 V a piena carica); l'integrato è l'MCP73831T, un SMD incapsulato in un case SOT-23 per montaggio superficiale e prodotto dalla Microchip.

La sezione caricabatteria accetta in ingresso (piedini 4, positivo, e 2, ossia GND) tipicamente 5 volt, che in questo caso provengono da U2 e sono quindi stabilizzati; comunque sappiate che il regolatore di carica MCP73831T (U1) accetta all'ingresso tensioni dell'ordine dei 3,75÷6 volt e fornisce all'uscita

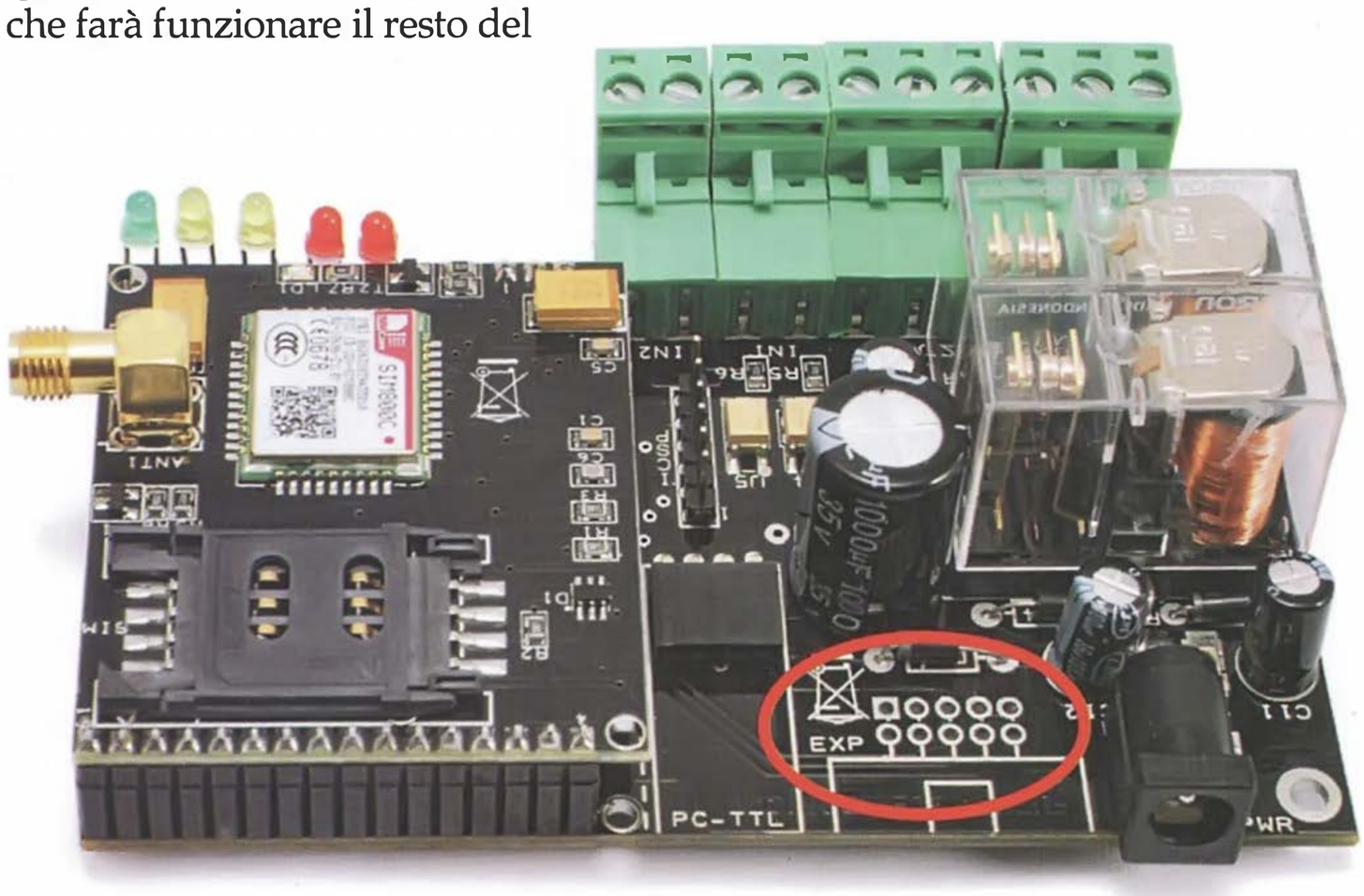
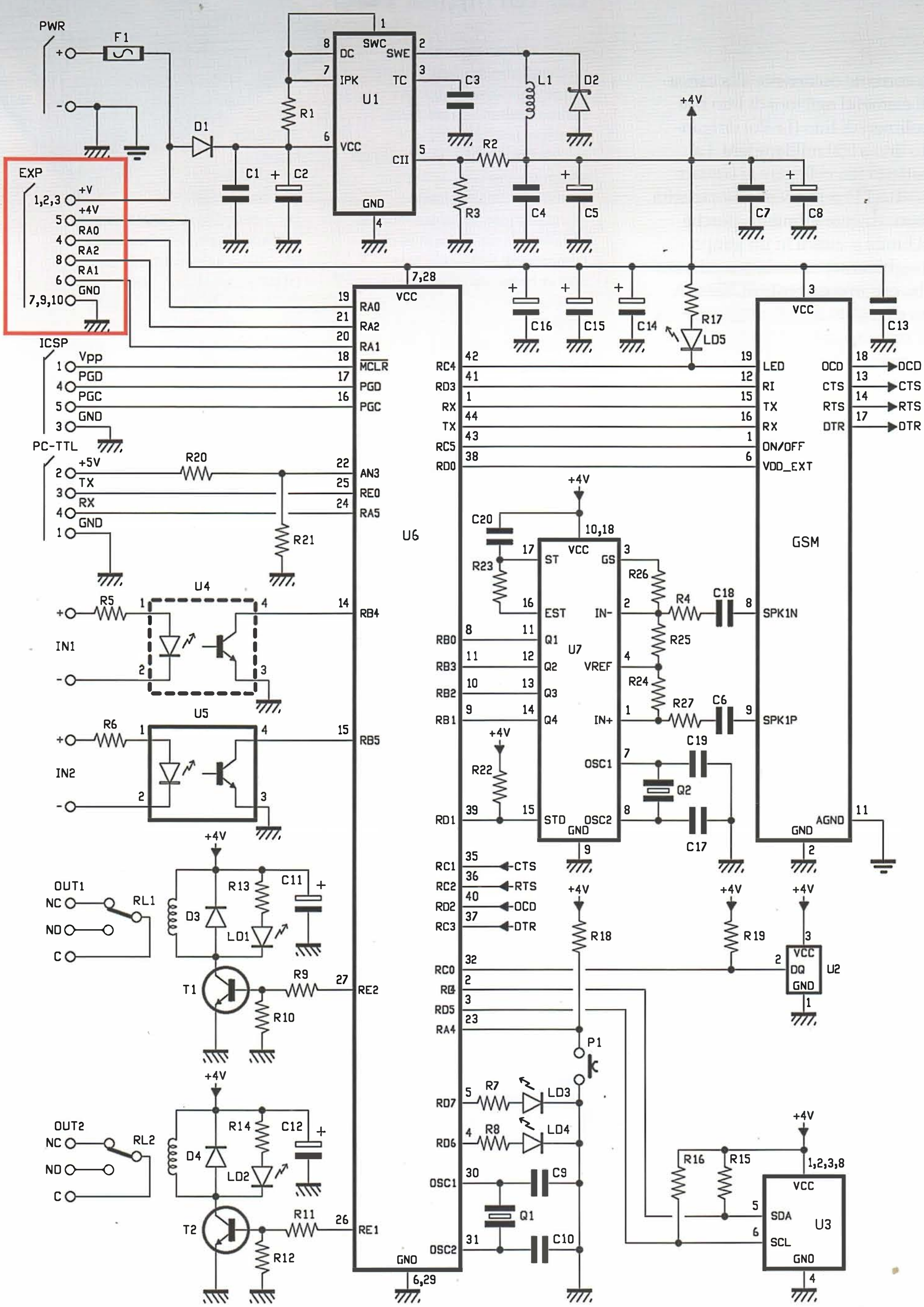


***Fig. 1** - L'ovale rosso evidenzia le piazzole del telecontrollo in cui saldare il connettore 2x5 vie EXP dove innestare il PCB del gruppo di continuità.*


PWR
F1
+
-
D1
R1
U1
SWC
SWE
DC
IPK
TC
VCC
CII
GND
C1
C2
C3
L1
D2
R2
R3
C4
C5
+4V
C7
C8
EXP
+V
1,2,3
+4V
5
RA0
4
RA2
8
RA1
6
GND
7,9,10
7,28
VCC
C16
C15
C14
R17
LD5
C13
ICSP
Vpp
1
PGD
4
PGC
5
GND
3
PC-TTL
+5V
2
TX
3
RX
4
GND
1
R20
R21
U6
19
RA0
21
RA2
20
RA1
18
MCLR
17
PGD
16
PGC
22
AN3
25
RE0
24
RA5
14
RB4
15
RB5
27
RE2
26
RE1
6,29
GND
U4
R5
IN1
U5
R6
IN2
OUT1
NC
NO
C
RL1
D3
R13
C11
LD1
T1
R9
R10
OUT2
RL2
D4
R14
C12
LD2
T2
R11
R12
42
RC4
41
RD3
1
RX
44
TX
43
RC5
38
RD0
35
RC1
CTS
36
RC2
RTS
40
RD2
OCD
37
RC3
DTR
32
RC0
2
RC4
3
RD5
23
RA4
P1
5
RD7
R7
LD3
4
RD6
R8
LD4
30
OSC1
C9
Q1
31
OSC2
C10
RD1
39
15
STD
OSC2
GND
9
+4V
C20
10,18
VCC
17
ST
GS
3
R23
16
EST
IN-
2
R26
R4
C18
8
SPK1N
R25
U7
8
11
RB0
Q1
11
12
RB3
Q2
VREF
4
10
13
RB2
Q3
R24
9
14
RB1
Q4
IN+
1
R27
C6
9
SPK1P
+4V
R22
C19
7
OSC1
Q2
8
C17
GND
2
R18
R19
+4V
+4V
3
VCC
2
DQ
GND
1
U2
3
LED
RI
TX
RX
ON/OFF
VDD_EXT
OCD
CTS
RTS
DTR
18
13
14
17
DCD
CTS
RTS
DTR
GSM
AGND
11
GND
2
+4V
R16
R15
1,2,3,8
VCC
5
SDA
6
SCL
GND
4
U3

La famiglia **TDG**



Si tratta di telecontrolli modulari che realizzano varie funzioni utilizzando per la comunicazione la rete GSM, che abbiamo pubblicato a partire dal n° 146; per modulari s'intende che sono sviluppati usando un unico circuito stampato (con un unico modulo cellulare) la cui dotazione di componenti viene adattata alla funzione da realizzare. I telecontrolli sono l'apricancello GSM (TDG134, pubblicato nel fascicolo n° 147), il telecontrollo 2IN 2 OUT (TDG133, fascicolo n° 148), il termostato GSM per il controllo di riscaldamento e condizionamento (fascicolo n° 149) e il telecontrollo con DTMF (fascicolo n° 150). Tutti, condividendo il PCB, possono essere dotati del gruppo di continuità, a patto di montare il connettore EXP.

la corrente occorrente alla carica di elementi agli ioni di litio o ai polimeri di litio (Li-Po) erogando fino a 550 milliampere. La batteria (da collegare ai contatti +/- BATT) può avere una capacità teoricamente illimitata, perché al limite si carica in un tempo lunghissimo; tuttavia considerate che con una corrente di 550 mA un elemento da 550 mAh si carica in un'ora, uno da 1.100 mAh per la carica richiederà 2 ore, uno da 5,5 Ah necessiterà di 10 ore e via di seguito. Questo, ammettendo che non vi siano perdite rilevanti nella batteria; nella pratica il tempo è più lungo.
Nel nostro circuito l'integrato funziona nella configurazione canonica, che vede il diodo luminoso LD1 alimentato (mediante la resistenza R2) dall'uscita STAT, la quale si pone a livello logico basso durante la carica, mentre rimane a livello alto a fine carica; la stessa si porta ad alta impedenza (open) quando l'MCP73831T è in shutdown o se all'uscita VBAT non risulta attualmente collegata alcuna batteria. Il piedino VBAT (pin 3) è la linea cui si connette il positivo della batteria, il cui negativo termina a massa.
L'integrato opera la carica a corrente e tensione costante. La corrente di carica (Ireg) viene impostata con la resistenza collegata al piedino 5 (R1 nel nostro caso), al cui valore è legata dalla relazione:

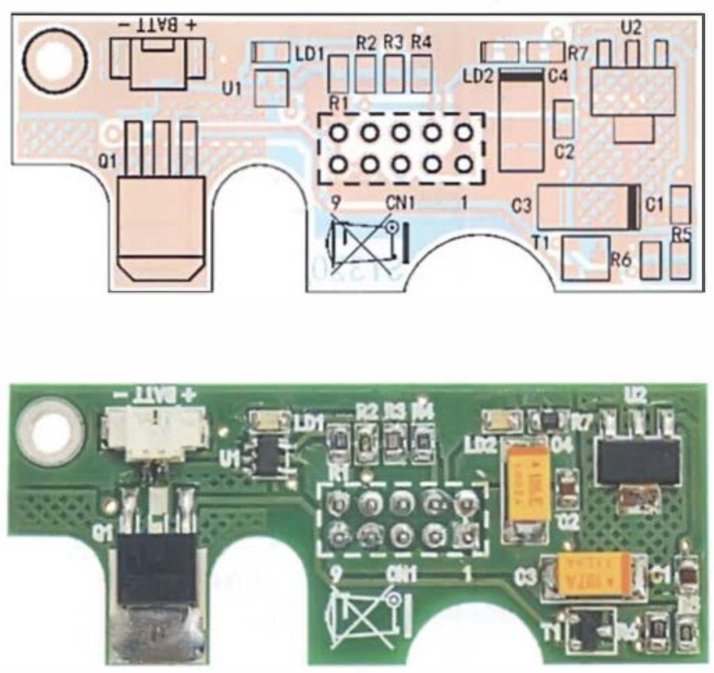
Ireg = 1.000/R

dove il valore della R risulta in ohm se la corrente Ireg è espressa in A. Ad esempio, con un resistore da 4,7 kohm si ottiene una limitazione a 212 mA, mentre con una R di 22 kohm (il nostro caso) la corrente vale circa 45,4 mA. Se il piedino 5 viene aperto, l'integrato si porta in shutdown e assorbe appena 2 µA; il piedino può quindi essere usato come enable (funzione che qui non viene gestita).
La carica si svolge in diverse fasi: una volta connessa la batteria all'uscita Vb (piedino 1) se la tensione rilevata risulta minore di una certa soglia (tipicamente intorno al 70 % del valore di piena carica) viene avviato il precondizionamento ad un valore di corrente modesto, che serve a preparare la carica rapida; quest'ultima si avvia quando la tensione ai capi della batteria supera la predetta soglia e viene effettuata ad un valore di corrente stabilito dalla resistenza collegata al piedino 5.
Non appena la batteria raggiunge una tensione pari a quella di regolazione impostata nel chip, smette la carica rapida e inizia quella di mantenimento, che punta a stabilizzare la tensione acquisita a 4,2 V, 4,35 V, 4,4 V o 4,5 V a seconda della versione dell'integrato, che nel nostro progetto è la MCP73831T-2, la quale regola a 4,2 V.

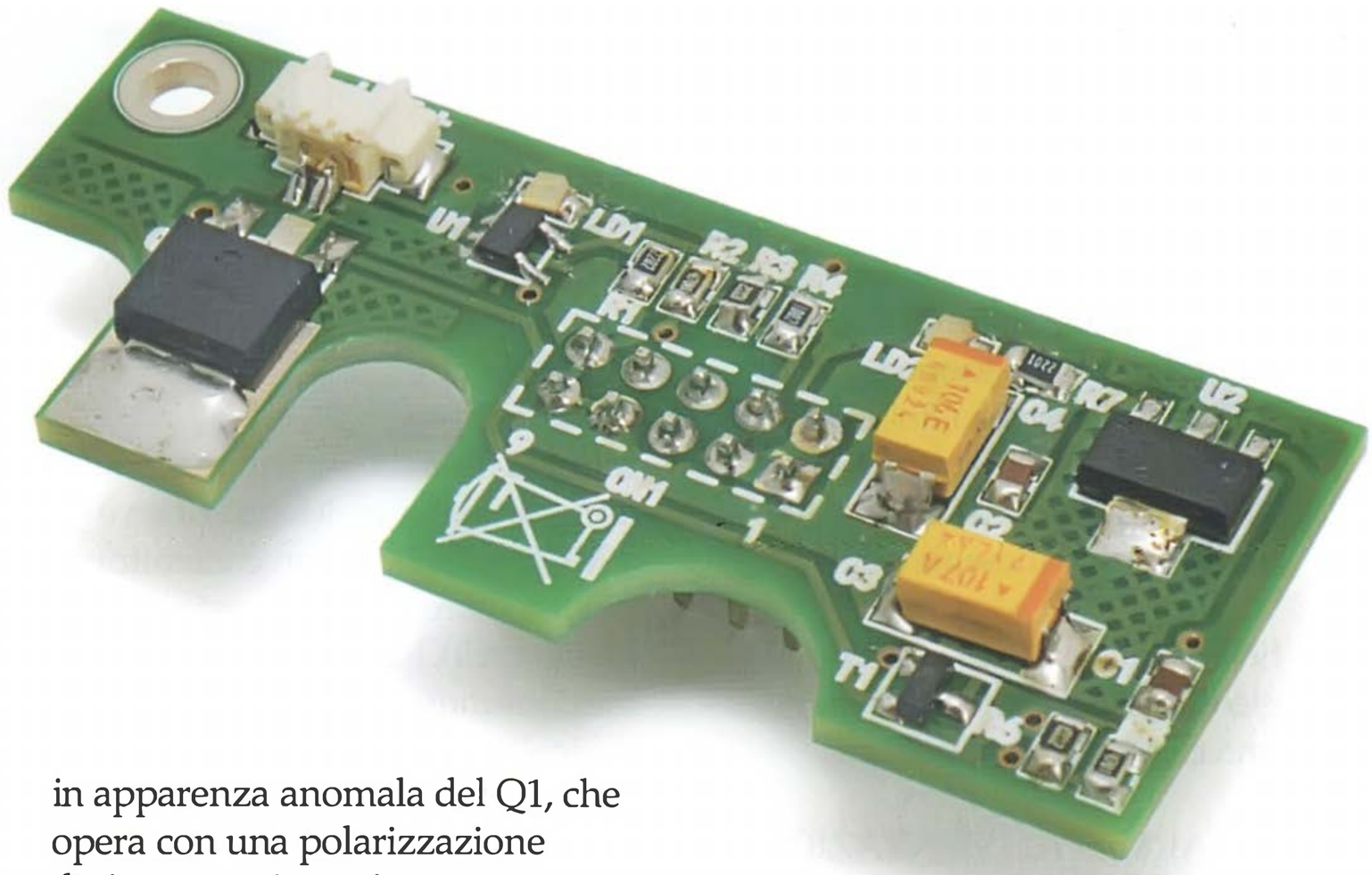
[piano di **MONTAGGIO**]

Elenco Componenti:

C1, C2: 100 nF ceramico (0805)
C3: 100 µF 10 VL elettrolitico
C4: 10 µF 25 VL elettrolitico
LD1: LED rosso (0805)
LD2: LED verde (0805)
R1: 22 kohm (0805)
R2: 470 ohm (0805)
R3: 100 kohm (0805)
R4: 10 kohm (0805)
R5: 4,7 kohm (0805)
R6: 10 kohm (0805)
R7: 2,2 kohm (0805)
Q1: RSD046P05TL
T1: BC817
U1: MCP73831T
U2: LD1117-50
BATT: Connettore JST 2 vie passo 1.25 mm



Varie:
- Strip maschio 2x5 vie
- Strip femmina 2x5 vie
- Ciruito stampato S1320 (56x23 mm)



Le diverse versioni e quindi tensioni di regolazione dell'integrato, sono state volute dal costruttore per venire incontro alle esigenze di carica dei nuovi modelli di batterie al litio, che possono lavorare con tensioni anche superiori a 4 volt in base alla loro chimica: per esempio le LiPo a piena carica arrivano anche a 4,3 V.
Finora abbiamo spiegato come funziona il circuito quando è alimentato dal telecontrollo; ma in realtà a noi serve quando manca l'alimentazione, perché il circuito ha una funzione mutua, ossia quando c'è corrente dalla rete vi attinge per caricare la batteria e quando manca è esso a cedere l'energia immagazzinata. Infatti, se all'ingresso +/- PWR del telecontrollo manca alimentazione (condizione evidenziata dallo spegnimento del LED LD2), la batteria BATT eroga corrente al punto +4V del connettore CN1 attraverso il diodo interno al MOSFET; guardando lo schema elettrico della serie TDG, il contatto 5 del connettore EXP porta alla linea +4V che fa capo all'uscita del DC/DC MC34063 e che alimenta il microcontrollore e il modulo cellulare, perciò in assenza di alimentazione principale tale linea è "sostenuta" dalla batteria, la quale eroga circa 4 V a piena carica e scende gradualmente fino a 3,5 volt man mano che perde la propria energia.
Al ritorno dell'alimentazione principale, il caricabatterie viene nuovamente alimentato dall'U2 e l'U1 e riprende a caricare l'elemento agli ioni di litio.
Prima di procedere soffermiamoci un istante sullo stadio di commutazione a MOSFET, che serve a bloccare l'erogazione di corrente dalla batteria al telecontrollo durante la carica, vale a dire quando non serve perché è presente l'alimentazione principale sul jack +/- PWR: noterete la connessione in apparenza anomala del Q1, che opera con una polarizzazione drain-source invertita, ma non allarmatevi, perché tutto funziona ugualmente. Per l'esattezza, in presenza della tensione di rete il regolatore integrato fornisce 5 volt al partitore resistivo R5-R6, che polarizza la base del transistor T1, operante in configurazione a inseguitore di emettitore; questo, fornisce al gate del MOSFET tramite il proprio emettitore un potenziale di poco superiore ai 4,3 volt, sufficiente a mantenere interdetto il Q1 ad impedire il flusso di corrente dal contatto +4V al positivo della batteria, cosicché l'elemento agli ioni di litio viene caricato esclusivamente dall'U1, come dev'essere per assicurare una corretta carica.
Il diodo di protezione interno al MOSFET non conduce in alcuna delle fasi di carica, giacché anche a cella completamente carica (tensione di 4,2V tra i contatti BATT) la tensione di soglia non viene raggiunta.
Quando viene meno l'alimentazione principale e il regolatore smette di fornire i propri 5 volt, il transistor NPN non è più polarizzato in base e va in interdizione, allorché la sua resistenza di emettitore porta a massa il gate del MOSFET scaricandone la capacità parassita.
Ora Q1 conduce pienamente e la corrente può percorrere il suo canale, seppure in senso anticonvenzionale, ossia da drain a source (nei MOS a canale P dovrebbe essere il contrario). Fin quanto la resistenza presentata tra drain e source è tale da determinare una caduta Vds negativa minore della tensione di soglia del diodo interno di protezione, quest'ultimo viene bypassato.
Notate che la conduzione inversa è possibile in virtù della bassissima resistenza elettrica del canale, tale da rendere trascurabile la tensione tra drain e substrato (connesso internamente al source) del MOSFET; se così non fosse, la giunzione formata tra la zona diffusa di drain e il substrato entrerebbe in conduzione.
Chiusa questa parentesi, torniamo all'analisi circuitale dicendo che la presenza e l'assenza della tensione di rete, oltre ad essere segnalate da LD2, vengono rilevate dal microcontrollore che governa il telecontrollo, attraverso la linea RA0, cui fa capo il contatto 4 del connettore CN1; RA0 è internamente assegnata all'A/D converter del PIC e tramite essa il firmware legge il valore della tensione riportata dal partitore R3-R4, dimensionato per fornire, con 12V tra +V e GND, circa 1,15 volt. Dunque, leggendo questa tensione il microcontrollore ritiene che sia presente la rete e senza sa che è in corso un blackout.
Un aggiornamento firmware

provvederà, nel caso il blackout si protragga per oltre 2 secondi, all'invio di un ciclo di SMS di notifica dell'accaduto; i comandi da impartire allo scopo sono:
PWR0 = disattiva la notifica della mancanza di alimentazione;
PWR1 = attiva la notifica della mancanza di alimentazione.

Sono sempre previsti un messaggio di perdita alimentazione ed uno di ripristino; il primo è:

ALLARME!! ALIMENTAZIONE ASSENTE

e il secondo contiene il testo:

ALIMENTAZIONE RIPRISTINATA

REALIZZAZIONE PRATICA

Bene, descritto lo schema elettrico vediamo come costruire il circuito del gruppo di continuità; per prima cosa dovete preparare lo stampato, seguendo le tracce (si tratta di un PCB a doppia ramatura) pubblicate nel nostro sito web *www.elettronicain.it*. Ottenuto il c.s. disponetevi i componenti iniziando dagli integrati. Ricordiamo che per una buona saldatura è consigliabile operare con un saldatore da 20 o 25 watt a punta fine e filo di lega saldante del diametro di 0,5 mm o meno, stagnando le piazzole interessate e poi ricoprendole di flussante per agevolare la saldatura dei terminali; conviene altresì servirsi di una lente d'ingrandimento, meglio se di quelle illuminate da tavolo, per verificare che i componenti siano centrati rispetto alle rispettive piazzole e che non vi siano baffi di stagno a creare cortocircuiti. La disposizione e l'orientamento dei componenti sono visibili nei disegni di montaggio e nelle foto dei prototipi che trovate in queste pagine.
L'MCP73831T, montatelo aiutandovi con il flussante per evitare che lo stagno unisca piedini vicini tra loro (il passo è piccolissimo) e con una lente d'ingrandimento. Meno critico è il MOSFET, che ha un contenitore più familiare, essendo sostanzialmente un TO-220 miniaturizzato ma comunque saldabile senza difficoltà. Completate il circuito con le resistenze, i LED in SMD (la tacca su di esso contrassegna il catodo), i condensatori e il connettore in miniatura per la batteria al litio. Infine, inserite e saldate il doppio pin-strip da 5 poli a passo 2,54 mm per costituire il connettore CN1, che dovrà essere montato sul lato opposto a quello dei componenti. In luogo della doppia fila di pin-strip potete montare un connettore maschio a 2 file per 5 contatti, avente lo stesso passo; in ogni caso il connettore va come in **Fig. 2**.

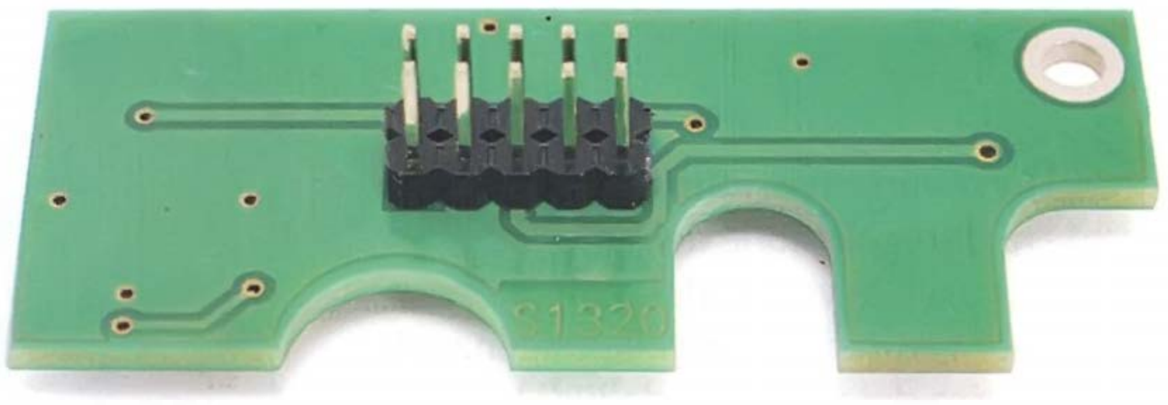


***Fig. 2** - Il connettore 2x5 vie va montato sul lato opposto a quello dei componenti.*

UTILIZZO

Una volta completato, il circuito è pronto all'uso; per l'abbinamento al telecontrollo va applicato sopra alla relativa scheda con il lato componenti rivolto verso l'alto e deve innestarsi con i propri pin (il doppio pin-strip...) nel connettore EXP del telecontrollo stesso. A riguardo ricordiamo che normalmente i sistemi della famiglia TDG non hanno a bordo il connettore femmina EXP; quindi per l'abbinamento al gruppo di continuità dovete prima montarlo. Assemblate le due schede (TDG+gruppo di continuità) inserite il connettore del cavetto della batteria nel corrispondente del gruppo di continuità (BATT) e alimentate il TDG.
In tema di batteria, notate che l'autonomia dipende sia dalla sua capacità, sia dall'utilizzo del telecontrollo; con un evento di comunicazione GSM all'ora, una 1.000 mAh potrebbe durare anche mezza giornata in assenza di rete elettrica.



per il MATERIALE

Il gruppo di continuità per telecontrolli GSM (cod. FT1320M) viene venduto montato e collaudato presso Futura Elettronica al prezzo di Euro 14,00, la batteria ai polimeri di litio da 3,7V-1000mAh (cod. PLIB1000) è disponibile a Euro 6,90.
Si ricorda che la compatibilità del gruppo di continuità è legata alla versione di firmware presente sui telecontrolli. Verificare la presenza del connettore EXP per le schede acquistate in passato, nel caso non sia presente è necessario un aggiornamento. I telecontrolli attualmente venduti sono compatibili con il gruppo di continuità:
-TDG133 telecontrollo GSM bicanale (cod. TDG133) a Euro 96,00.
-TDG134 (cod. TDG134) apricancello GSM, a Euro 92,00.
-TDG138 (cod. TDG138) teleallarme GSM con anti jammer a Euro 96,00.
-TDG139 (cod.TDG139) termostato GSM a Euro 96,00.
-TDG140 (cod. TDG140) Telecontrollo gsm con DTMF a Euro 105,00.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775
http://www.futurashop.it

GESTISCI E CONTROLLA LA TUA CASA CON I NOSTRI SISTEMI GSM

ATTIVA E CONTROLLA DA REMOTO QUALSIASI DISPOSITIVO ELETTRICO O ELETTRONICO: DALLA CENTRALINA ANTIFURTO ALL'IMPIANTO DI IRRIGAZIONE, DALLE LUCI DI CASA AL SISTEMA DI CONDIZIONAMENTO E RAFFRESCAMENTO. GESTIONE TRAMITE CHIAMATA VOCALE O MESSAGGI SMS.



cod. TDG133
€ 96,00

GSM™

Telecontrollo GSM bidirezionale 2 IN / 2 OUT

- 2 uscite a relè con funzionamento monostabile o bistabile e 2 ingressi d'allarme optoisolati.
- Dimensioni 103 x 67 x 28 mm
- Alimentazione 9÷32 Vdc/1 A
- Completo di antenna GSM esterna

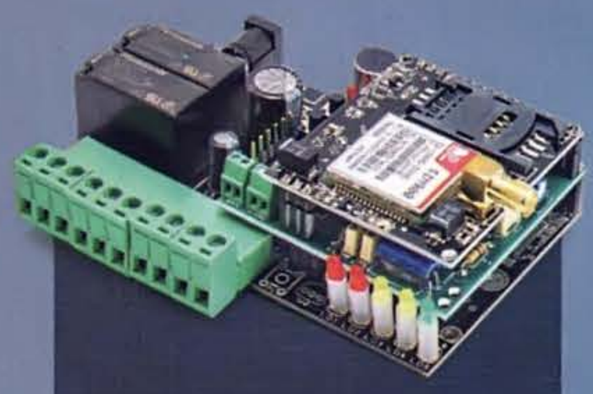
Disponibili anche moduli con funzione di:



TIDIGINO TELECONTROLLO GSM SU PIATTAFORMA ARDUINO cod. FT924M • € 138,00

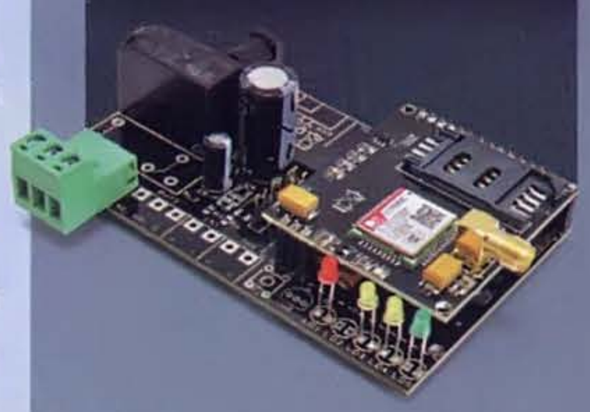
COMBINATORE TELEFONICO GSM CON SINTESI VOCALE TTS cod. TDG145 • € 112,00

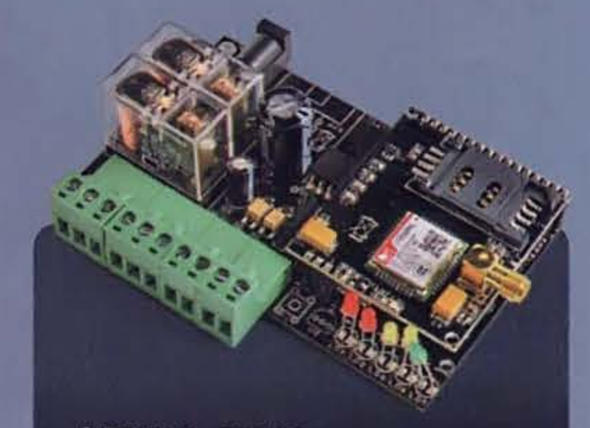




cod. TDG135 • € 128,00
COMBINATORE TELEFONICO GSM

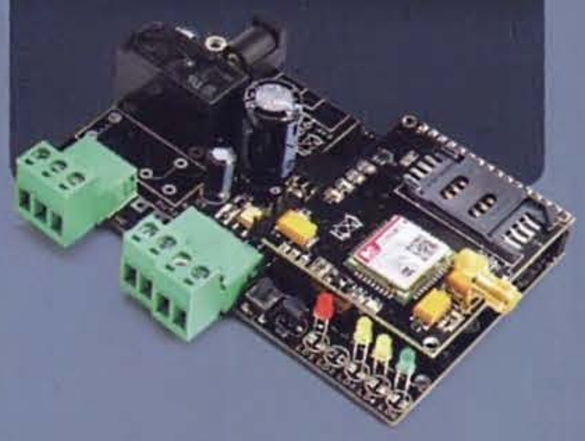
APRICANCELLO GSM cod. TDG134 • € 92,00

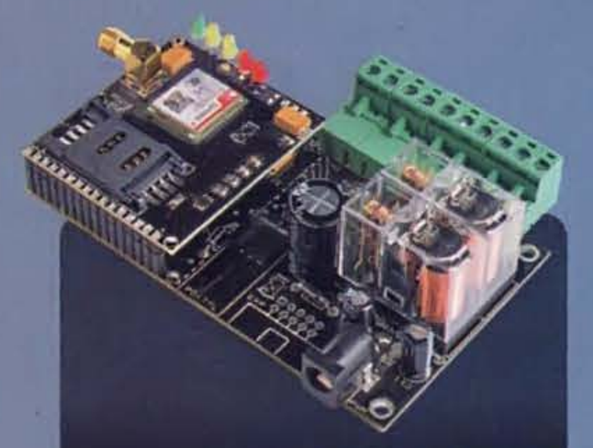




cod. TDG138 • € 96,00
TELEALLARME GSM CON ANTI JAMMER

TERMOSTATO CON CONTROLLO GSM cod. TDG139 • € 96,00





cod. TDG140 • € 105,00
TELECONTROLLO GSM CON COMANDI DTMF

MADE IN ITALY

Prezzi IVA inclusa.

FUTURA ELETTRONICA®

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche e vendita on-line su:
www.futurashop.it

Tutti i telecontrolli vengono forniti già montati e collaudati e sono facilmente configurabili mediante SMS o in locale tramite PC (per il collegamento al PC è necessario il modulo opzionale d'interfaccia USB cod. FT782M • € 13,00).

NUOVI ALIMENTATORI SWITCHING INDUSTRIALI

VASTA GAMMA DI MODELLI PER TUTTE LE ESIGENZE

BEST OF BEST by FUTURA ELETTRONICA BEST OF BEST



MODULI DI ALIMENTAZIONE
da 5-12-24-36-48V con potenza da 50 a 350W

Prezzi IVA inclusa.

DISPONIBILI CON LE SEGUENTI CARATTERISTICHE:

CODICE	TENSIONE D'INGRESSO	TENSIONE D'USCITA	CORRENTE D'USCITA	POTENZA	RIPPLE	DIMENSIONI	PREZZO
MW05005	170 - 264 VAC	5 VDC	10 A	50 W	75 mV	159 X 97 X 38 mm	15,00 €
MW05012	170 - 264 VAC	12 VDC	4,2 A	50 W	100 mV	159 X 97 X 38 mm	15,00 €
MW05024	170 - 264 VAC	24 VDC	2,1 A	50 W	100 mV	159 X 97 X 38 mm	15,00 €
MW10005	170 - 264 VAC	5 VDC	20 A	100 W	100 mV	199 X 98 X 38 mm	20,00 €
MW10012	170 - 264 VAC	12 VDC	8,5 A	100 W	100 mV	199 X 98 X 38 mm	20,00 €
MW10024	170 - 264 VAC	24 VDC	4,5 A	100 W	100 mV	199 X 98 X 38 mm	20,00 €
MW14505	176 - 264 VAC	5 VDC	25 A (30 A max.)	125 W	100 mV	199 X 98 X 38 mm	24,00 €
MW14512	88- 132 VAC / 176- 264 VAC (selez.)	12 VDC	12 A	144 W	120 mV	199 X 98 X 38 mm	24,00 €
MW14524	88- 132 VAC / 176- 264 VAC (selez.)	24 VDC	6 A	144 W	150 mV	199 X 98 X 38 mm	24,00 €
MW35005	90-132 VAC /180-264 VAC (selez.)	5 VDC	50 A	250 W	150 mV	215 X 115 X 50 mm	39,00 €
MW35012	90-132 VAC /180-264 VAC (selez.)	12 VDC	29 A	350 W	150 mV	215 X 115 X 50 mm	39,00 €
MW35024	90-132 VAC /180-264 VAC (selez.)	24 VDC	14,6 A	350 W	150 mV	215 X 115 X 50 mm	39,00 €
MW35036	90-132 VAC /180-264 VAC (selez.)	36 VDC	9,7 A	350 W	150 mV	215 X 115 X 50 mm	39,00 €
MW35048	90-132 VAC /180-264 VAC (selez.)	48 VDC	7,3 A	350 W	240 mV	215 X 115 X 50 mm	39,00 €

FUTURA ELETTRONICA®

www.futurashop.it

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche di questi prodotti e acquisti on-line su *www.futurashop.it*

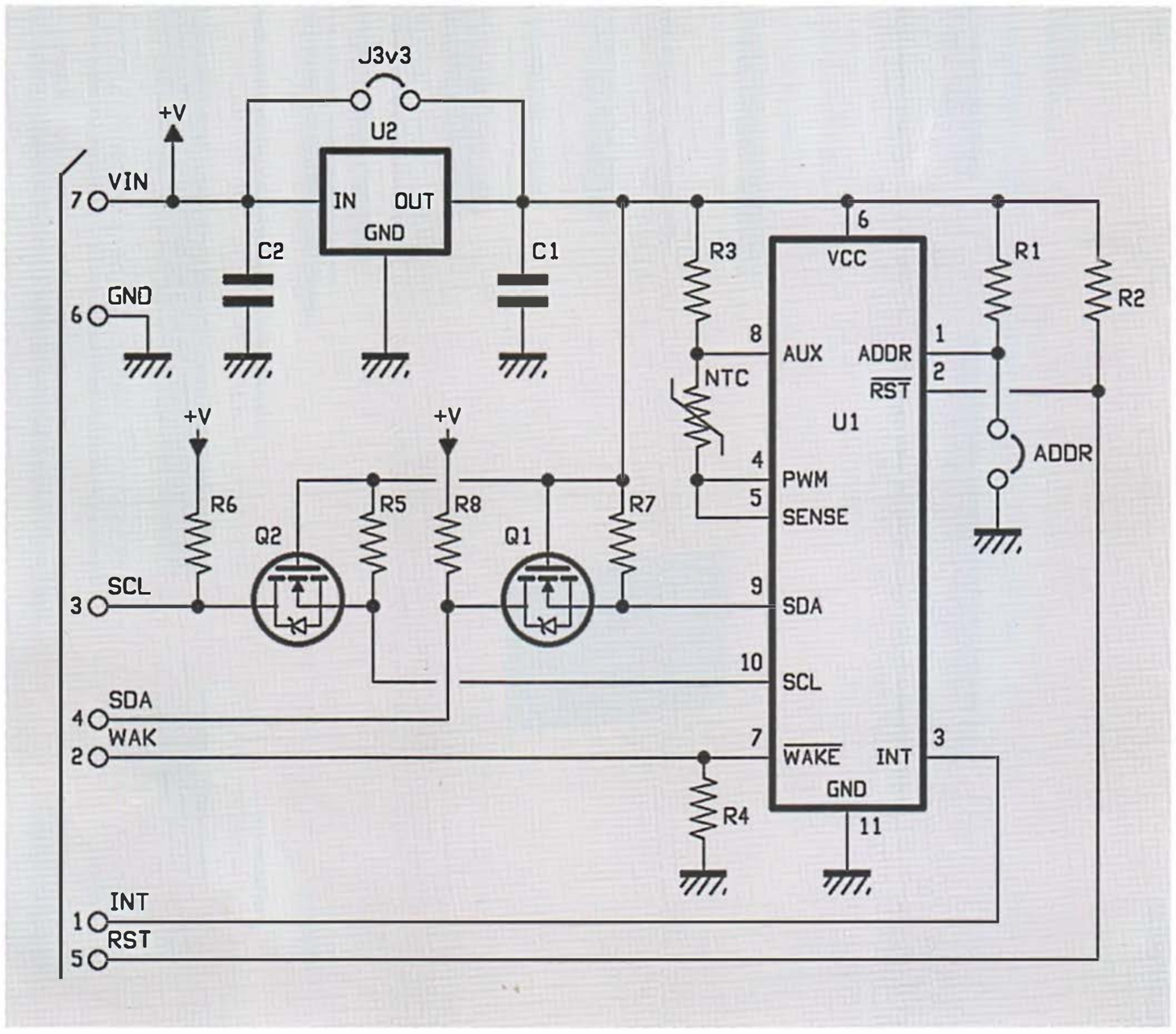


Grazie a un sensore capace di rilevare la concentrazione di particelle inquinanti in sospensione, realizziamo uno strumento in grado di comunicarci quando l'aria è viziata ed è opportuno aerare il locale.

RILEVIAMO LA QUALITA' DELL'ARIA

di ALESSANDRO SOTTOCORNOLA

Quello dell'inquinamento atmosferico è un problema ormai attuale da decenni, in special modo se lo collochiamo nelle città, piccole o grandi che siano e in prossimità dei grandi insediamenti industriali; ne consegue una preoccupazione, a volte anche poco motivata, per la qualità dell'aria che respiriamo. Esistono da tempo centraline per il monitoraggio degli agenti inquinanti, in special modo di quelli che gli studi condotti periodicamente portano alla ribalta della cronaca, ma che sono essenzialmente il monossido di carbonio, l'anidride carbonica e le famigerate polveri sottili; i dati che esse rilevano vengono poi elaborati e sovente resi pubblici sui siti meteo e su quelli degli organi preposti alla vigilanza



sulla salute pubblica, oltre che su quelli di alcuni grandi comuni. Malgrado questa forma di informazione, un po' come avviene con le centraline meteo, che tanti vorrebbero in casa per avere le proprie previsioni personalizzate, anche per la qualità dell'aria c'è forte interesse verso dispositivi domestici da utilizzare per conoscere la situazione all'aperto ed anche nei luoghi chiusi, dove la questione diventa rilevante nella stagione fredda, quando per non disperdere calore si aprono poco porte e finestre; in quest'ultimo caso, più che gli inquinanti interessa la concentrazione di ossigeno rispetto all'anidride carbonica, ma non vanno trascurati gli inquinanti, perché escluso per legge il fumo di sigaretta, nei locali di lavoro possono però proliferare solventi, polveri ed anche l'ozono prodotto dalle fotocopiatrici e il toner sfuggito ad esse.
Trovandoci tra le mani un sensore di qualità dell'aria e considerato l'interesse per l'argomento, ci siamo subito messi al lavoro ed abbiamo realizzato una breakout board per mille esperimenti ed anche per l'utilizzo come elemento rilevatore ambientale di sistemi in grado di monitorare la qualità dell'aria, basati anche solo su un piccolo microcontrollore ad 8 bit come quello di Arduino Uno.
Nello specifico, il nostro progetto permette di conoscere la concentrazione di anidride carbonica (così da indicarci quanto l'aria in un ambiente chiuso è viziata) e il TVOC (Total Volatile Organic Compounds) ossia la concentrazione in parti per miliardo di particelle in sospensione.

SCHEMA ELETTRICO

Il circuito è stato realizzato sotto forma di breakout board per rendere facilmente accessibili le connessioni dell'integrato, che come molti componenti ormai è incapsulato in un case per montaggio superficiale: in questo caso l'integrato è il sensore CCS811 della AMS (*http://ams.com*) ed è reso disponibile in un piccolissimo case LGA (con in contatti sotto) da appena 2,7 x 4 mm.
Questo componente integra un sensore ad ossido metallico che durante il funzionamento può essere penetrato da determinati ioni presenti nell'aria e in base alla "contaminazione" varia i propri parametri, che vengono letti dalla logica interna e convertiti in dati resi disponibili attraverso una connessione a bus I²C. Rispetto ai canonici sensori di gas e impurità che conosciamo (per esempio gli storici Figaro...) che incorporano un elemento a biossido di stagno riscaldato da una resistenza che assorbe una discreta corrente, questo consuma pochissimo: tipicamente 1,2 milliwatt, alimentato con circa 3 volt. Diventa quindi l'ideale per molti dispositivi e sensori da alimentare con soluzioni di energy harvesting o a pile.
Il basso consumo deriva dalle ridottissime dimensioni dell'elemento sensibile, che richiedono quindi un riscaldatore piccolo e di potenza minima.
Diamo uno sguardo allo schema elettrico per vedere com'è stato utilizzato: il nostro sensore (U1) è interfacciato con il modo esterno tramite un bus I²C dotato di traslatori di livello, giacché funziona con una tensione di 3,3 volt, assicurati dal regolatore LDO TS3480CX33 della Taiwansemi (*www.taiwansemi.com*) che nello schema è siglato U2 e riceve la tensione d'ingresso (filtrata dai disturbi mediante il condensatore C2) dal contatto VIN della breakout board, riferita alla massa comune (GND).
Il regolatore si rende indispensabile se la tensione disponibile supera i 3,3 volt o non è stabilizzata, mentre laddove siano disponibili 3,3V stabilizzati è possibile bypas-

sarlo chiudendo il ponticello J3v3. L'alimentazione a 3,3V viene ripulita dal condensatore di filtro C1 e giunge all'integrato sensore sul piedino 6.
Il CCS811 rileva la qualità dell'aria attraverso il rilevamento della concentrazione di taluni gas e particelle in sospensione che si introducono dalle finestrelle presenti sulla parte superiore del corpo dell'integrato; la concentrazione dei gas altera la conducibilità dell'elemento sensibile a ossido metallico, collegato tra il piedino di massa e l'ingresso della MCU interna, il quale viene riscaldato da un piccolo elemento riscaldatore alimentato con un segnale PWM prodotto dalla MCU interna fornito da un transistor che ha il riscaldatore stesso come carico di drain (il positivo del riscaldatore è alimentato dal piedino 6 dell'integrato) per essere più sensibile.
La temperatura dell'elemento ad ossido metallico è stabilizzata grazie alla logica che fa capo al piedino SENSE (5).
La variazione di resistenza determina una modifica del potenziale presente sul S+ rispetto a massa (**Fig. 1**) che viene letta dalla MCU interna e interpretata come concentrazione di inquinanti.
Lo schema interno proposto dalla **Fig. 1** aiuta a comprendere come funziona l'integrato della AMS.
Per funzionare correttamente e non fornire indicazioni errate deve auto-calibrarsi, ovvero deve trascorrere un periodo relativamente lungo (quasi un giorno intero) nell'ambiente di cui deve rilevare la qualità dell'aria, allo scopo di accumulare valori per costruire una media di riferimento.
L'integrato prevede la possibilità di correggere le letture sulla base della temperatura ambiente, cosa utile perché la penetrazione degli inquinanti nel sensore a ossido metallico cambia dal caldo al freddo e con essa la concentrazione degli stessi rilevata dal CCS811; il rilevamento della temperatura è, nel nostro circuito, affidato a un termistore NTC collegato tra i piedini 8 (AUX) e SENSE. Notate che l'NTC non è obbligatorio, ma si è deciso di inserirlo per rendere maggiormente veritieri i dati forniti.
Per completezza va detto che il CCS811 può anche effettuare una correzione dei valori sulla base dell'umidità dell'aria ed allo scopo richiede la connessione di un sensore di umidità in parallelo o in serie (a seconda della configurazione scelta) al termistore. Meglio ancora sarebbe utilizzare l'apposito sensore combinato di temperatura e umidità ENS210, sempre della AMS.
Il termistore si connette tra il piedino AUX (8) e il SENSE; l'adozione di un NTC permette di alzare la corrente nell'AUX al crescere della temperatura. Il resistore R3, collegato tra l'AUX e il positivo, fa partitore con il termistore ed è calcolato di pari valore a 25 °C di temperatura ambiente.
Nel nostro caso, avendo omesso l'igrometro, il parametro di umidità dev'essere forzato; nell'applicazione che vi proporremo in abbinamento a una scheda Arduino, l'umidità è stata imposta fissa al 50% in modo da far compiere in ogni caso il corretto calcolo; tale forzatura è stata effettuata da software attraverso il bus di comunicazione.
Per rendere la breakout board compatibile con le varie board Arduino e in generale con tutti i circuiti a microprocessore funzionanti con livelli logici TTL, abbiamo previsto nel circuito un adattatore di livelli in modo da poter lavorare con dispositivi con logica sia a 5V che 3,3V. In pratica il traslatore permette di fornire all'uscita livelli alti a 5 volt e adattare a 3,3V gli uno logico entranti. Questo adattatore è stato applicato sia a SCL che a SDA del bus I²C.
Invece le linee INT ed RST, pur portate all'esterno del componente tramite i contatti 1 (INT) e 5 (RST) della breakout board non hanno alcun adattatore e devono quindi essere gestite con dispositivi operanti con livelli logici compatibili con l'alimentazione del CSS811.
Soffermiamoci un istante sul

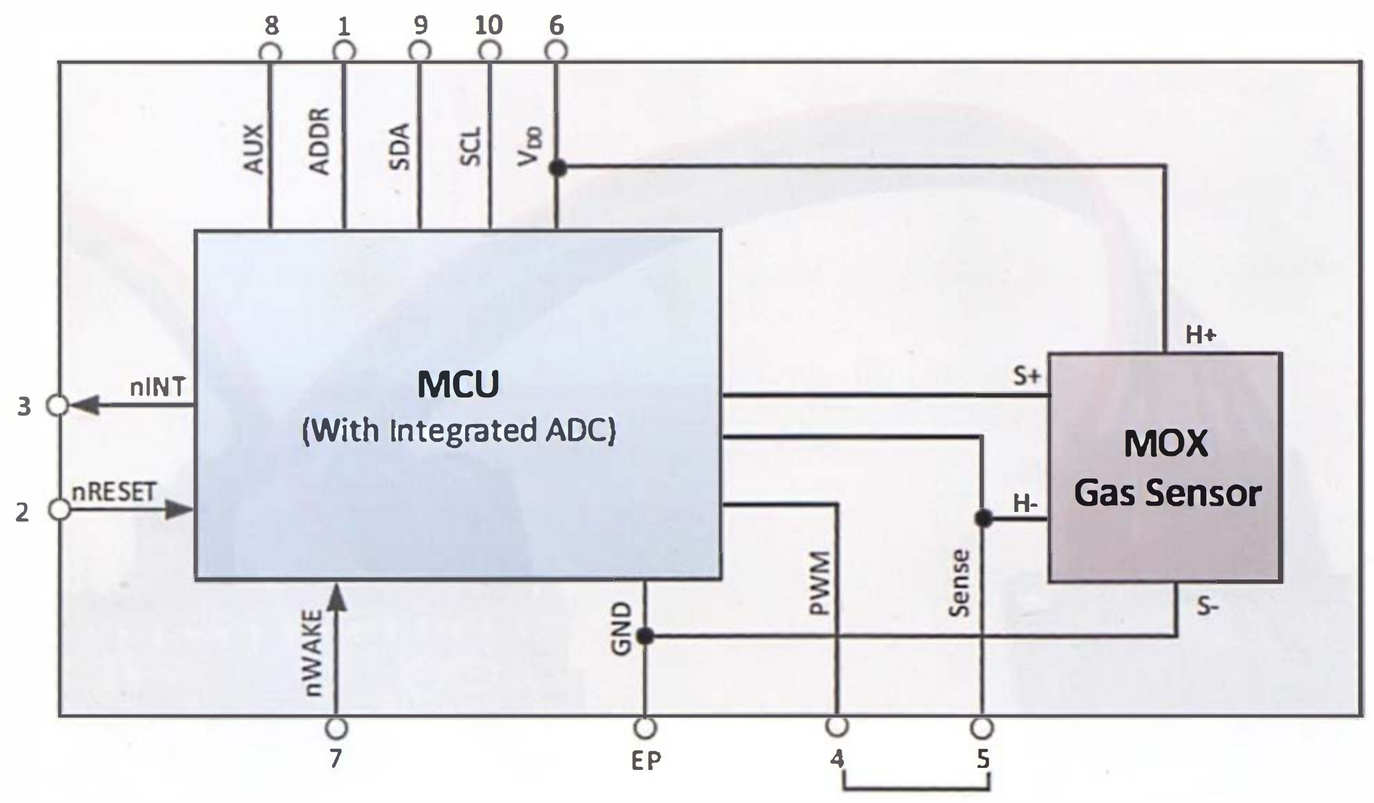
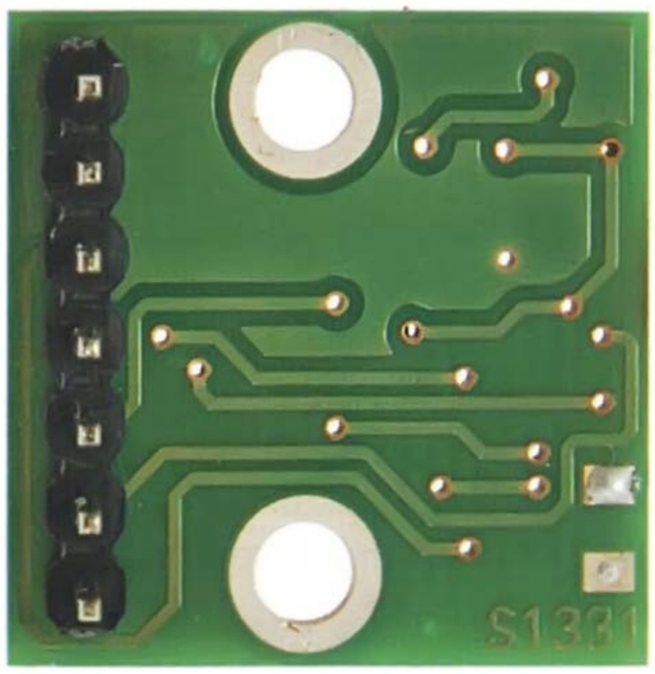
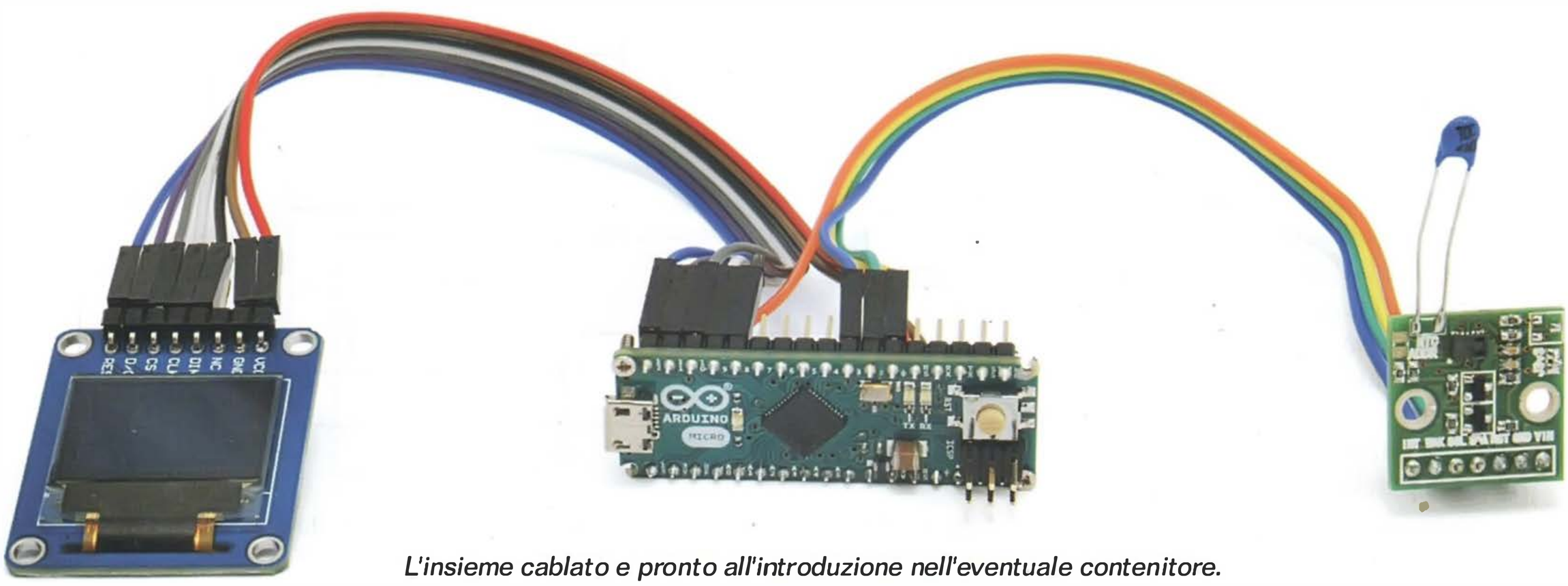


***Fig. 1** - Schema a blocchi dell'integrato CCS811: affinché il riscaldatore dell'elemento sensibile funzioni occorre unire i piedini 4 e 5.*



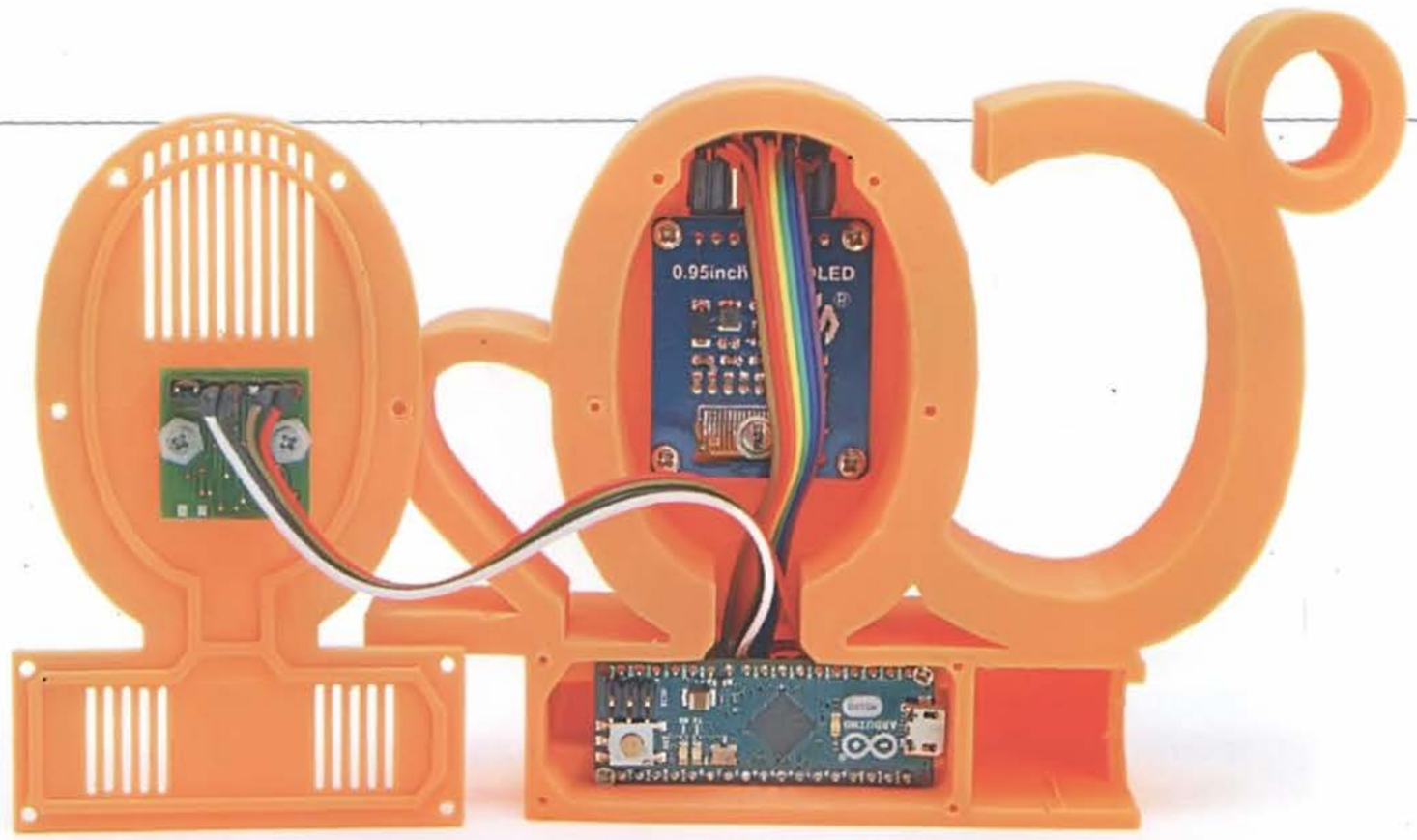
Sul lato inferiore, un pin-strip agevola il montaggio della breakout board.

traslatore di livelli logici posto sull'I²C-Bus per spiegare come funziona; i due MOSFET sono stati inseriti come traslatori di livello e la loro configurazione, sebbene appaia strana, riesce ad adattare i livelli logici TTL (0/5V) a quelli dell'integrato U1 e viceversa.
Affinché lo stadio funzioni, i MOSFET (che sono degli enhancement-mode a canale N) hanno il gate polarizzato con i 3,3V della linea di alimentazione positiva del CCS811 e il drain collegato all'alimentazione VIN mediante un resistore di pull-up; Q1 ha quindi il drain collegato al contatto SDA (piedino 4) del connettore della breakout board, mentre il drain del Q2 termina sul contatto SCL (3) dello stesso connettore. Notate che siccome l'U1 è un dispositivo slave dell'I²C-Bus, SCL è una linea di input dello stesso e della breakout board, perciò è unidirezionale, mentre SDA deve funzionare in maniera bidirezionale. SCL e SDA dell'U1 hanno ciascuna un resistore di pull-up collegato al solito positivo d'uscita del regolatore U2 e terminano ciascuna sul source del rispettivo MOSFET.
Per comprendere il funzionamento della sezione analizziamo quello della parte relativa alla linea SCL, che è un segnale entrante: quando l'uscita SCL di Arduino (o del microcontrollore che governa la breakout board) è a livello logico basso, il MOSFET Q2 si trova interdetto in quanto il gate è equipotenziale rispetto al source, tuttavia il suo diodo di protezione stavolta è polarizzato direttamente e porta a 0 logico (0,6V) la linea SCL (piedino 10) dell'U1. Invece quando la linea corrispondente di Arduino fornisce l'1 logico, il diodo si interdice ed anche il MOSFET è interdetto, ma il resistore R5 assicura i 3,3 volt sulla linea / SCLK. Quindi con SCL del connettore posto a 0 volt, SCL si trova a zero logico, mentre con lo stesso a 5 V è a 1 logico, cioè a 3,3 volt.
Alla luce di questo modo di funzionamento verrebbe da obiettare che per ottenerlo basterebbero solo il diodo e le due resistenze di pull-up; ma il MOSFET serve in realtà per garantire la comunicazione bidirezionale, anche se in effetti in questo circuito ci occorre solo per la linea SDA. Vediamo quindi come funziona la cosa nel caso di comunicazione sull'SDA: quando è l'integrato U1 a dover inviare dati, se si porta a zero logico manda in conduzione il Q1, giacché quest'ultimo ha il gate polarizzato con 3,3V; il drain si porta circa allo stesso potenziale e trascina a zero logico SDA del connettore.
Invece quando il piedino 9 dell'U1 è a livello alto, il MOSFET Q1 è interdetto e il resistore R8 mantiene SDA a 5 volt.
Analizziamo adesso cosa succede quando a inviare il segnale è Arduino o altro dispositivo collegato al connettore: portando SDA a zero logico, come visto per la linea SCL il diodo di protezione del MOSFET (Q1, in questo caso...) trascina a circa 0,6V il piedino 9 dell'U1, portandolo di fatto a livello basso, mentre se viene portato a 5V, il diodo di protezione e il MOSFET sono interdetti, cosicché è il resistore di pull-up R7 a mantenere a 3,3V il predetto piedino 9 dell'integrato.
Per concludere la descrizione



L'insieme cablato e pronto all'introduzione nell'eventuale contenitore.

dello schema elettrico occupiamoci del jumper ADDR e delle linee RST e WAK; ADDR consente di sceglie tra due valori l'indirizzo I²C-Bus assegnato al CCS811, agendo sul piedino di indirizzo (1); il ponticello imposta l'ultimo bit dei tre che definiscono l'address I²-C, mentre i primi due bit sono impostati fissi all'interno dell'integrato. Quando ADDR è a livello basso, l'address vale 90 decimale (0x5A esadecimale) mentre se è 1 l'indirizzo vale 91 decimale o 0x5B hex.
RST è l'ingresso di reset e, posto a zero logico, resetta l'integrato U1; normalmente è disattivato perché il resistore di pull-up R2 mantiene /RST (piedino 2) a livello alto.
Infine, WAK è il pin di wake-up: normalmente è posto a massa dalla resistenza di pull-down R4 ma se viene portato a 1 logico blocca l'interfaccia I²C-Bus. Viene gestito da Arduino o dall'eventuale microcontrollore cui si affida il governo della breakout board.



I componenti inseriti nel contenitore che abbiamo stampato in 3D.

LA GESTIONE SOFTWARE

Per mostrarvi le potenzialità della breakout board e del sensore che ne è alla base, abbiamo deciso di integrarla in un progetto che si propone di mostrare su un piccolo display OLED i valori rilevati, con uno sfondo il cui colore cambia in base a come viene valutata la situazione, dal verde (aria buona) al rosso (aria di pessima qualità) passando per il giallo (aria accettabile ma non ideale).
Come mostra lo schema di cablaggio visibile in queste pagine, per l'acquisizione dei dati dalla breakout sensore e la loro visualizzazione su display previa elaborazione, utilizziamo una scheda Arduino Micro, che è una delle più semplici e piccole Arduino.

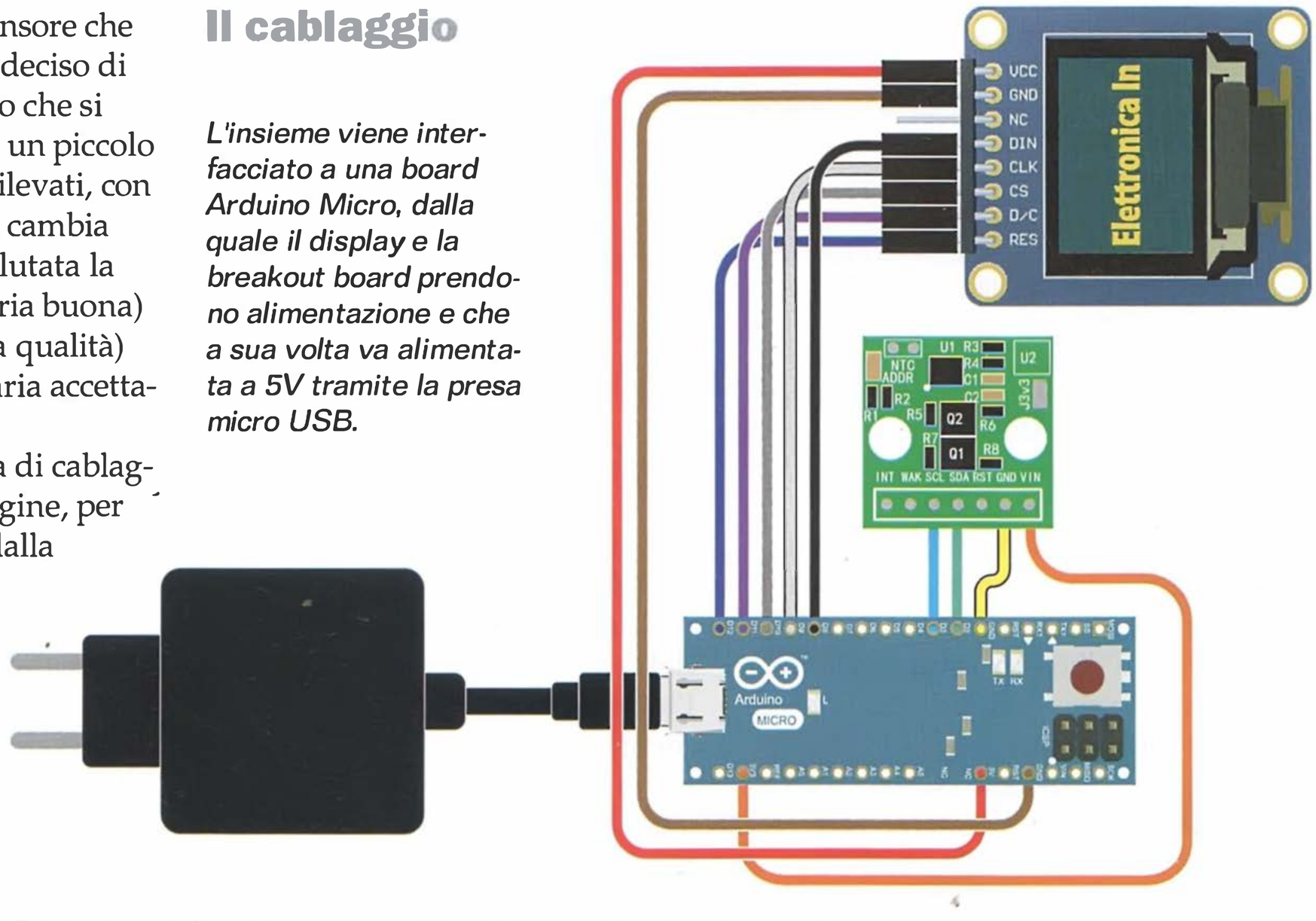
Il display OLED è da 0,95" ed è commercializzato dalla Futura Elettronica (*www.futurashop.it*) con il codice 2850-OLEDRGBA ed ha interfaccia SPI; si basa su controller grafico SSD1331 della SOLOMON SYSTECH ed ha una risoluzione di 96x64.
Nel nostro esempio applicativo, per la gestione del sensore CCS811 allo scopo di acquisire le informazioni su inquinamento e temperatura dell'aria è stata sfruttata la libreria fornita dalla Sparkfun (*www.sparkfun.com/products/14193*) per il proprio sensore, dove abbiamo modificato la funzione di lettura NTC (readNTC).
Per il calcolo del valore di temperatura ci siamo basati sulla documentazione ufficiale fornita dalla AMS e sulla formula riportata nel rispettivo datasheet, che è la seguente:

RNTC = VNTC x RREF / VREF

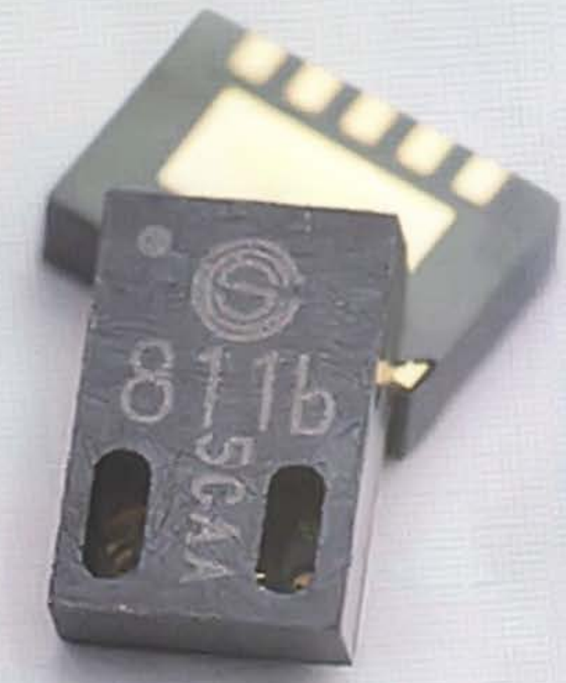
Questa formula correla la resistenza assunta dal termistore con la tensione ai suoi capi e con il

Il cablaggio

L'insieme viene interfacciato a una board Arduino Micro, dalla quale il display e la breakout board prendono alimentazione e che a sua volta va alimentata a 5V tramite la presa micro USB.



Il sensore



Per determinare la qualità dell'aria circostante utilizziamo un sensore siglato CCS811 e basato su un film sensibile di ossido metallico riscaldato da un microriscaldatore a ridottissimo consumo di energia, tanto che il componente potrebbe essere integrato in uno smart watch. Dotato di un microcontrollore che ne governa le funzioni e che trasforma il segnale analogico ottenuto dall'elemento sensibile in dati digitali disponibili tramite un bus I²C, il nostro componente è in grado di determinare la concentrazione di un ampio campo di Volatile Organic Compounds (VOC) dai quali desume la concentrazione di anidride carbonica.
Il chip è incapsulato in contenitore LGA ad 11 contatti, più quello centrale, che fa da massa e da dissipatore di calore; l'alimentazione richiesta è a tensione continua di valore compreso fra 1,8 e 3,6 V e l'assorbimento molto modesto: al massimo 30 mA. Le figure in questo riquadro ne mostrano l'aspetto e il tipico schema applicativo: appare evidente la doppia finestrella attraverso quale passa l'aria che raggiunge lo strato sensibile di ossido metallico.
Il componente rileva la qualità dell'aria attraverso il rilevamento della concentrazione di taluni gas e particelle in sospensione che si introducono dalle finestrelle; la concentrazione dei gas altera la conducibilità dell'elemento sensibile a ossido metallico, collegato tra il piedino di massa e l'ingresso della MCU interna, il quale viene riscaldato da un piccolo elemento riscaldatore alimentato con un segnale PWM prodotto dalla MCU interna (il positivo del riscaldatore è alimentato dal piedino 6 dell'integrato) per essere più sensibile. La temperatura dell'elemento ad ossido metallico è stabilizzata grazie alla logica che fa capo al piedino SENSE (5). La variazione di resistenza determina una modifica del potenziale presente sul +S rispetto a massa che viene letta dalla MCU interna e interpretata come concentrazione di inquinanti. La logica di controllo dell'integrato prevede alcuni pin di gestione, che sono:

- INT (piedino 3) è un'uscita attiva a livello alto che viene portata a zero logico dalla MCU per indicare il compimento di un ciclo di misura della qualità dell'aria o che un valore di soglia preimpostato è stato raggiunto;
- RST (piedino 2) attivo a livello basso, va normalmente mantenuto a Vdd mediante un resistore di pull-up, tipicamente da 4,7 kΩ;
- WAK (piedino 7) è il wake, ossia a livello zero risveglia la comunicazione: il microcontrollore lo deve porre a zero prima di iniziare qualsiasi sessione di comunicazione su I²C-Bus e deve mantenerlo in tale stato finché la sessione non viene terminata.

Per non fornire indicazioni errate l'integrato deve autocalibrarsi, ovvero deve trascorrere un periodo relativamente lungo (quasi un giorno interno) nell'ambiente di cui deve rilevare la qualità dell'aria, allo scopo di accumulare valori per costruire una media di riferimento.
L'integrato prevede la possibilità di correggere le letture sulla base della temperatura, il cui rilevamento è affidato a un termistore NTC collegato tra i piedini 8 (AUX) e SENSE. Il CCS811 può anche effettuare una correzione dei valori sulla base dell'umidità dell'aria ed allo scopo richiede la connessione di un sensore di umidità in parallelo o in serie (a seconda della configurazione scelta) al termistore. Il termistore o l'igrometro si connettono tra il piedino AUX (8) e il SENSE.

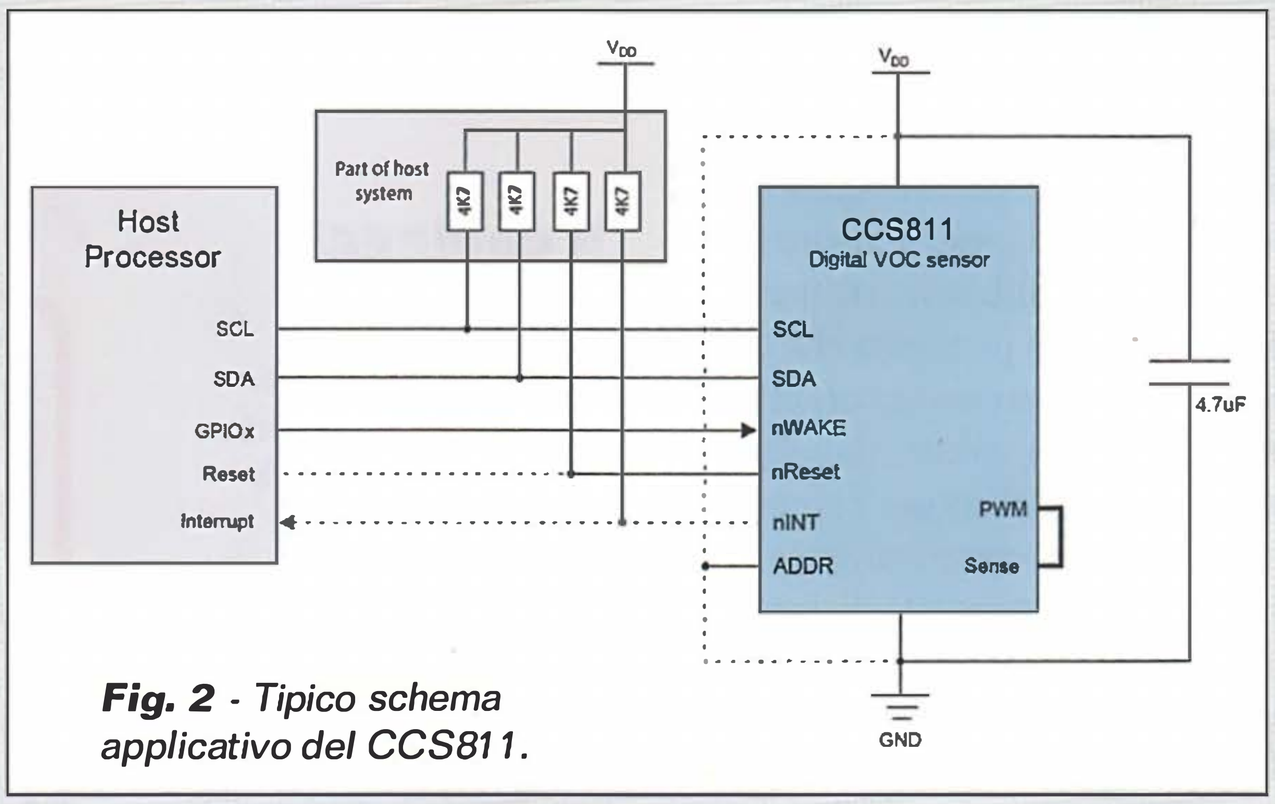


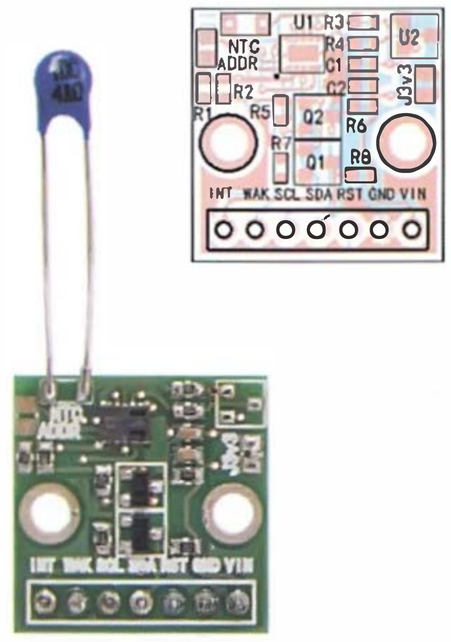
***Fig. 2** - Tipico schema applicativo del CCS811.*

valore del resistore che vi fa partitore (R3, nello schema elettrico...) e la Vref.
Il valore di temperatura rilevato viene espresso in forma numerica e scritto nel registro a 4 bit NTC Register (indirizzo 0x06).
Nel nostro esempio, abbiamo deciso di rendere disponibili i dati direttamente sul display. In fase di avvio dello sketch per Arduino, viene testata la presenza del sensore, mostrando a display l'indicazione di sensore correttamente rilevato se il dialogo va a buon fine, ovvero errore in caso contrario; dopo verranno mostrati i dati provenienti dal modulo.
Tenete presente che per una corretta indicazione, il sensore si deve auto-calibrare, pertanto è importante rispettare il tempo fornito dalla casa madre per avere un valore veritiero.
Il tempo minimo di attesa per avere un dato attendibile è di 24 ore. Per rendere più intuitiva la

Elenco Componenti:

C1, C2: 100 nF ceramico (0603)
R1, R3, R4: 100 kohm (0603)
R2: 4,7 kohm (0603)
R5÷R8: 2,2 kohm (0603)
U1: CCS811
U2: TS3480CX33 RFG
Q1, Q2: BSS138W-7-F
NTC: NTC 100 kohm

Varie.
- Strip maschio 6 vie 90°
- Circuito stampato S1331 (21x21 mm)



visualizzazione da parte del display, i valori mostrati hanno una colorazione differente in base alla qualità dell'aria cui corrispondono; la **Tabella 1** mostra i colori adottati per la visualizzazione della concentrazione di anidride carbonica. Invece nella **Tabella 2** vedete a che colori corrisponde lo sfondo quando il display visualizza la concentrazione totale delle particelle organiche in sospensione nell'aria. Infine nella **Tabella 3** sono riepilogati i colori corrispondenti alla visualizzazione della temperatura ambiente desunta dallo stesso termistore NTC che utilizziamo per effettuare la compensazione in temperatura delle concentrazioni rilevate di TVOC e CO_2.

REALIZZAZIONE PRATICA

Passiamo adesso alla costruzione del nostro strumento, per il quale abbiamo anche previsto un simpatico contenitore da tavolo (pensando all'utilizzo all'interno) stampato in PLA con la nostra stampante 3Drag.
Ma andiamo con ordine: di tutta l'elettronica coinvolta, l'unica scheda da autocostruire è la breakout board, perché Arduino Micro e il display con il suo PCB si comperano già pronti; per la breakout board dovrete preparare il circuito stampato per fotoincisione, utilizzando come pellicole le stampe delle tracce lato rame scaricabili dal nostro sito *www.elettronicain.it* insieme agli altri file del progetto. Una volta inciso e forato lo stampato potete iniziare il montaggio.
Il circuito fa largo uso di componentistica SMD (non sono per montaggio superficiale solo il termistore e i jumper), ragion per cui bisogna avere un po' di manualità e l'attrezzatura di base, consistente in un saldatore da non più di 20 W con punta molto fine, del filo di lega saldante non più grande di 0,5 mm di diametro, della trecciola dissaldante e della pasta flussante, una pinzetta (tipo quelle da garza) e una lente d'ingrandimento per verificare le saldature e aiutarvi nel posizionamento degli elementi più piccoli.
Per il montaggio posizionate i componenti, con l'aiuto della pinzetta, nelle loro piazzole ben centrati, quindi saldateli dopo aver deposto della pasta flussante sulle rispettive piazzole; la pasta agevola lo scioglimento della lega saldante e il suo posizionamento sulle piazzole e sui contatti dei componenti. Iniziate con gli integrati, che vanno centrati nelle rispettive piazzole e poi saldati partendo da un terminale su uno degli angoli, quindi procedendo

CO_2	Colore	Qualità aria
0 - 600	Verde	Excellent Air
601 - 800	Verde	Good Air
801 - 1000	Verde	Fair Air
1001 - 1500	Giallo	Mediocre Air
1501 - 2100	Rosso	Bad Air

Tabella 1

TVOC	Colore
0 - 50	Verde
51 - 100	Giallo
101 - 150	Giallo
151 - 200	Rosso
201 - 300	Rosso
301 - 500	Rosso

Tabella 2

°C	Colore
Inferiore a 15	Blu
15 - 30	Giallo
Maggiore di 30	Rosso

Tabella 3



Il colore dei parametri cambia a seconda che siano buoni (verde), accettabili (giallo) o pericolosi (rosso).

L'inquinamento dell'aria

L'aria che ci circonda non è fatta solo di quell'ossigeno che ci permette di vivere, ma in buona parte di altri gas (primo fra tutti l'azoto); eppure quando la concentrazione di ossigeno scende sensibilmente inizia una serie di problemi, primo fra tutti il bisogno di accelerare la massa d'aria introdotta nei polmoni e il battito cardiaco per far giungere l'ossigeno alle cellule. La carenza di ossigeno diviene più rilevante col caldo, giacché a parità di volume d'aria inspirata il contenuto di ossigeno è minore, a causa della dilatazione termica che tutti i gas subiscono. Poco ossigeno nell'aria, condizione che si verifica frequentemente quando si soggiorna in ambienti chiusi e poco aerati, è causa di difficoltà di concentrazione, stanchezza ecc.
Ugualmente importante è la concentrazione di inquinanti in sospensione, tra i quali ricordiamo il monossido di carbonio (CO, velenoso perché una volta inspirato associa un atomo di ossigeno impedendone la cessione alle cellule) frutto della combustione e le polveri sottili causate dall'incompleta combustione degli idrocarburi (il famigerato particolato fine o PM). Di tali micropolveri, sebbene se ne parli a sproposito, non sono responsabili solo gli autoveicoli a motore endotermico (che contribuiscono per una percentuale molto ridotta) ma anche gli scarichi delle centrali termoelettriche e delle aziende, oltre che quelli degli impianti di riscaldamento; sicuramente il particolato derivante dalla combustione nei motori è formato da idrocarburi incombusti (HC).
Le polveri sottili si dividono per perigolosità: quelle grossolane (PM30 e oltre) causano nella maggior parte dei casi irritazione delle vie aeree superiori ma possono essere espulse insieme al muco, mentre quelle finissime (PM10 e inferiori) si è dimostrato che attraversano le membrane degli alveoli polmonari ed entrano nel sangue, comportando seri rischi per la salute.
Conoscere la percentuale di inquinanti è utile a valutare i rischi per la salute; valutare la concentrazione di gas come la CO_2, non pericolosa ma che a parità di volume d'aria "ruba" posto all'ossigeno, ci aiuta a capire quando è il caso di cambiare aria negli ambienti chiusi. Per quanto riguarda le particelle organiche in sospensione, può essere interessante guardare la **Tabella 4**, che consente di valutare la qualità dell'aria e la pericolosità in base alla concentrazione, espressa in parti per miliardo.

Valori VOC	Qualità aria	Precauzioni
0÷50	buona	esposizione senza rischi
51÷100	accettabile	limitare l'esposizione prolungata di soggetti molto sensibili
101÷150	non salutare per soggetti sensibili	limitare l'esposizione prolungata di soggetti sofferenti di patologie respiratorie e bambini molto piccoli
151÷200	non salutare	evitare l'esposizione anche breve di soggetti sofferenti di patologie respiratorie; limitare l'esposizione dei bambini piccoli
201÷300	limite di pericolosità	evitare l'esposizione anche breve di soggetti sofferenti di patologie respiratorie e l'esposizione dei bambini piccoli
301÷500	pericolosa	evitare tutti l'esposizione

Tabella 4

un terminale di un lato per volta; sistemate poi condensatori, resistenze e i MOSFET. Il termistore NTC va montato anch'esso come fosse SMD, ma in piedi; meglio sarebbe farlo toccare con il PCB o il sensore AM. Per le connessioni con l'esterno dovete infilare nei rispettivi fori dei pin-strip a passo 2,54 mm e saldarli; due pin-strip da 2 poli ciascuno serviranno per i jumper previsti nella breakout board.
Fatte le saldature verificate che non vi siano cortocircuiti e baffi di stagno, quindi la vostra basetta è pronta per essere collegata come mostra l'apposito disegno.
A questo punto dovete caricare in Arduino Micro lo sketch del caso: dopo aver collegato con un cavo USB-A/micro USB la board Arduino ed il computer, avviate quest'ultimo e una volta aperto l'IDE Arduino, selezionate il dispositivo target (Arduino Micro) quindi aprite lo sketch (disponibile per il download sul nostro sito insieme agli altri file del progetto) ed eseguite il caricamento.
Completata l'operazione potete subito testare il sistema, alimentando Arduino Micro con un alimentatore da rete dotato di cavetto terminante con uno spinotto microUSB, che eroghi 5V stabilizzati e una corrente di circa 500 milliampere. ■



per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono di facile reperibilità, mentre quelli elencati di seguito si possono acquistare presso Futura Elettronica. La breakout CCS811 air quality (cod. FT1331M) è in vendita montata e collaudata al prezzo di Euro 32,00, il termistore NTC 100 K (cod. NTC100K0) è disponibile separatamente a Euro 0,70. La board Arduino Micro con ATMEGA32U4 (cod. ARDUINOMICRO) costa Euro 25,00, il display OLED a colori (cod. OLEDRGBA) si può acquistare a Euro 13,60, mentre l'alimentatore micro USB 5V/3A (cod. AL05-3USB) è diponibile a Euro 10,50. Per chi desidera stampare il contenitore utilizzato nel progetto, i file sono scaricabili dal sito della rivista. In alternativa è possibile sfruttare il servizio stampa 3D di Futura Elettronica ordinandolo con il codice CASEAIRQUALITY al prezzo di Euro 36,00. Tutti i prezzi si intendono IVA compresa.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775
http://www.futurashop.it

trackimo

Con lui nulla è perduto!

LOCALIZZATORE SATELLITARE GPS/GSM CONTROLLABILE DA SMARTPHONE E PC



SIM CON 12 MESI DI SERVIZIO INCLUSI.
Rinnovo annuale € 49,00 oppure mensile € 4,90.



€ 119,00
Cod. **TRACKIMO**

Vuoi sapere dove si trovano i tuoi cari, i tuoi animali domestici, oppure vuoi localizzare i bagagli, la moto, il drone, l'auto, la barca, la bicicletta? Niente di più semplice!
Grazie a Trackimo, uno tra i più piccoli (47x40x17 mm) e leggeri (42 grammi) localizzatori GPS/GSM al mondo potrai monitorare tramite smartphone (iOS e Android) o PC, tutto ciò che desideri in tempo reale. Trackimo dispone di funzione SOS, batteria ricaricabile agli ioni di litio in grado di garantire un'autonomia in standby di oltre 144 ore (48-72 ore in funzione) e viene fornito con una SIM da attivare gratuitamente con copertura MONDIALE per 12 mesi senza costi extra con allerte illimitate tramite APP ed e-mail.













FUTURA ELETTRONICA®
Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Prezzo IVA Inclusa.

Caratteristiche tecniche e vendita on-line su:

www.futurashop.it

Vuoi entrare nell'affascinante mondo di FISHINO e imparare a utilizzarlo al meglio?

Allora non perderti questa FANTASTICA OFFERTA!

B&B
Board & Book

Acquista insieme libro & scheda: avrai uno sconto di 3 euro!

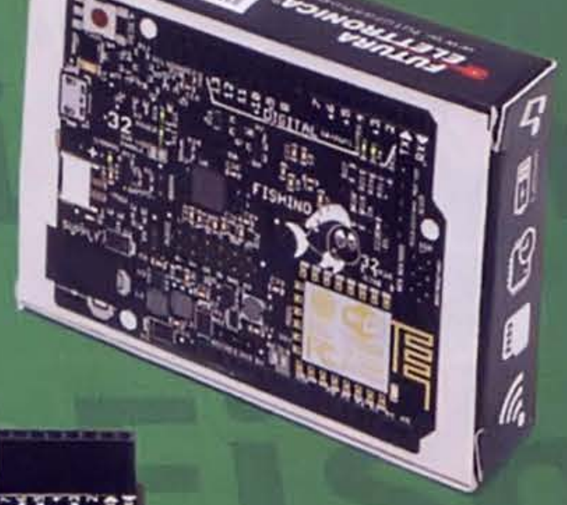


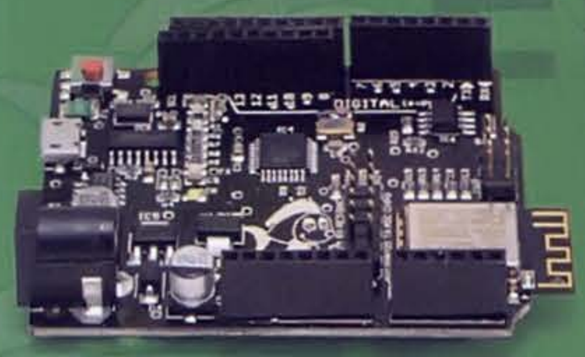
IVA INCLUSA

€ 48,00

anziché

~~€ 51,00~~



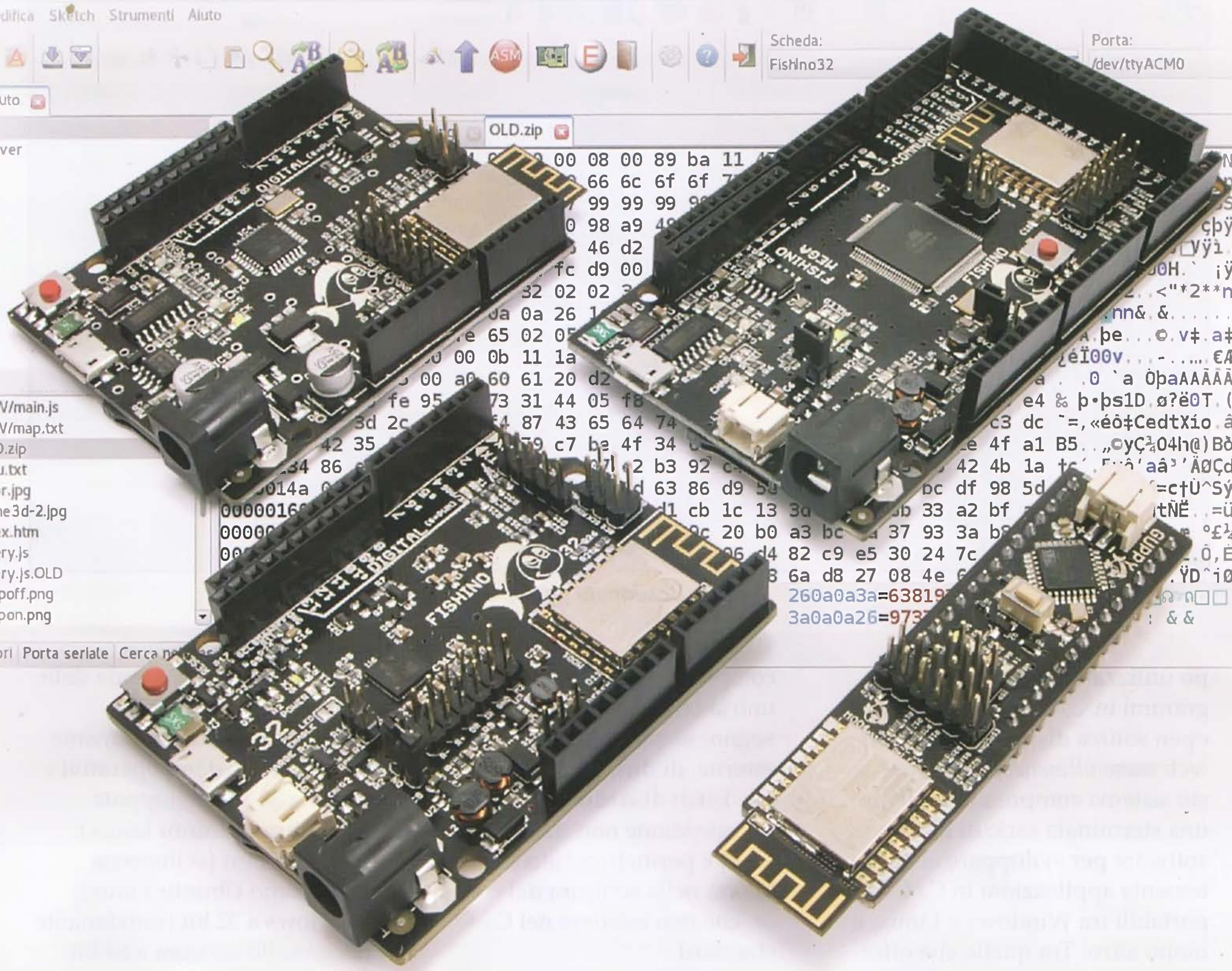


cod. FISHKIT1

FUTURA ELETTRONICA®

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

www.futurashop.it



FishIDE

Un nuovo ambiente di sviluppo nato per le nostre schede Fishino, ma adatto anche alle Arduino ufficiali. Prima puntata.

di MASSIMO DEL FEDELE

Durante lo sviluppo delle schede Fishino, soprattutto quelle a 32 bit, ci siamo resi conto di dover scrivere un gran numero di librerie software e di sketch di esempio e di test dedicati. Se per gli sketch semplici l'IDE di Arduino si è rivelato adeguato, per quelli più complessi e soprattutto per lo sviluppo delle librerie ha iniziato a mostrare i propri limiti. L'IDE di Arduino, infatti, pur essendo in grado di lavorare su più file e di aprire più finestre, non è in grado di mostrare la relazione tra lo sketch e le librerie utilizzate; aprendo più finestre, inoltre, durante gli errori di compilazione non è in grado di mostrare gli stessi nel codice sorgente selezionandoli. Senza contare che l'IDE Arduino è un po' "scarno" per chi, come noi, è abituato alle "comodità" dei moderni sistemi di sviluppo, soprattutto in fase di editing del codice, come per esempio la possibilità di commentare blocchi di codice con un solo comando, di spostare l'indentatura, di eseguire ricerche e sostituzioni in file multipli, eccetera. Abbiamo quindi deciso di sviluppare un nostro IDE prendendo come modello il sistema di svilup-

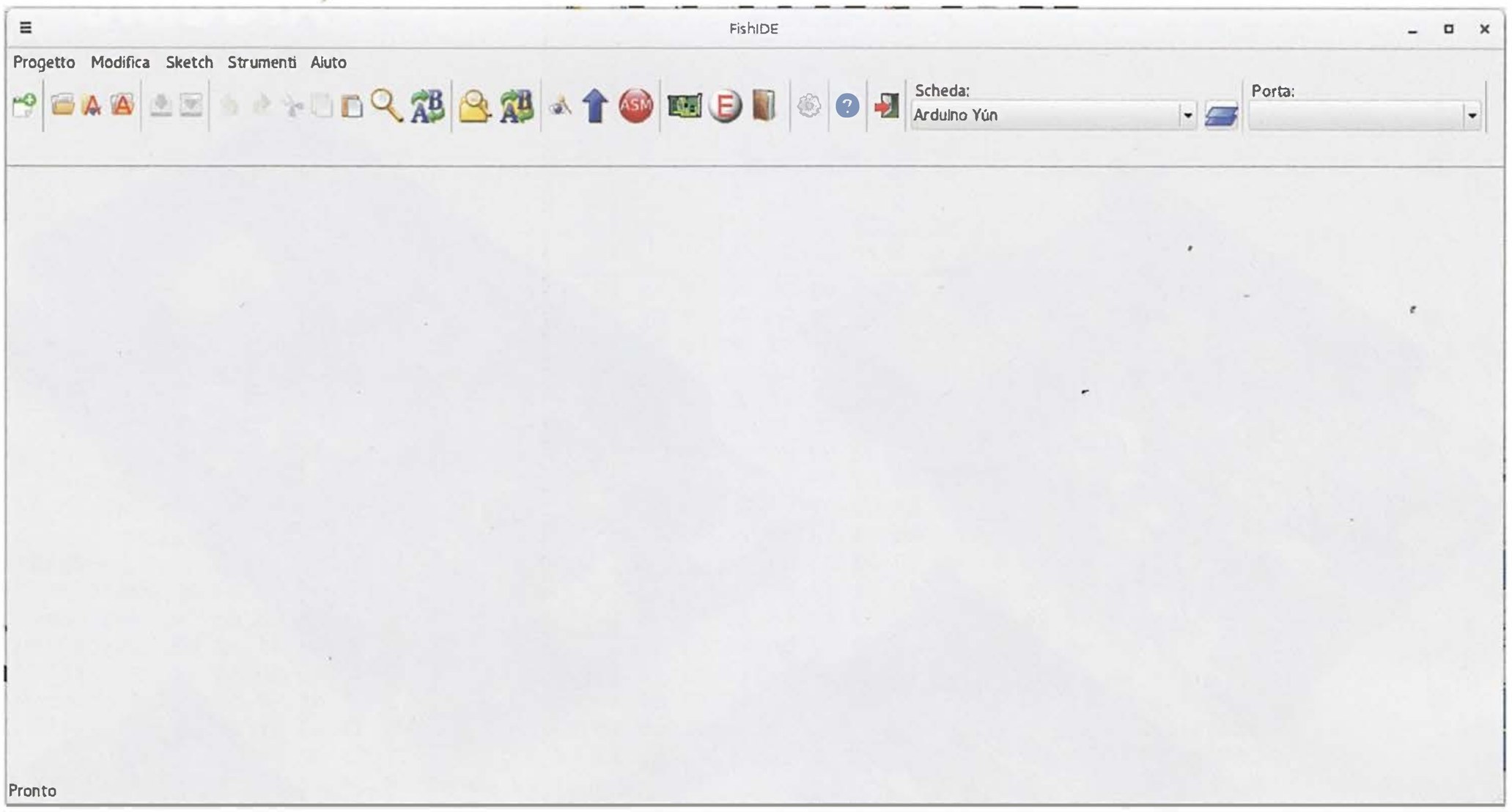


Fig. 1 *- Schermata principale.*

po utilizzato per scrivere programmi in C/C++, un sistema open source disponibile sul sito web *www.ultimatepp.org*. Questo sistema comprende un IDE, una sterminata serie di librerie software per sviluppare efficientemente applicazioni in C/C++ portabili tra Windows e Linux, e molto altro. Tra quello che offre è compreso un ottimo editor per programmi, comprendente tante "comodità" molto apprezzabili durante lo sviluppo. Lo stesso FishIDE è stato sviluppato utilizzando questo sistema. Chiaramente, l'editor è solo il punto di partenza, perché il sistema Arduino è piuttosto differente da un sistema di sviluppo C/C++, soprattutto per le seguenti motivazioni:

- nel C/C++ occorre specificare in qualche modo quali librerie utilizzare; questo si fa solitamente tramite un makefile oppure selezionandole tramite un'interfaccia grafica;
- Arduino, a differenza del C standard, provvede automaticamente ad "agganciarsi" ad alcuni file necessari per la compilazione degli sketch; uno sketch Arduino non ha bisogno, se non utilizza librerie esterne, di direttive #include;
- gli sketch di Arduino hanno un'estensione non standard (.ino) e permettono alcune libertà nella scrittura del codice, che non esistono nel C++ standard.

Abbiamo quindi preso l'editor di cui sopra, l'abbiamo "circondato" da un'interfaccia grafica, aggiungendogli alcuni strumenti, indispensabili e/o soltanto comodi nello sviluppo del software.

INSTALLAZIONE DI FISHIDE

Prima di addentrarci nelle caratteristiche del nostro sistema di sviluppo spendiamo due parole sull'installazione, anche se questo è un termine improprio, visto che l'IDE consiste in un solo file eseguibile compresso (.zip), da posizionare in qualsiasi parte del nostro hard disk.
Non è richiesta alcuna cartella particolare, tutti i file e i programmi necessari verranno scaricati e installati direttamente e automaticamente a seconda delle schede selezionate.
L'IDE è disponibile attualmente per i seguenti sistemi operativi :

- Linux a 32 bit (sviluppata sull'ultimo Ubuntu Linux);
- Linux a 64 bit (sviluppata sull'ultimo Ubuntu Linux);
- Windows a 32 bit (funzionante anche sulle versioni a 64 bit);
- Mac OS/X (versione beta, non nativa, sviluppata su Sierra).

L'ultima versione, quella per Mac, è ancora in beta (al momento della stesura dell'articolo) e presenta alcune problematiche che vedremo in seguito. FishIDE è scaricabile da *www.fishino.it*.
Per utilizzare FishIDE è sufficiente fare doppio clic sul file .zip, decomprimerlo in una cartella a scelta ed eseguirlo.
L'IDE si presenterà all'inizio con una schermata di AutoSetup, che permette di installare in modo automatico i file relativi alle schede di Arduino di base (quelle ad 8 bit) , alcune librerie tra quelle più utilizzate e gli esempi di base.
Non abbiamo volutamente inserito l'installazione automatica

dei package relativi alle schede Fishino, essendo i package piuttosto corposi e, soprattutto, volendo evitare di imporre l'installazione di schede che poi non verranno utilizzate.
Pur essendo possibile saltare questa fase e procedere manualmente in seguito (se si vuole utilizzare l'IDE solo per schede differenti, per esempio, è conveniente), consigliamo di eseguire questa semplicissima operazione in modo da avere il nostro sistema di sviluppo pronto all'uso da subito.
La finestra di dialogo dell' AutoSetup propone la richiesta: "Eseguire il setup automatico?"; cliccando su *Si* inizia il setup.
La procedura durerà alcuni minuti, nei quali appariranno sullo schermo i file in fase di installazione. Al termine apparirà la schermata principale del FishIDE (**Fig. 1**).
Prima di addentrarci nell'utilizzo vediamo subito di completare l'installazione aggiungendo le schede Fishino e le relative librerie; selezioniamo innanzitutto il menu delle impostazioni, che potremo aprire sia selezionandolo dal menu a tendina, alla voce *Strumenti>Impostazioni* oppure, più semplicemente, cliccando sull' "ingranaggio" presente nelle icone degli strumenti; si aprirà la finestra di dialogo proposta dalla **Fig. 2**.
Lasciamo perdere al momento le impostazioni varie che vedremo in dettaglio in seguito e selezioniamo il tab "Percorsi dei file", allorché appare la finestra di dialogo di **Fig. 3**.
Vediamo qui di seguito gli elementi della schermata in dettaglio nei suoi vari elementi.

- Cartella degli sketch: questo è il percorso della cartella principale che conterrà i nostri sketch; possiamo tranquillamente utilizzare la stessa dell'IDE di Arduino, oppure sceglierne una nuova. L'AutoSetup dovrebbe essere in grado di trovare la cartella automaticamente; nel caso in cui il campo fosse vuoto, è possibile selezionarlo manualmente utilizzando il pulsante con la cartellina sulla destra.
- URL per i package: analogamente all'IDE di Arduino è possibile specificare più URL (percorsi internet) contenenti le descrizioni ed i file di schede. Il nostro AutoSetup ha precompilato questo campo con le URL relative alle schede originali Arduino, le schede Fishino ed i moduli WiFi ESP8266 e derivati. È possibile aggiungere un numero illimitato di URL per schede aggiuntive; i percorsi si trovano solitamente sul sito del produttore.
- URL per le librerie : questa è una novità rispetto all'IDE di Arduino, visto che in quest'ultima non è possibile inserire dei percorsi addizionali per le librerie; l'unico percorso,

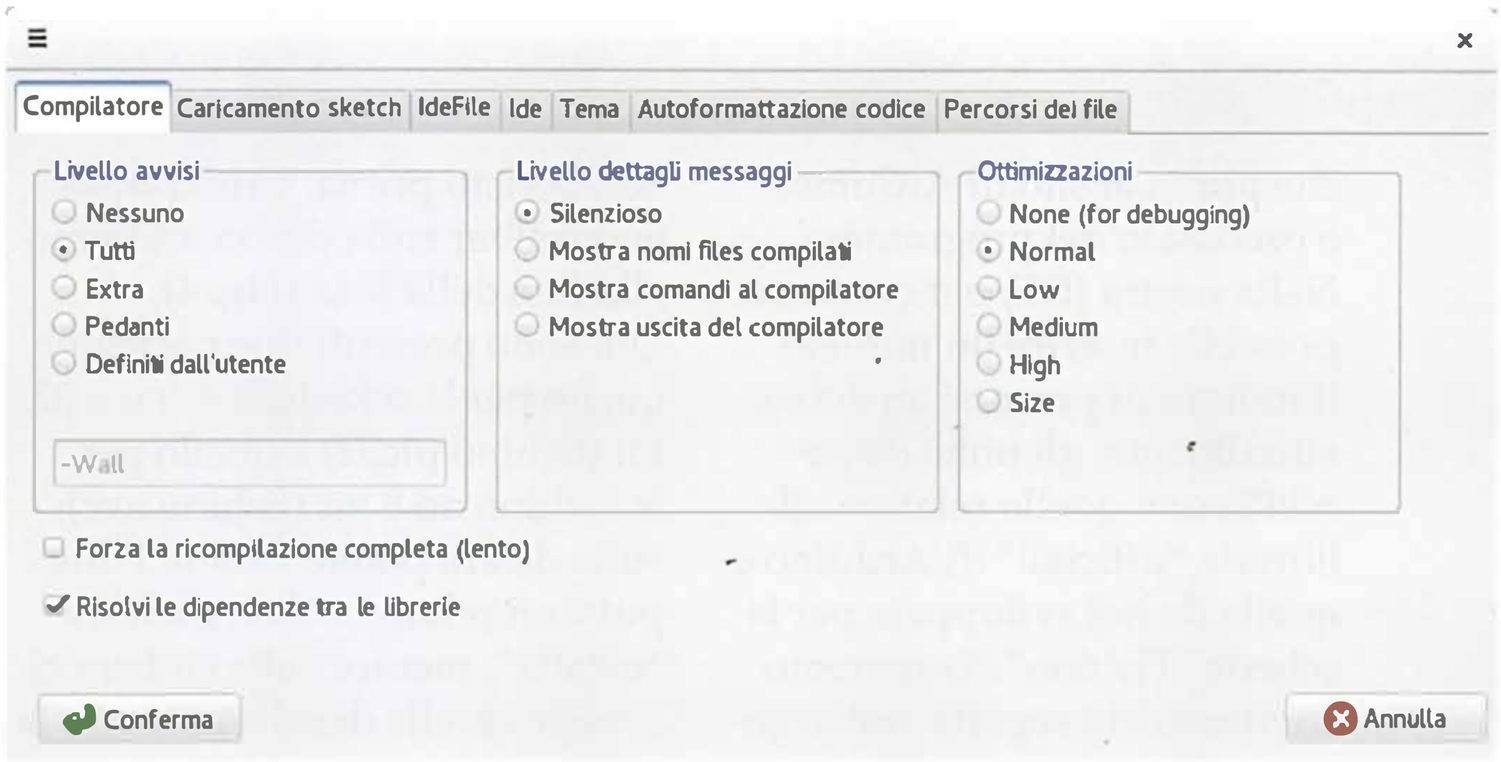


Fig. 2 *- Scheda Compilatore.*

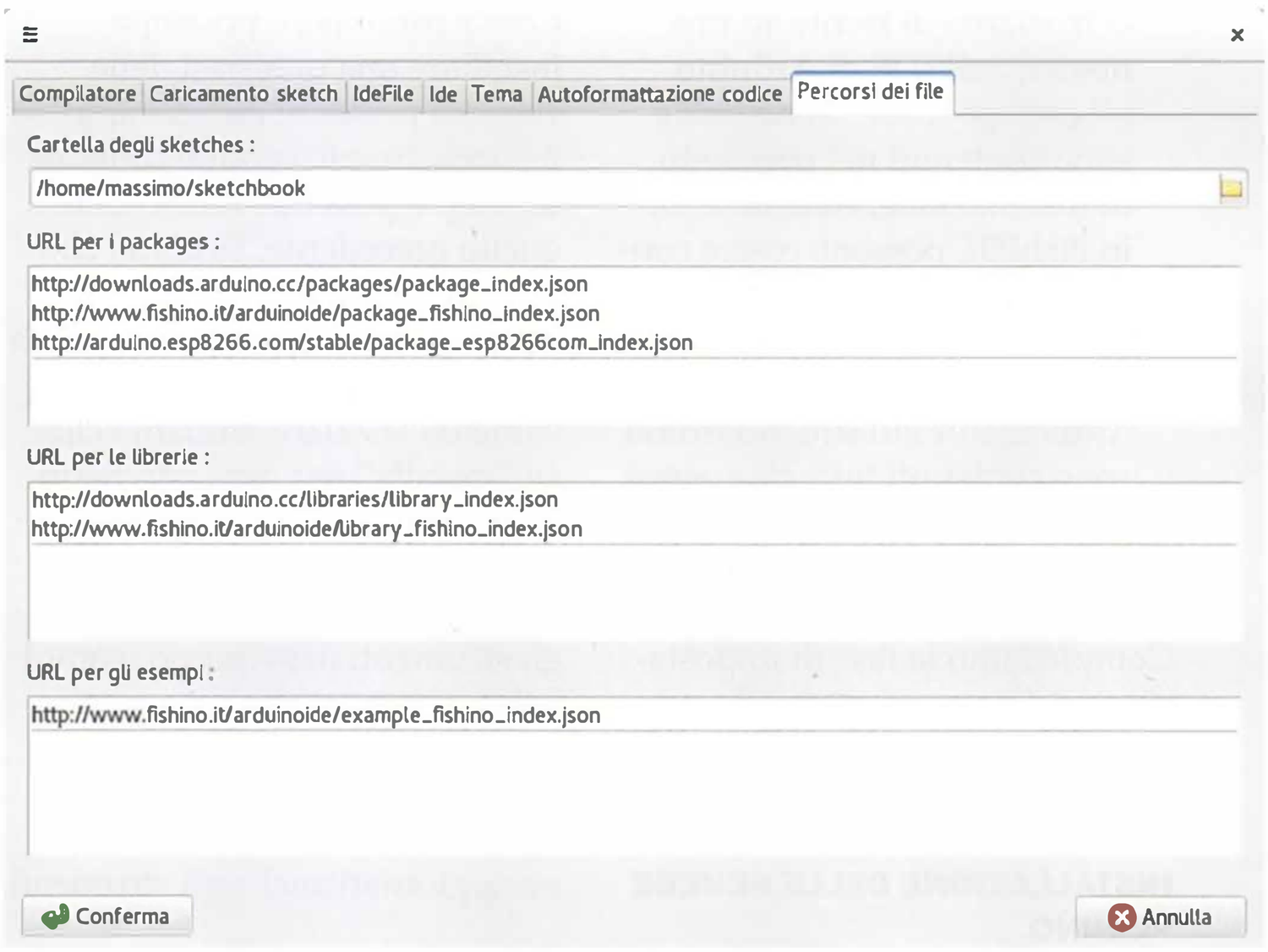


Fig. 3 *- Tab Percorsi dei file.*

che punta al sito di Arduino, è prefissato nel programma. Nella nostra IDE, per contro, è possibile inserire un numero illimitato di percorsi anche se, attualmente, gli unici disponibili sono quello relativo alle librerie "ufficiali" di Arduino e quelle da noi sviluppate per le schede "Fishino". Torneremo comunque in seguito sull'argomento librerie.

- URL per gli esempi: anche qui ci troviamo di fronte ad una novità; nell'IDE di Arduino gli esempi "base" sono fissi e sono contenuti nel pacchetto di installazione. Al contrario, in FishIDE possono essere contenuti in vari server remoti e da questi installati, ottenendo quindi una notevole flessibilità. Attualmente sul sito di Fishino sono contenuti tutti gli esempi base di Arduino e qualche demo per le schede Fishino.

Completiamo la fase di impostazione dei percorsi cliccando sul pulsante "Conferma" e torniamo all'IDE.

INSTALLAZIONE DELLE SCHEDE FISHINO

Come abbiamo accennato, l'AutoSetup provvede ad un'installazione minima contenente il supporto alle sole schede di Arduino, per evitarvi di perdere tempo nel download di librerie e driver di schede che poi potrebbero non essere utilizzate.
Per installare schede aggiuntive utilizziamo il "Gestore di schede", accessibile sia dal consueto menu *Strumenti>Gestore schede* che dall'apposito pulsante della toolbar. Cliccandovi visualizziamo la relativa finestra di dialogo nella quale risulta già installato il primo package: quello corrispondente alle schede Arduino di base, grazie all'AutoSetup visto prima. Utilizzando la scrollbar sulla destra andiamo alla fine della lista (**Fig.** 4).
Qui sono presenti due package: quello per le schede Fishino a 32 bit (fishino pic32) e quello per le Fishino ad 8 bit (fishino avr); sulla destra potete vedere i due pulsanti relativi con la dicitura "installa", mentre sulla sinistra ci sono le caselle di selezione con la versione dei pacchetti (in questo caso, 7.0.0 per entrambi).
Com'è intuibile, è possibile installare una qualsiasi delle versioni presenti; per esempio, se il pacchetto più recente contiene un bug, è possibile retrocedere a quello precedente. Siccome vogliamo installare i pacchetti più recenti, controlliamo nella casella che l'ultima versione sia effettivamente la 7.0.0 e facciamo clic su "installa" per ogni pacchetto.
La prima installazione di un pacchetto può durare parecchi minuti, dovendo scaricare tutti gli strumenti di sviluppo (compilatori, eccetera) che spesso sono piuttosto corposi; solitamente negli aggiornamenti questo non avviene (a meno che non vengano aggiornati anche gli strumenti, cosa che capita raramente) ed il procedimento risulta velocissimo.
Completata l'installazione la schermata presenterà la scritta "rimuovi" al posto di "installa" sui pulsanti a destra, ad indicare che il pacchetto è installato ed è possibile rimuoverlo. Selezionando invece una versione precedente, la scritta passerà a "retrocedi", mentre per un pacchetto più nuovo sarà "aggiorna", in modo da permetterci un controllo completo di quanto installato.

INSTALLAZIONE DELLE LIBRERIE DI FISHINO

Una volta installato il package con le definizioni delle schede, per le nostre Fishino è indispensabile scaricare anche una serie di librerie software che permettono di gestirne le caratteristiche aggiuntive rispetto alle schede Arduino; per questo utilizzeremo il gestore delle librerie, raggiungibile tramite il menu *Strumenti>Gestore librerie* oppure dall'apposito pulsante della toolbar.
Lo strumento, una volta avviato, si presenta come nella **Fig. 5**; il suo utilizzo è semplicissimo,

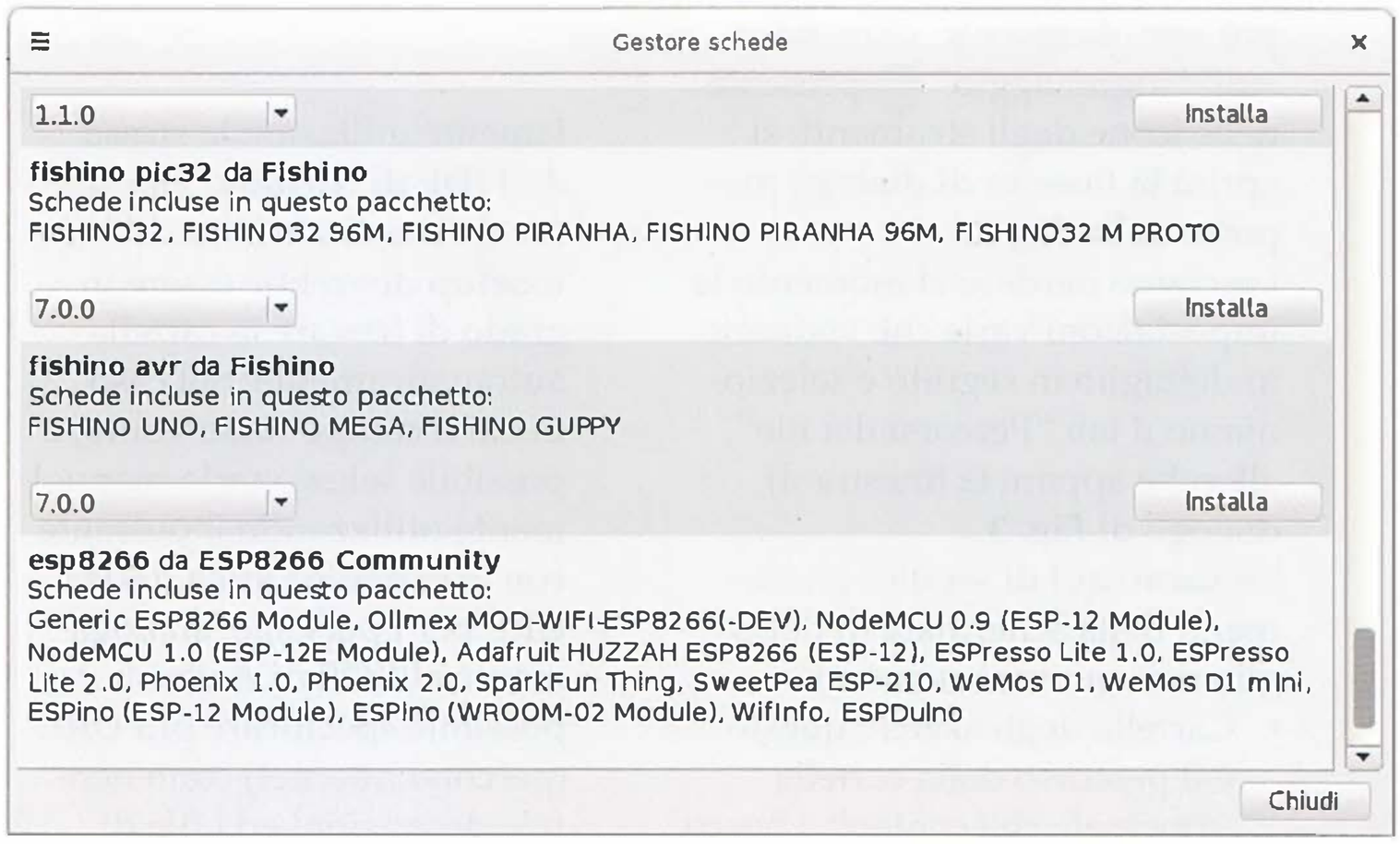


***Fig. 4** - Gestore schede.*

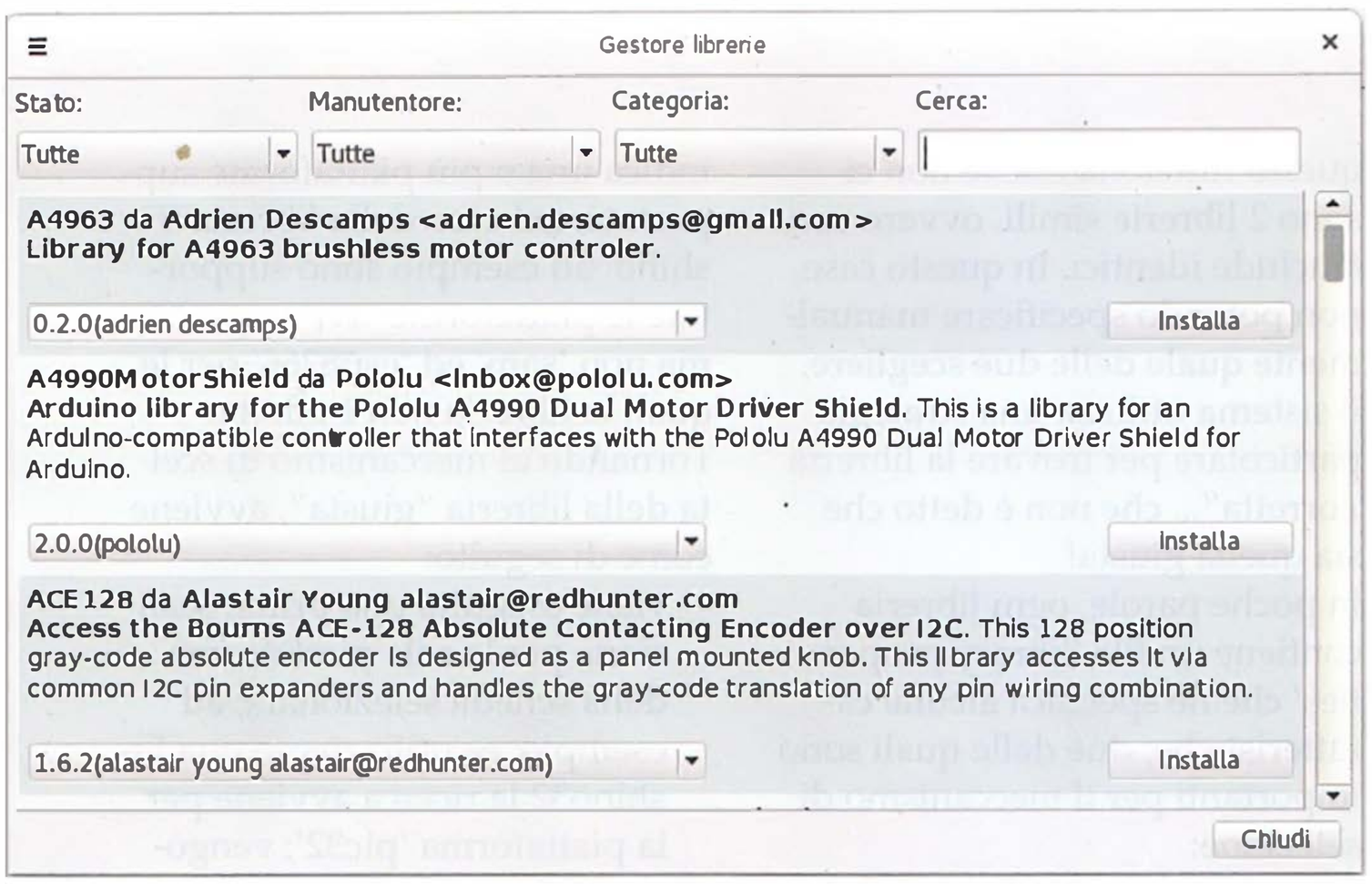


***Fig. 5** - Gestore librerie.*

ma permette di eseguire ricerche e selezioni delle librerie secondo vari criteri. Nella **Fig. 5** potete distinguere i seguenti elementi:

- in alto, tre caselle di selezione (Stato, Manutentore e Categoria) ed una casella di testo dove inserire una parte del nome o della descrizione per restringere la ricerca;
- sotto alla prima riga (analogamente al gestore di pacchetti) l'elenco delle librerie disponibili con le caselle di selezione versione sulla sinistra ed i pulsanti di installazione sulla destra.

Potete tranquillamente provare le varie opzioni e, tramite la barra di scorrimento sulla destra, vedere le librerie che sono state installate automaticamente dall'AutoSetup, che presenteranno il pulsante di destra con la scritta "rimuovi" al posto di "installa". Per installare le librerie di Fishino rimettiamo tutto a posto e, dalla casella Manutentore, selezioniamo *Fishino;* in modo da restringere la visualizzazione alle sole librerie da noi gestite; la schermata apparirà come in **Fig. 6**.
È sufficiente fare clic sul pulsante "installa" per ognuna delle librerie a cui si è interessati.
La libreria indispensabile per il funzionamento del modulo WiFi è la Fishino; le altre sono opzionali ed offrono funzionalità aggiuntive.

COLLOCAZIONE DELLE LIBRERIE

Chi utilizza abitualmente l'IDE di Arduino sa che le librerie vengono inserite in una cartella 'libraries' che si trova all'interno degli sketch. Aprendo però la cartella degli sketch gestita da FishIDE non troverete le librerie installate dal gestore, che vengono caricate in un'altro posto; nella 'libraries' restano tutte le librerie installate "a mano" o gestite dall'IDE di Arduino, che vengono comunque correttamente rilevate ed utilizzate da FishIDE.
Come mai un posizionamento "fuori standard" ? Principalmente per evitare che un utente inesperto, ma anche uno più esperto ma distratto, possa metterci mano per sbaglio. Tramite FishIDE è infatti possibile caricare ed aggiornare le librerie situate su un server remoto solo se queste non vengono "pasticciate" a mano; inoltre, se uno le modificasse a mano (magari per migliorarne qualche caratteristica) le modifiche verrebbero sovrascritte da un eventuale aggiornamento, o addirittura cancellate se disinstallassimo la libreria.
Abbiamo quindi optato per gestire le librerie su tre livelli, ben separati:

- le librerie delle schede, cioè quelle vincolate all'hardware,



***Fig. 6** - Visualizzazione, in Gestore Librerie, delle sole librerie Fishino.*

che sono contenute nei pacchetti del gestore di schede;
- le librerie "di sistema", cioè quelle gestite dal Gestore Librerie;
- le librerie "utente", che possono essere installate a mano o caricando un file .zip nella solita cartella 'libraries'.

Le prime sono gestite esattamente come fa l'IDE di Arduino; si trovano in una cartella del package per la scheda selezionata e sono "intoccabili", o quasi, come vedremo in seguito.
Le seconde, gestite dal Gestore Librerie, si trovano in un'altra cartella di sistema, ed anch'esse sono quasi "intoccabili".
Le terze, nella cartella "libraries" possono essere "pasticciate" liberamente senza il pericolo di vederle cancellate o sovrascritte dal Gestore Librerie.
Vedremo in seguito, parlando delle impostazioni in dettaglio, che un file delle librerie nei primi due livelli, anche se aperto, **non è modificabile**, quindi apparirà nell'editor del codice in modalità sola lettura, in modo da evitare modifiche non volute, a meno di non abilitare un'apposita casella nelle impostazioni che attiva la scrittura dei file di sistema.
In questo modo potremo comunque vedere e studiare il codice delle librerie, vedere dove si verificano eventuali errori o avvisi di compilazione, ma non potremo fare danni a meno di ...non volerlo davvero!
Ma cosa succede se ci sono due librerie con lo stesso nome in due posti diversi? Come noto, a differenza di un sistema di sviluppo C/C++ standard, Arduino gestisce le librerie in modo quasi automatico; è sufficiente inserire una #include nello sketch ed l'IDE si occupa di trovare la libreria corretta. Ovviamente questo funziona finchè non ci sono 2 librerie simili, ovvero con #include identici. In questo caso, non potendo specificare manualmente quale delle due scegliere, il sistema utilizza una strategia particolare per trovare la libreria "corretta"... che non è detto che sia quella giusta!
In poche parole, ogni libreria contiene un file 'library.properties' che ne specifica alcune caratteristiche, due delle quali sono importanti per il meccanismo di selezione:
- la piattaforma supportata;
- gli include file che definiscono la libreria.

Vediamoli in dettaglio iniziando dal primo: la piattaforma.
La piattaforma definisce la tipologia delle schede; nel caso delle schede Fishino abbiamo due piattaforme definite, 'avr' per le schede ad 8 bit e 'pic32' per quelle a 32 bit. I package di Arduino forniscono la piattaforma 'avr' come base, ed altre, per esempio 'sam' per Arduino Due. Le schede con il modulo ESP8266 forniscono la piattaforma 'esp8266'.
Cosa c'entra con le librerie?
Presto detto! Alcune librerie non vanno a toccare direttamente l'hardware delle schede, ma utilizzano solo i comandi standard di Arduino, per esempio 'pinMode()' e 'digitalWrite()'; non accedono quindi direttamente ai registri del controller e non utilizzano caratteristiche che possono cambiare in base alla piattaforma scelta. In queste librerie, nel file 'library.properties' il parametro 'platform' è indicato con un asterisco, '*', ad indicarne l'universalità.
Altre librerie per contro sono specifiche per alcune piattaforme e non è detto, anzi, è quasi sicuro che non funzionino su altre; in queste il parametro 'platform' indica una o più piattaforme supportate; nel caso della libreria 'Fishino' ad esempio sono supportate le piattaforme 'avr' e 'pic32' ma non 'sam' ed 'esp8266' per le quali la libreria non è adatta.
Tornando al meccanismo di scelta della libreria "giusta", avviene come di seguito:
- viene eseguita una prima scansione per la sola piattaforma della scheda selezionata; ad esempio, se utilizziamo una Fishino32 la ricerca avviene per la piattaforma 'pic32'; vengono SCARTATE tutte le librerie che non dichiarano ESPLICITAMENTE quella piattaforma, anche quelle universali con '*';
- la scansione avviene nelle librerie utente (cartella 'libraries', che ha la precedenza), poi nella cartella delle librerie di sistema (quelle gestite dal Gestore Librerie) e, per ultimo, in quelle della scheda (che ovviamente appartenendo alla piattaforma della scheda la supportano sicuramente);
- alla prima libreria trovata (ovvero alla prima che dichiara un file #include come quello da noi inserito nello sketch), la ricerca si ferma e viene utilizzata quella libreria;
- se non viene trovata nessuna libreria, la ricerca riparte da capo considerando anche le librerie 'universali', cioè quelle che dichiarano '*' come piattaforma, sempre nell'ordine librerie utente-sistema-scheda;
- nel caso anche qui non venga trovato nulla, il sistema tenta l'utilizzo di una libreria specifica per una piattaforma 'sbagliata'; alcune librerie di Arduino, per esempio, dichiarano la piattaforma 'avr' ma in pratica funzionano su molte più schede;
- infine, se anche in quest'ultimo caso non si trova la libreria,

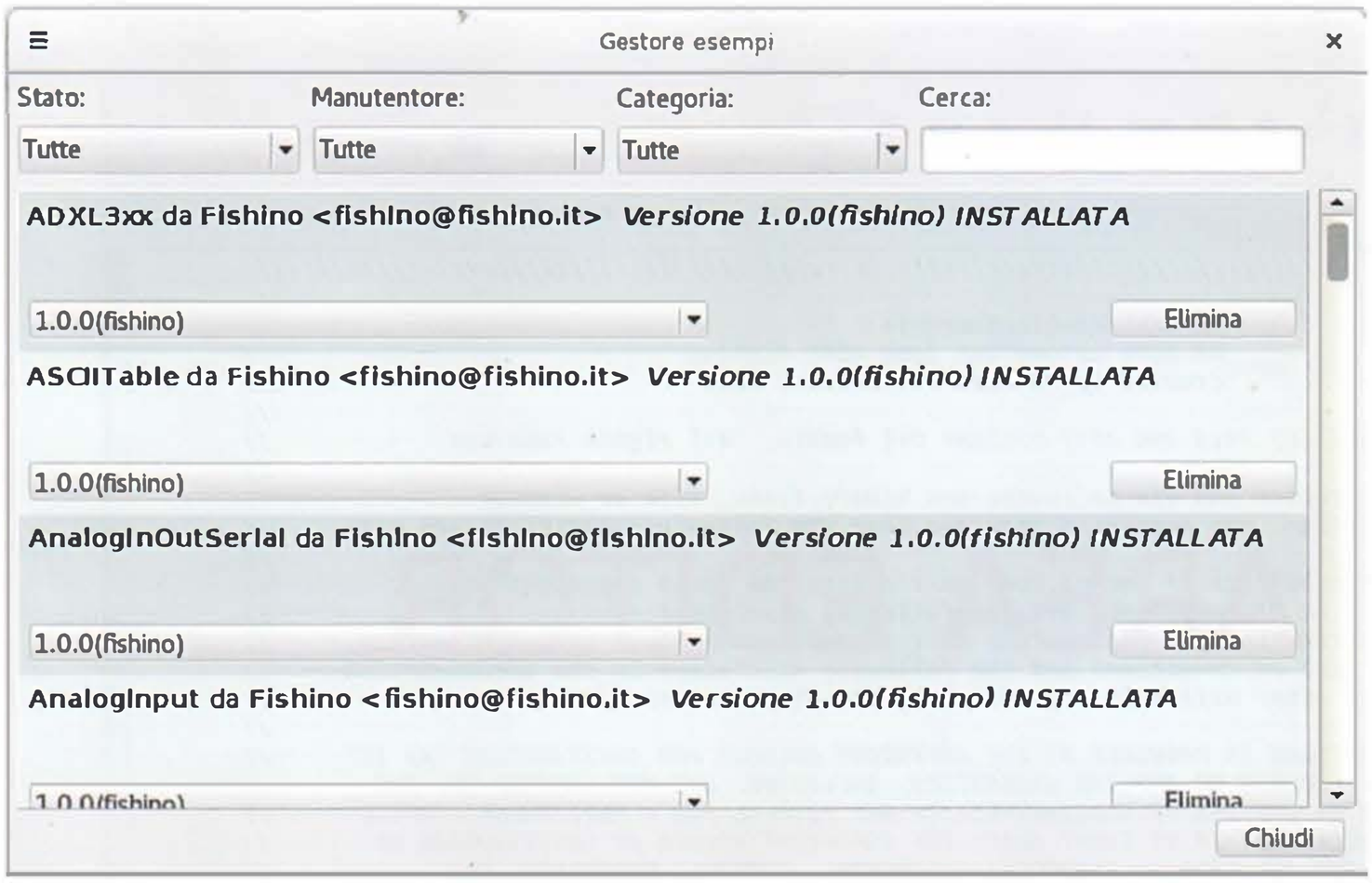


***Fig. 7** - Finestra Gestore esempi.*

viene mostrato un errore di compilazione.

Con questo metodo FishIDE utilizza quindi sempre la libreria che appare più adatta e, nel caso volessimo modificarne una di sistema, ci basterà copiarla nella cartella 'libraries' e lavorare su quella, certi che avrà la precedenza sull'originale.

INSTALLAZIONE DEGLI ESEMPI

Anche gli esempi sono installabili, aggiornabili e rimuovibili tramite un apposito modulo, il Gestore esempi, richiamabile tramite il menu *Strumenti>Gestore esempi*, oppure con l'apposito pulsante della toolbar. Per brevità mostriamo solo la schermata del modulo, il cui funzionamento è assolutamente intuitivo (**Fig. 7**). Come per il Gestore librerie, anche qui disponiamo di alcuni selettori per restringere la visualizzazione agli esempi richiesti, insieme ai consueti selettori di versione e pulsanti di installazione.
Anche gli esempi vengono installati in una cartella di sistema, nascosta all'utente, come le librerie; è comunque possibile sia aprirli e provarli direttamente (si aprono in modalità sola lettura) che importarli nella cartella degli sketch per poter essere modificati, come vedremo a breve parlando della schermata principale di FishIDE.

UTILIZZIAMO FISHIDE

Ed eccoci finalmente arrivati alla parte pratica e quindi all'utilizzo del nostro nuovo IDE: lanciamo FishIDE e, apertasi la finestra di lavoro, osserviamo i pulsanti a sinistra della toolbar (**Fig. 8**).
Questi pulsanti corrispondono al menu 'Progetto' e permettono di aprire e salvare sketch, rinominarli ed importare gli esempi.
Che cos'è un 'progetto'? Ebbene, a differenza dell'IDE di Arduino, che si limita a gestire i file in una cartella, il nostro FishIDE utilizza un vero e proprio file di progetto, nella stessa cartella, avente il nome dello sketch e l'estensione '.fish' (abbiamo una fantasia illimitata!!!).
Questo file attualmente è poco utile, ma destinato ad estensioni future che appariranno a breve.
In sintesi, raccoglie tutti i file presenti nella cartella, permetterà di "metterli in ordine" di visualizzazione nel FishIDE, di raggrupparli e di nasconderne alcuni che potrebbero essere dei documenti di contorno, quali PDF o altro.



***Fig. 8** - Pulsanti per l'accesso ai comandi del menu Progetto.*

Il file .fish contiene anche l'elenco delle finestre aperte quando chiudiamo FishIDE e le posizioni del cursore in ogni editor; ci permetterà quindi di chiudere uno sketch e ritrovarcelo aperto e con il cursore posizionato allo stesso punto in cui l'avevamo lasciato. Non solo, nel caso avessimo aperto anche altri file, per esempio i sorgenti di una libreria, anche questi verranno ripristinati.
Torniamo ora ai pulsanti della toolbar, le cui funzioni sono le seguenti:

- **Nuovo progetto**; permette di creare uno sketch da zero;
- **Apri progetto**; permette di aprire uno sketch esistente;
- **Importa esempio**; copia uno sketch di esempio nella cartella degli sketch e lo apre;
- **Apri esempio**; apre un esempio in modalità sola lettura senza copiarlo;
- **Salva sketch**; salva lo sketch corrente;
- **Salva sketch con nome**; permette di salvare lo sketch corrente in un'altra cartella.

Per iniziare, utilizziamo il comando '**Importa esempio**', scegliendo un esempio abbastanza complesso che ci permetterà di esaminare tutte le principali caratteristiche di FishIDE; una volta cliccato sul pulsante corrispondente, apparirà un menu da cui impartire il comando '**Esempi delle librerie di sistema**', perché vogliamo usare un esempio della



Fig. 9 *- Finestra aperta dal comando Importa esempio.*

libreria **Fishino**, e quindi l'esempio '**FishinoHomeAuto**'. In **Fig. 9** vedete come apparirà la schermata una volta aperto l'esempio; in essa abbiamo indicato in rosso i vari elementi dell'interfaccia.

- l'**AREA EDITOR** è, chiaramente, l'area in cui verrà scritto il nostro sketch e/o comunque il codice della libreria, o qualunque file di testo.
- l'**AREA LIBRERIE** permette di selezionare le librerie utilizzate (o comunque librerie che si vogliono esaminare, come vedremo in seguito) o i file dello sketch, alla voce 'Principale', la prima della lista. L'ultima voce ('<altri file>') ci permetterà di aprire qualsiasi file esterno al nostro progetto, per esempio il file di un altro sketch da cui vogliamo copiare qualcosa. La cosa importante è che quest'area NON contiene nomi di file ma è una sorta di raggruppamento logico dei medesimi.
- l'**AREA FILE** contiene invece l'elenco dei file appartenenti alla voce selezionata nell' AREA LIBRERIE.
- l'**AREA FINESTRE AUSILIARIE** contiene una serie di tab che ci permetteranno di visualizzare i messaggi di compilazione, il monitor seriale, gli errori ed i file trovati col comando 'Cerca nei file' che vedremo in seguito.
- il **SELETTORE SCHEDA** ci permetterà di selezionare la nostra scheda tra quelle disponibili; nell'esempio abbiamo selezionato una Fishino32.
- il **SELETTORE PORTA** ci permetterà di selezionare la porta a cui è connessa la nostra scheda, tra quelle disponibili; potrà essere anche una porta di rete.
- il **TAB PROGETTO** ci permetterà di passare da un progetto all'altro; FishIDE permette infatti di aprire più progetti nella stessa schermata e di passare facilmente con un click da uno all'altro per potere, ad esempio, copiare spezzoni di codice.
- il **TAB FILE** ci permetterà di passare da un file all'altro tra quelli aperti nel progetto corrente.

All'importazione dell'esempio vediamo aprirsi il **TAB PROGETTO** contenente il nome **FishinoHomeAuto**, che è il nome del nostro sketch ed il **TAB FILE** con il file **FishinoHomeAuto.ino**; nel **AREA LIBRERIE** vediamo, oltre alla voce '**Principale**' (preselezionata) l'elenco delle librerie usate dal progetto, oppure non usate ma aperte manualmente (in grigetto).
Avendo selezionato la voce '**Principale**' nell'AREA LIBRERIE, nell'AREA FILE vedremo l'elenco dei file contenuti nella cartella del nostro sketch.
Per familiarizzare con il nostro IDE, proviamo a scegliere un file (nell' AREA FILE) che sia un po' "strano", ovvero non un file di testo ma un'immagine; per

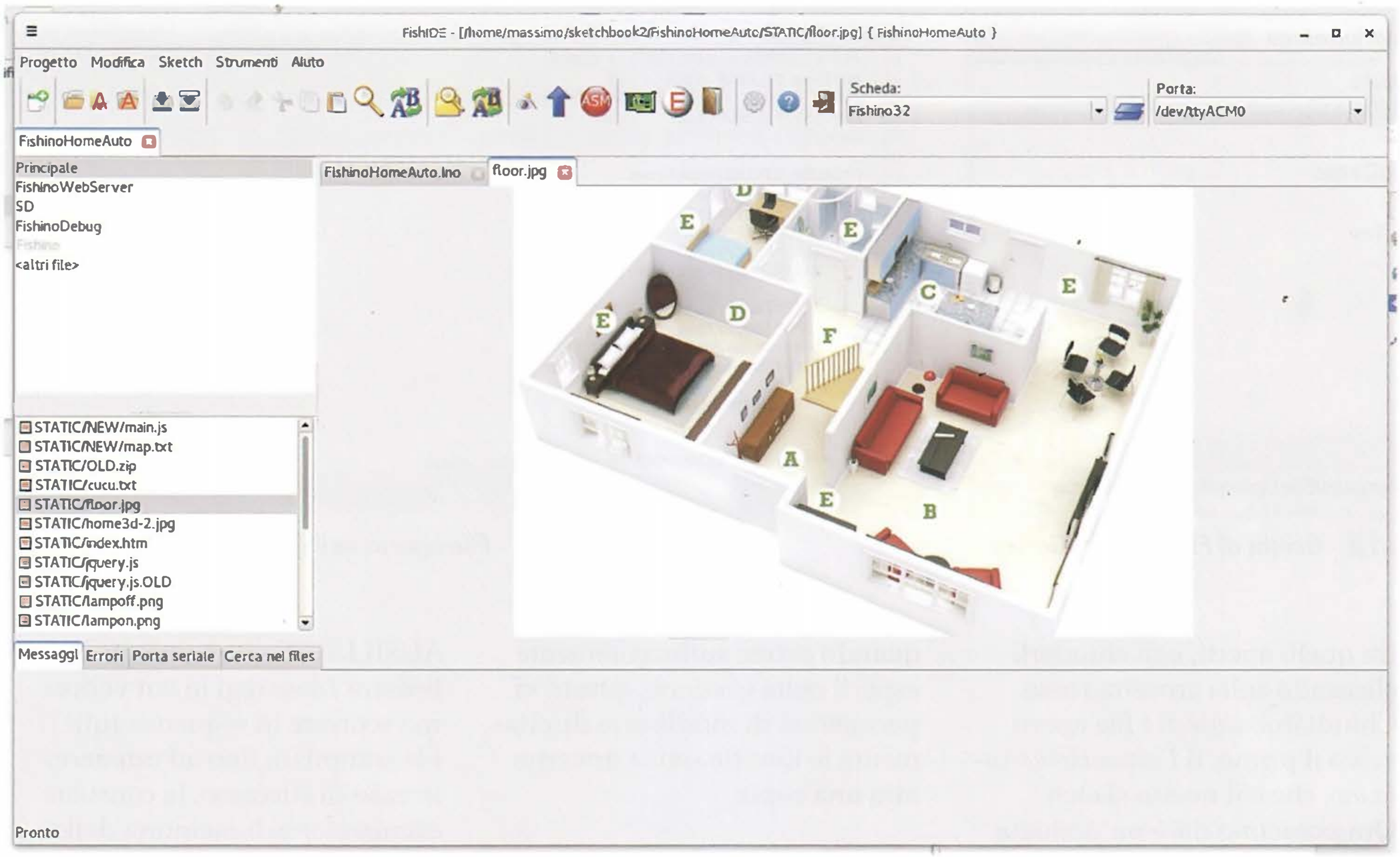


Fig. 10 *- Il file immagine importato in AREA FILE.*

esempio il file STATIC/floor.jpg (**Fig. 10**).
Sorpresa: FishIDE è in grado di visualizzare anche le immagini! Attualmente il supporto per le immagini è piuttosto limitato, non tanto per i formati ma per quello che ci si può fare : l'immagine viene visualizzata così com'è, senza scalarla (quindi potrebbe anche uscire dalla finestra, come parzialmente avviene nel nostro esempio) e senza poterla nè spostare nè modificare. Zoom e pan verranno aggiunti in una versione successiva; non prevediamo di inserire comandi per modificare le immagini, esistendo già molti programmi specializzati che lo farebbero comunque meglio. Si tratta nulla di più di una "comodità" che ci permette di visualizzare un file di immagine.
Tutti i file con formato non riconosciuto, per esempio un file .zip, verranno visualizzati come file esadecimali, anche qui senza possibilità di modifica (**Fig. 11**).
La visualizzazione di file in formato esadecimale risulta comodissima in casi particolari, per esempio per cercare le stringhe di testo in un file binario scaricato da una scheda.
Come avrete sicuramente notato, man mano che apriamo nuovi file cliccandoli nell'AREA FILE appaiono nuovi tab nel TAB FILE (**Fig. 12**). Questi tab permettono di passare da un file all'altro

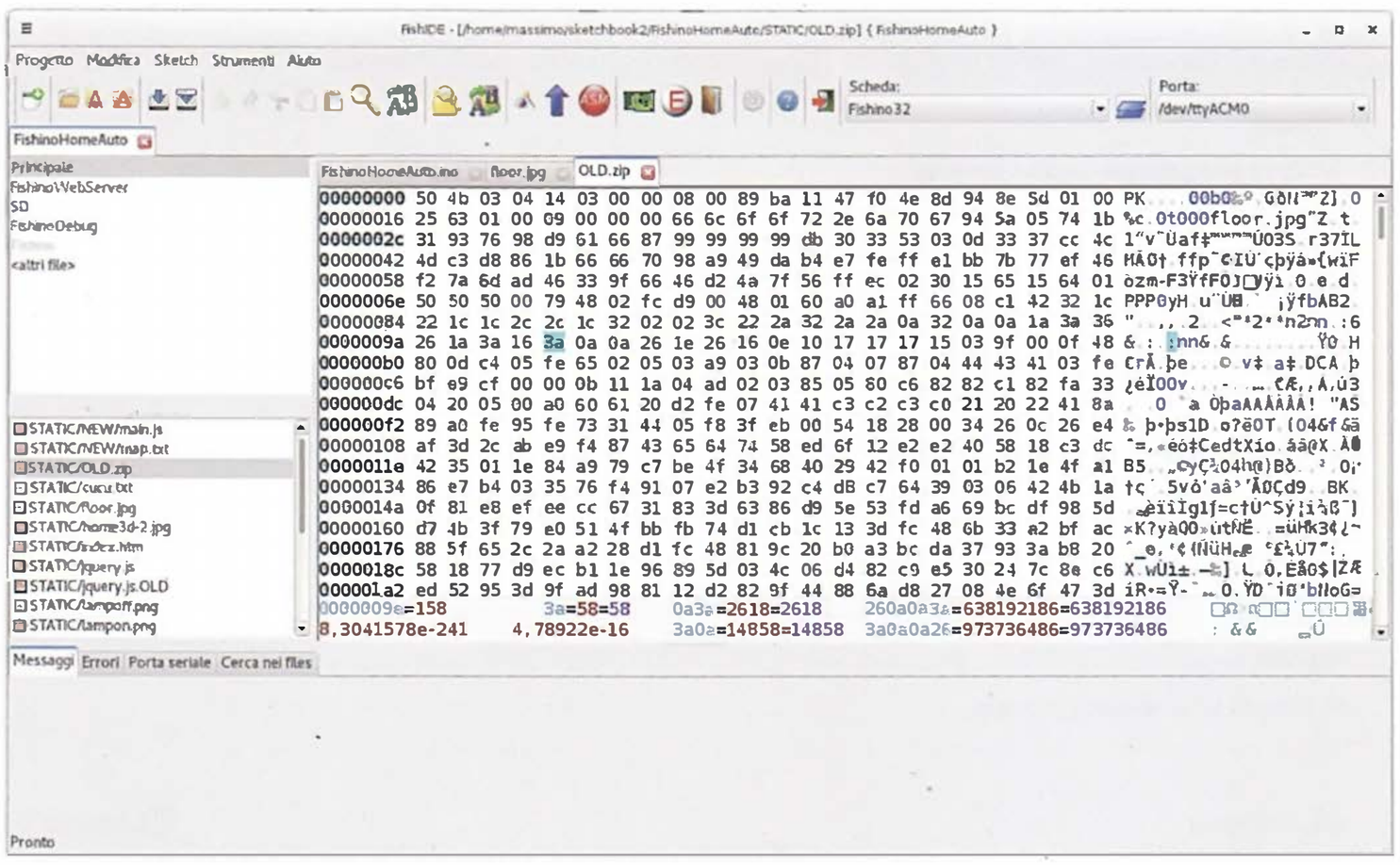


Fig. 11 *- Visualizzazione di file in formato non riconosciuto.*



Fig. 12 *- Tab corrispondenti ai file aperti.*



Fig. 13 - *Scelta di Fishino WebServer.*

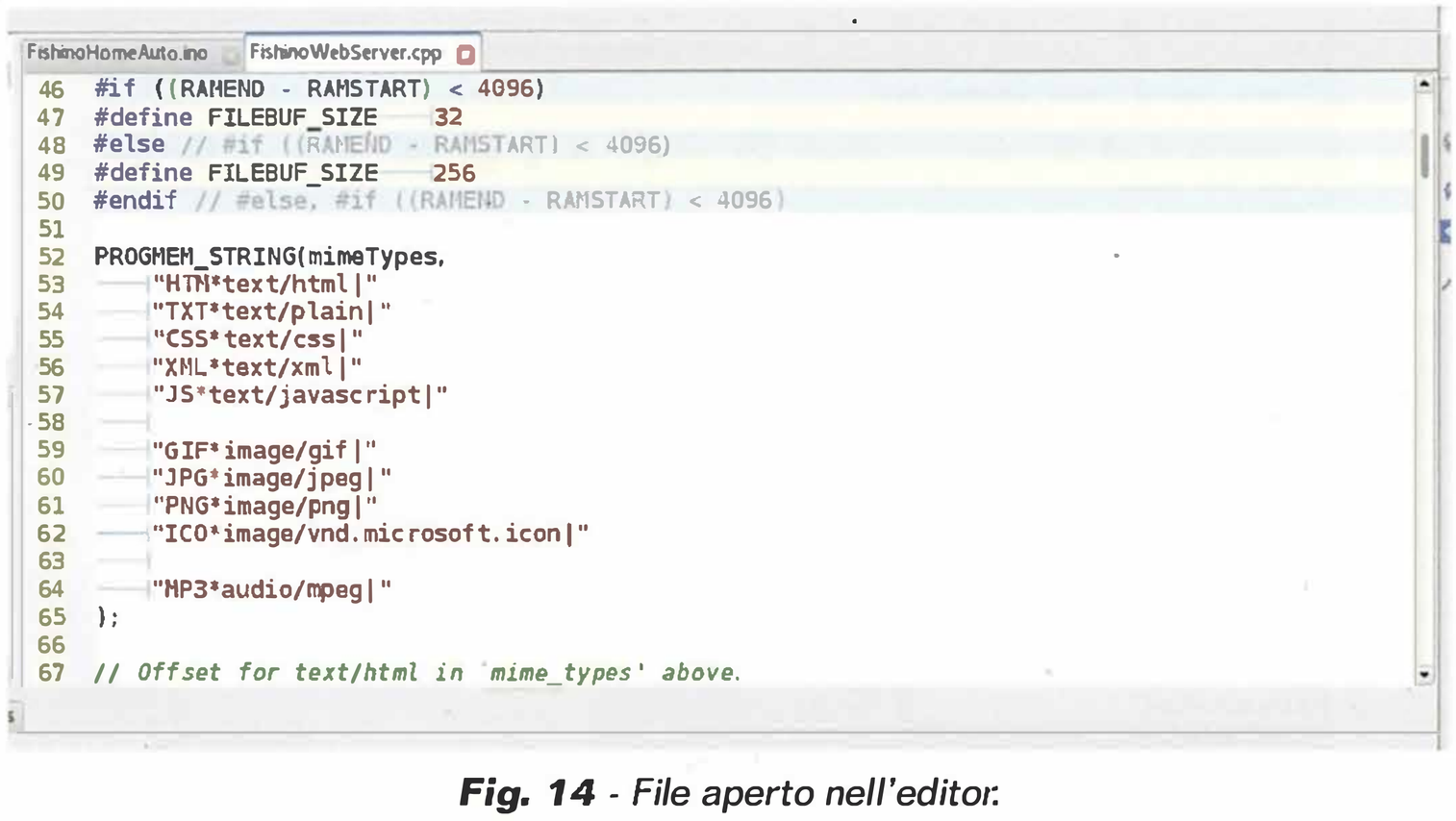


Fig. 14 - *File aperto nell'editor.*

tra quelli aperti, e di chiuderli cliccando sulla crocetta rossa.
Chiudiamo quindi i file aperti salvo il primo, il *FishinoHomeAuto.ino*, che è il nostro sketch.
Ora possiamo dare un'occhiata alle librerie utilizzate: nel riquadro AREA LIBRERIE selezioniamo la voce FishinoWebServer (**Fig. 13**). Nel riquadro AREA FILE utilizziamo la scrollbar per scorrere l'elenco fino a visualizzare il file *FishinoWebServer.cpp* e selezioniamolo. Il corrispondente file verrà aperto e visualizzato nell'editor (**Fig. 14**).
Se nelle impostazioni non avete modificato alcunché, vedrete che il file *FishinoHomeAuto.ino*, che abbiamo preso dagli esempi importandolo (quindi creandone una copia nella nostra cartella degli sketch) è modificabile; il file FishinoWebServer, per contro, appartenendo ad una libreria di sistema non è modificabile, quindi non potrete scrivere nella corrispondente finestra dell'editor.
Questo comportamento dell'IDE può essere modificato tramite le impostazioni (**Fig. 15**) alla voce '**Imposta le librerie di sistema a sola lettura**'; deselezionando la casella potrete modificare i file.
All'inizio vi sconsigliamo di abilitare la scrittura, per non vanificare la protezione dei file importanti da parte dell'IDE; in seguito, quando sarete sufficientemente esperti nella gestione, questo vi permetterà di modificare direttamente le librerie senza doverne fare una copia.

COMPILAZIONE E CARICAMENTO DELLO SKETCH

Restando sempre sulla demo FishinoHomeAuto, proviamo ora a compilare lo sketch, ma senza caricarlo sulla scheda; quello che viene chiamato 'verifica sketch' nell'IDE di Arduino.
È sufficiente premere l'icona con l'occhio per avviare la compilazione; nell'AREA FINESTRE AUSILIARIE verrà attivata la finestra Messaggi in cui vedremo scorrere in sequenza tutti i file compilati, fino ad ottenere, in caso di successo, la consueta occupazione di memoria dello sketch (**Fig. 16**).
Nel menu *Impostazioni* esistono varie opzioni per modificare sia i messaggi che appaiono nella schermata proposta nella suddetta **Fig. 16**, sia il livello degli avvisi di compilazione, fino ai dettagli più piccoli; le impostazioni standard sono ottimali per non avere un output troppo prolisso e vanno bene nella maggior

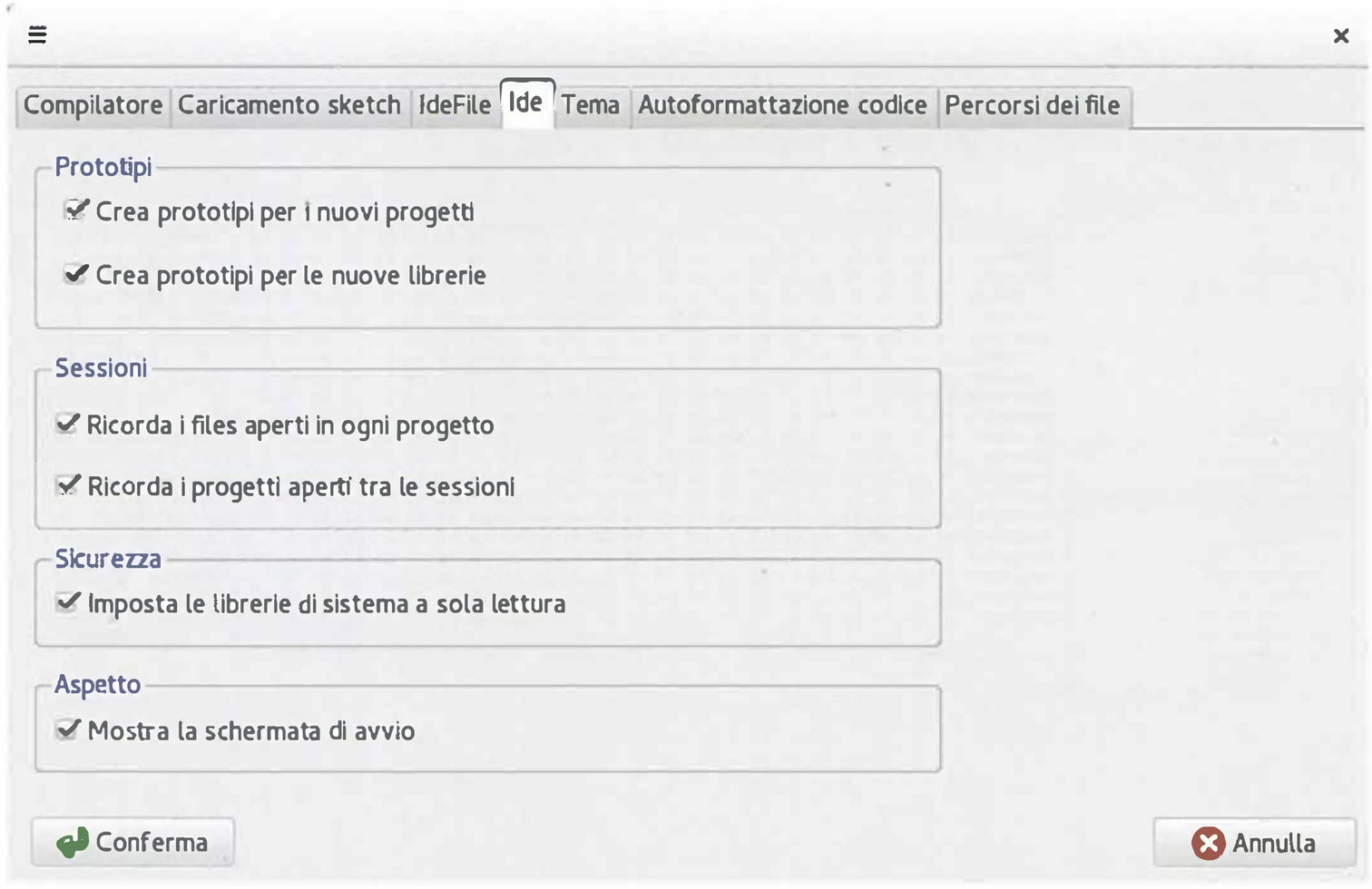


Fig. 15 - *Modifica del comportamento predefinito.*

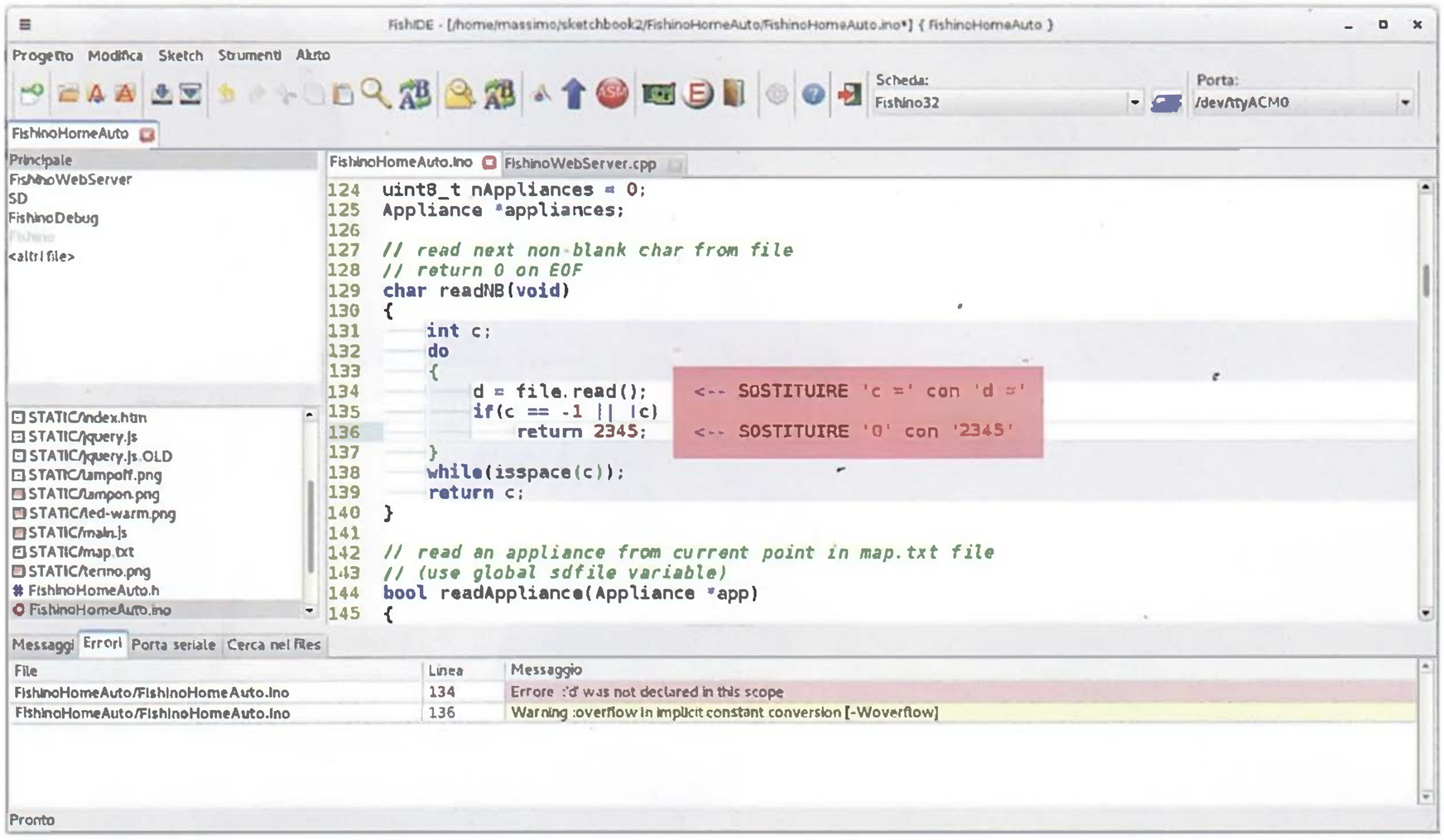


Fig. 17 *- Inserimento volontario di alcuni errori per verificare.*

parte dei casi. Potete comunque sperimentare con le varie opzioni; due di queste, il livello di avvisi e le ottimizzazioni, possono essere determinanti per scoprire errori subdoli nei programmi; anche qui abbiamo scelto per la prima un livello di avvisi molto elevato (decisamente più elevato di quello dell' IDE di Arduino, infatti in molte librerie scaricate vedrete apparire parecchi Warning), mentre le ottimizzazioni sono state impostate in modo da ridurre al minimo la dimensione dello sketch, come per l'IDE di Arduino. Solitamente, salvo per sketch molto piccoli, l'opzione senza ottimizzazione, che sarebbe perfetta per il debugging, non è applicabile poichè il file binario risulta troppo grosso e non risulta caricabile dal controller.

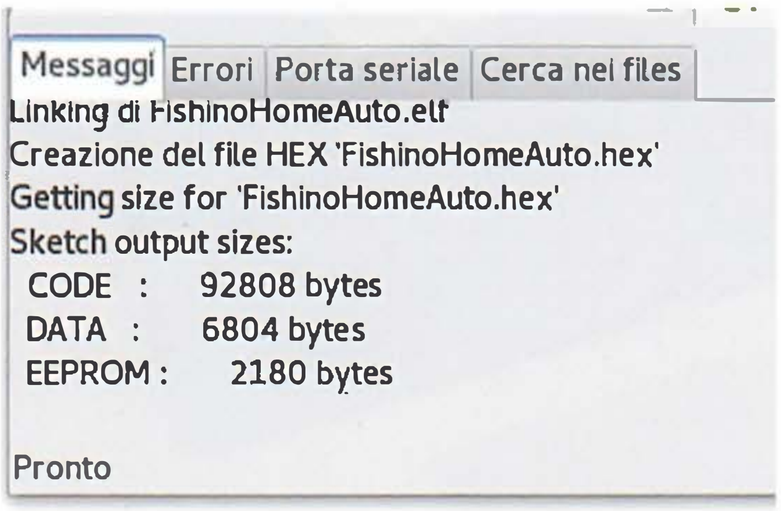


Fig. 16 *- Finestra Messaggi.*

Per caricare lo sketch è sufficiente ora cliccare sul pulsante con la freccia blu ⬆ e, se avrete precedentemente selezionato scheda e porta corretti, verrà avviato il caricamento dello sketch.

IN CASO DI ERRORI

Una delle cose più interessanti del FishIDE è l'immediatezza nell'identificazione degli errori di compilazione; per vedere come funziona questa caratteristica, riposizioniamoci sullo sketch *FishinoHomeAuto.ino* facendo clic sul relativo tab in alto ed inseriamo volutamente un errore di sintassi ed uno che normalmente "scapperebbe" al controllo se operassimo con un livello di avvisi più basso. Con riferimento alla **Fig. 17**, eseguiamo le modifiche riportate nel riquadro in rosso e rilanciamo la compilazione facendo clic sull'icona con il simbolo dell'occhio; una volta terminata, il pannello inferiore si posizionerà sulla finestra degli errori/avvisi, mostrandone l'elenco come in **Fig. 17**: in rosso saranno evidenziati gli errori e in giallino gli avvisi. Cliccando sulla linea dell'errore/avviso, il cursore si posizionerà nell'editor in corrispondenza dello stesso, permettendovi di correggere facilmente il codice.

In caso abbiate a che fare con uno sketch con più file o un errore in una libreria, cliccando sull'errore verrà aperto in automatico il file corrispondente, se non è quello già visualizzato, ed il cursore si posizionerà sempre nella giusta posizione. Ricorreggiamo ora i due punti e rilanciamo la compilazione, che, se abbiamo fatto le cose per bene, avviene senza intoppi.

CONCLUSIONI

Bene, con questo abbiamo concluso anche il discorso sulla compilazione e con esso la prima parte della descrizione del FishIDE, l'ambiente di sviluppo da noi creato per le piattaforme Fishino, che contrariamente all'IDE Arduino nasce predisposto per supportare la famiglia Fishino.
In questa prima puntata ci siamo soffermati sulle caratteristiche dell'IDE, evidenziando le differenze con quello di Arduino; inoltre abbiamo descritto installazione ed utilizzo dell'ambiente, che proseguiremo nella prossima puntata, descrivendo il preziosissimo editor di FishIDE. ■

Realizza un centralino telefonico VoIP con Raspberry Pi !

Set completo di tutto il necessario per realizzare un centralino VoIP con Raspberry Pi

cod. RASPBX

€ 88,00

Il kit comprende: board Raspberry Pi 3 Tipo B, alimentatore switching 5 VDC/3 A con uscita Micro-USB, contenitore trasparente per Raspberry Pi 3 Tipo B + E3, cavo FTP CAT5E con connettori RJ45, SD-Card da 16G con precaricata l'immagine di tutti i pacchetti open source necessari e localizzati per la lingua italiana.



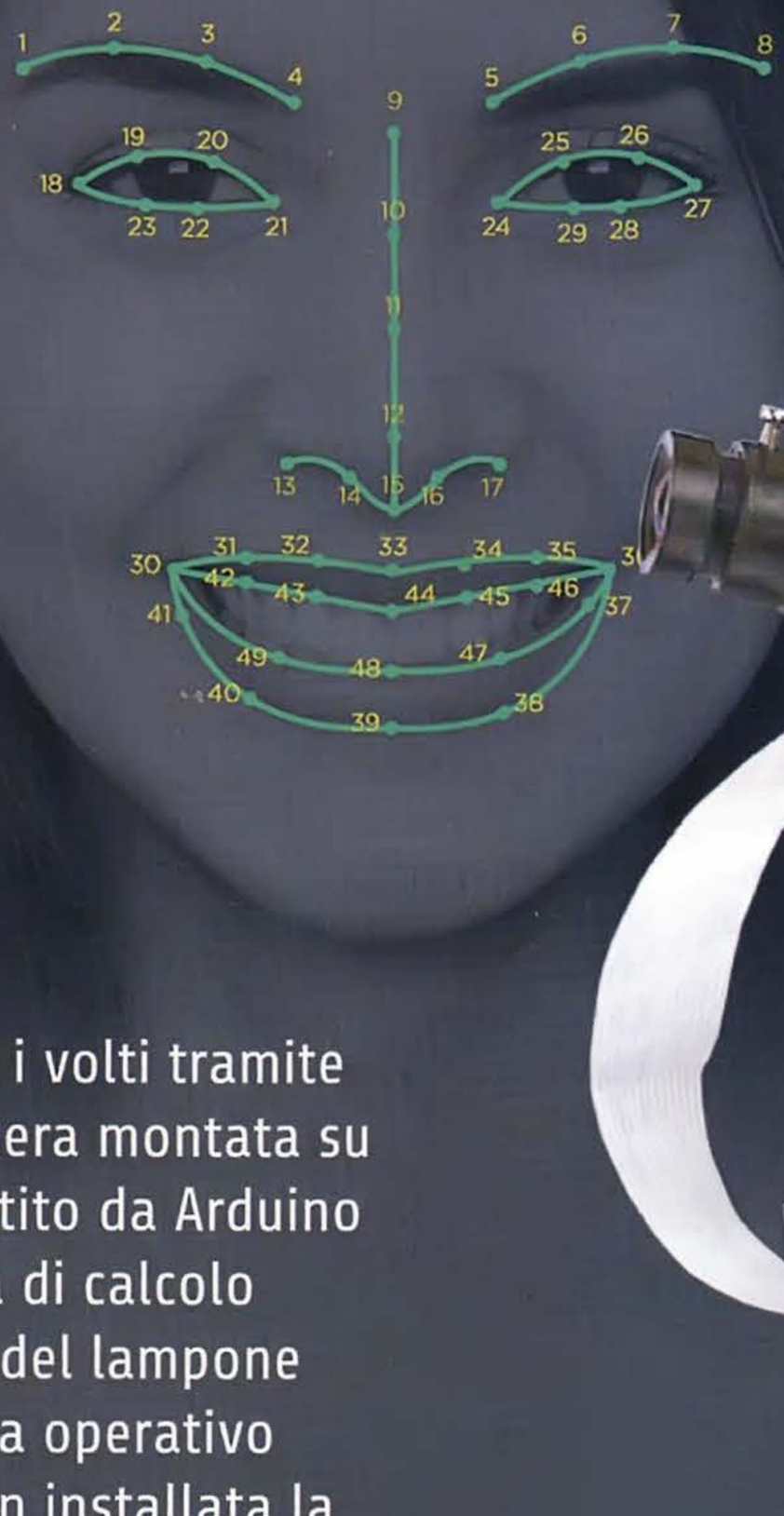
FUTURA ELETTRONICA®
Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

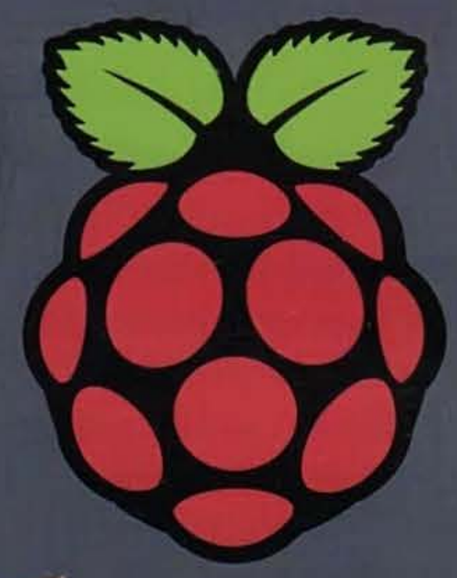
Prezzo IVA inclusa.

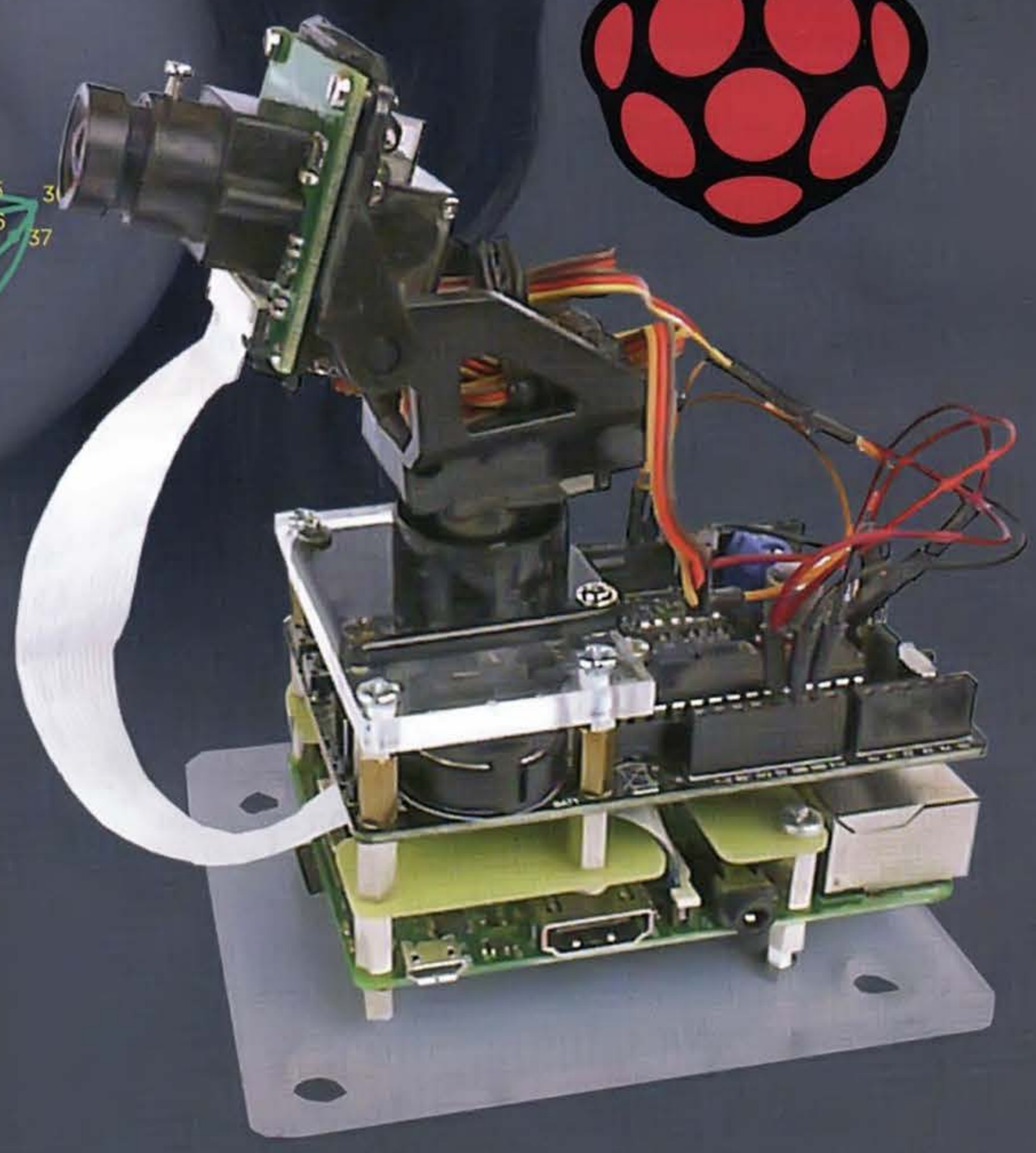
Caratteristiche tecniche e vendita on-line su: www.futurashop.it

Raspberry Pi

TRACKING DEL VISO CON RANDA







Analizziamo i volti tramite una telecamera montata su pan-tilt gestito da Arduino e la potenza di calcolo della board del lampone e del sistema operativo Raspbian con installata la libreria OpenCV.

dell'ing. DANIELE DENARO

Nell'articolo apparso nel fascicolo di settembre scorso, abbiamo prospettato la possibilità di gestire il tracking (inseguimento) del viso utilizzando una webcam collegata al PC, con Arduino come controller PID finalizzato a muovere il brandeggio (pan-tilt) che la sposta. Nell'applicazione proposta, Arduino era collegato anch'esso al PC tramite USB e quindi veniva gestito attraverso comandi inviati su porta seriale emulata. Il rilevamento del viso era realizzato da un programma scritto in Java, che utilizzava la libreria OpenCV e spediva le coordinate del centro del viso ad Arduino, il quale aveva il compito di gestire i servomotori

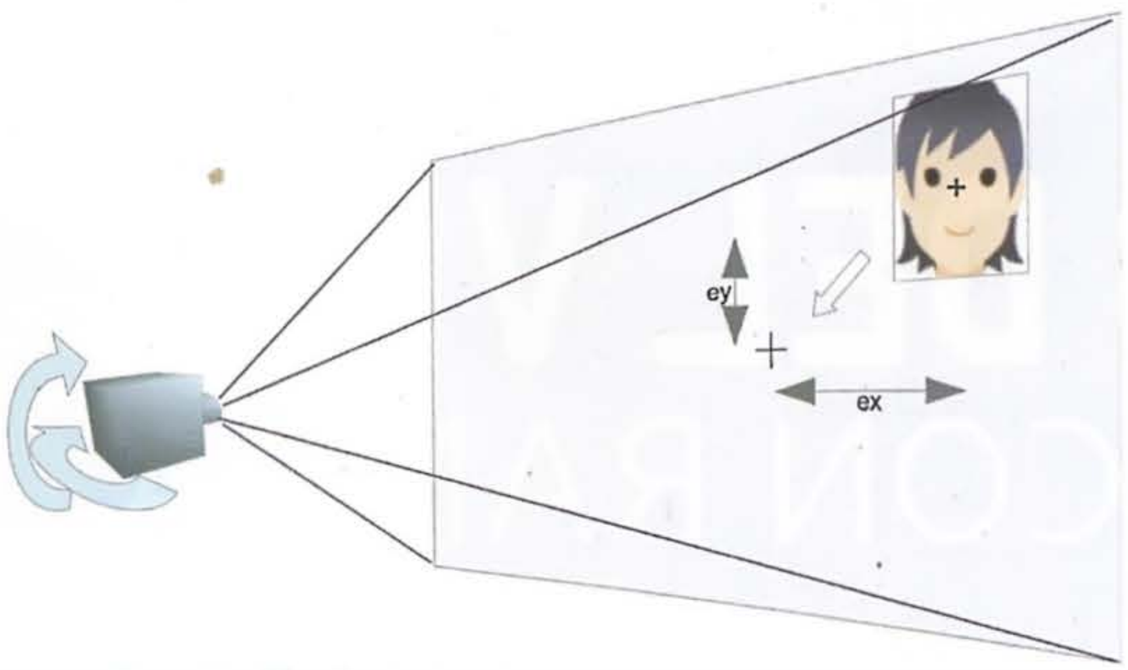


Fig. 1 - *Tracking del viso.*

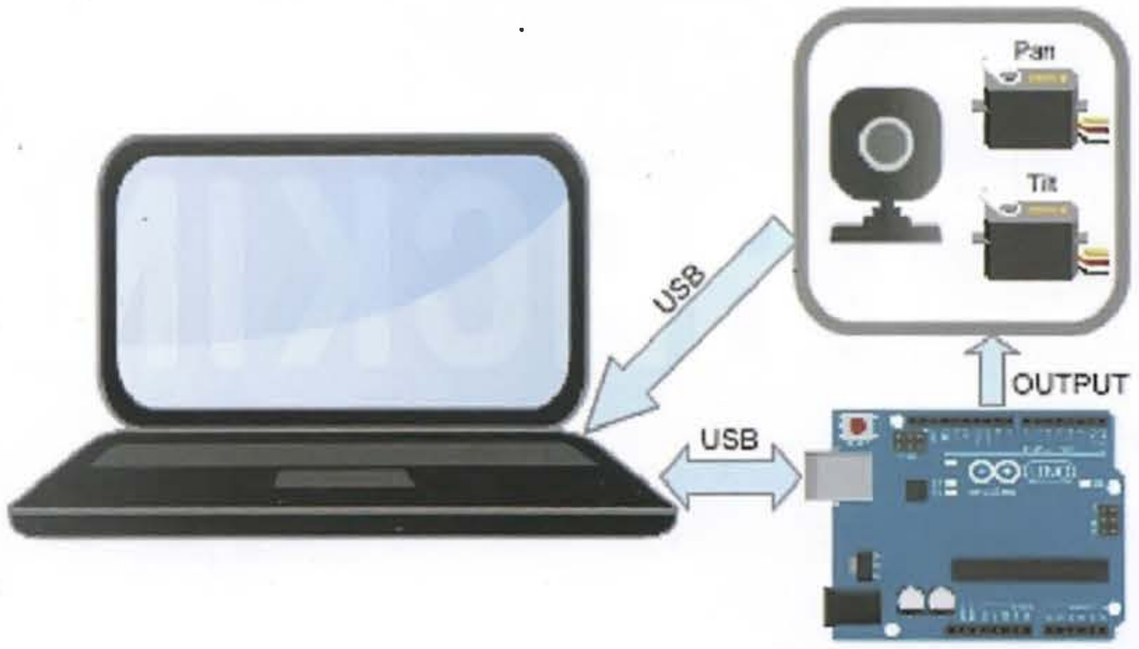


Fig. 2 - *Tracking con PC ed Arduino.*

del sistema pan-tilt (lo vedete nella **Fig. 1**) allo scopo di centrare la ripresa o di far seguire eventuali movimenti.
Dopo avervi presentato quel sistema (**Fig. 2**), vogliamo proporvi di fare un salto di qualità, procedendo al riconoscimento dei volti tramite la board Raspberry Pi. Infatti il PC può essere sostituito in maniera soddisfacente con una board Raspberry Pi e, per l'esattezza, con il modello Raspberry Pi 3, che è ben attrezzato per eseguire il compito.
Sebbene nulla vieti di utilizzare una webcam anche con Raspberry Pi 3, è preferibile provare la telecamera specializzata per la board (detta brevemente Raspicam); questa telecamera ha un'ottima risoluzione e fornisce un'immagine pulita e senza rumore. Abbiamo scelto la versione più economica da 5 Megapixel, reperibile su *www.futurashop.it* con il codice prodotto 2850-RPICAMERA-FOC; si tratta di una camera a colori per Raspberry Pi compatibile con quella originale e dispone di un sensore d'immagine a CCD da 1/4" Omnivision OV5647, oltre che di un'ottica fissa con apertura (F): 2,0, lunghezza focale di 6 mm e angolo di ripresa pari a 75,7°. È in grado di scattare foto con una risoluzione di 2.592x1.944 pixel e riprese video HD in formato 1.080p a 30 fps. Le dimensioni sono 32x32 mm e il peso è di soli 20 grammi. La telecamera rappresenta, insieme alla Raspberry Pi, l'elemento distintivo di questa nuova applicazione.

CARATTERISTICHE TECNICHE

- **Basato su RandA e Raspberry Pi 3**
- **Supporto software con libreria OpenCV**
- **Telecamera 5 megapixel per Raspberry Pi**
- **Pan-tilt con servo da 9 grammi**
- **Tenzione di alimentazione 5 Vcc**

L'ulteriore elemento di novità è il sistema pan-tilt, perché siamo passati da quello proposto nel fascicolo di settembre al mini pan-tilt della Futura Elettronica (lo trovate su *www.futurashop.it* con il codice 2846-PANTILT206), più adatto alle piccole dimensioni ed al peso della Raspicam. Inoltre questo sistema utilizza dei micro-servo da 9 grammi, molto meno esigenti in fatto di corrente assorbita rispetto a quelli utilizzati nella precedente applicazione.
La telecamera può essere posizionata sul supporto appositamente previsto, previo un piccolo adattamento meccanico; infatti occorre tagliare le due alette laterali e il piccolo gradino che eccedono le dimensioni del circuito stampato della telecamera, come mostrato nella **Fig. 3**.
Un elemento ingombrante può essere il flat-cable, utilizzato dalla telecamera, che serve per collegarla al connettore multifilare presente sulla scheda. Come vedremo più avanti, potrebbe essere necessario attorcigliare questo cavo per orientare correttamente la telecamera, ma, volendo, ciò può essere evitato ponendo la telecamera a testa in giù. Infatti il driver della Raspicam permette di

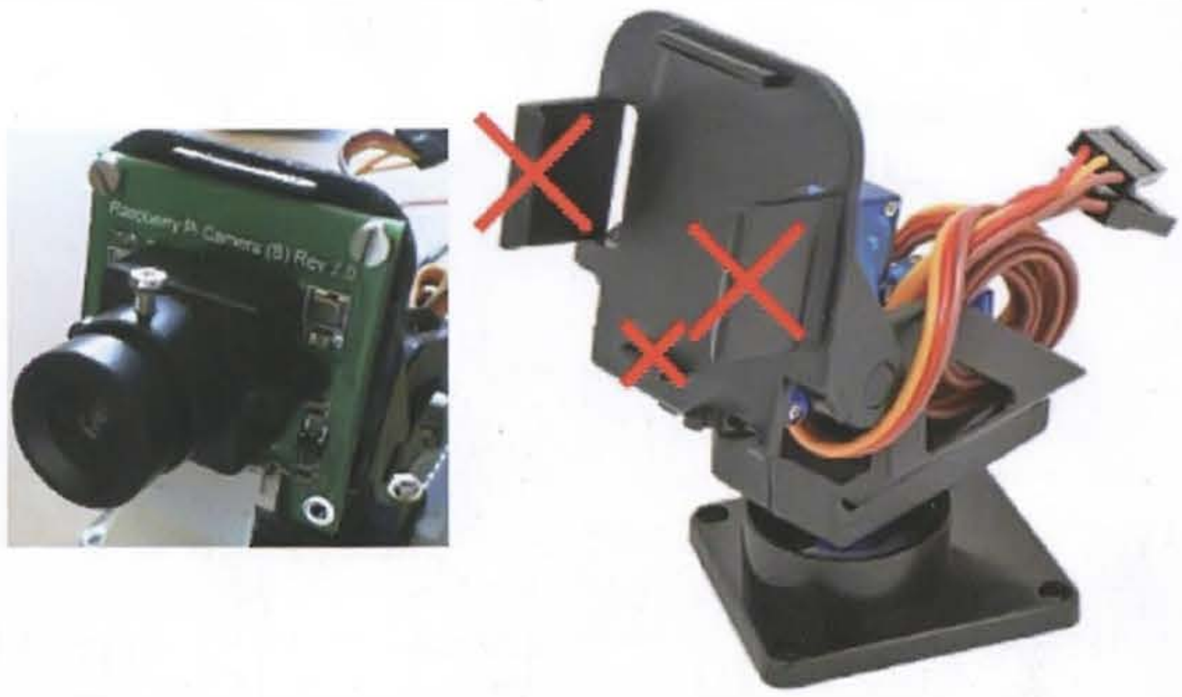


Fig. 3 - *Pan-Tilt con la Raspicam: le x rosse indicano cosa modificare per ospitarla.*

capovolgere automaticamente l'immagine, compensando il rovesciamento fisico.

UTILIZZIAMO RANDA

Pensando di utilizzare Raspberry Pi come processore dell'immagine, tramite OpenCV, il compito di Arduino può essere svolto egregiamente dalla nostra scheda RandA, perché con essa abbiamo i due sistemi in un unico complesso. Ricordiamo, allo scopo, che RandA incorpora l'hardware di Arduino Uno e un connettore per applicarla a una board Raspberry Pi con gli appositi distanziali, come visibile nella **Fig.** 4, che riporta il core di calcolo del sistema proposto; il processore Arduino di RandA dialoga con la Raspberry Pi per via seriale, come se fosse collegato tramite USB.
Dunque, in questo progetto non troverete più Arduino come scheda a sè stante, ma il suo hardware integrato in RandA, che provvederà alla gestione del brandeggio e Raspberry Pi che acquisirà ed elaborerà il segnale digitale fornito dalla Raspicam montata su di esso (**Fig. 5**).

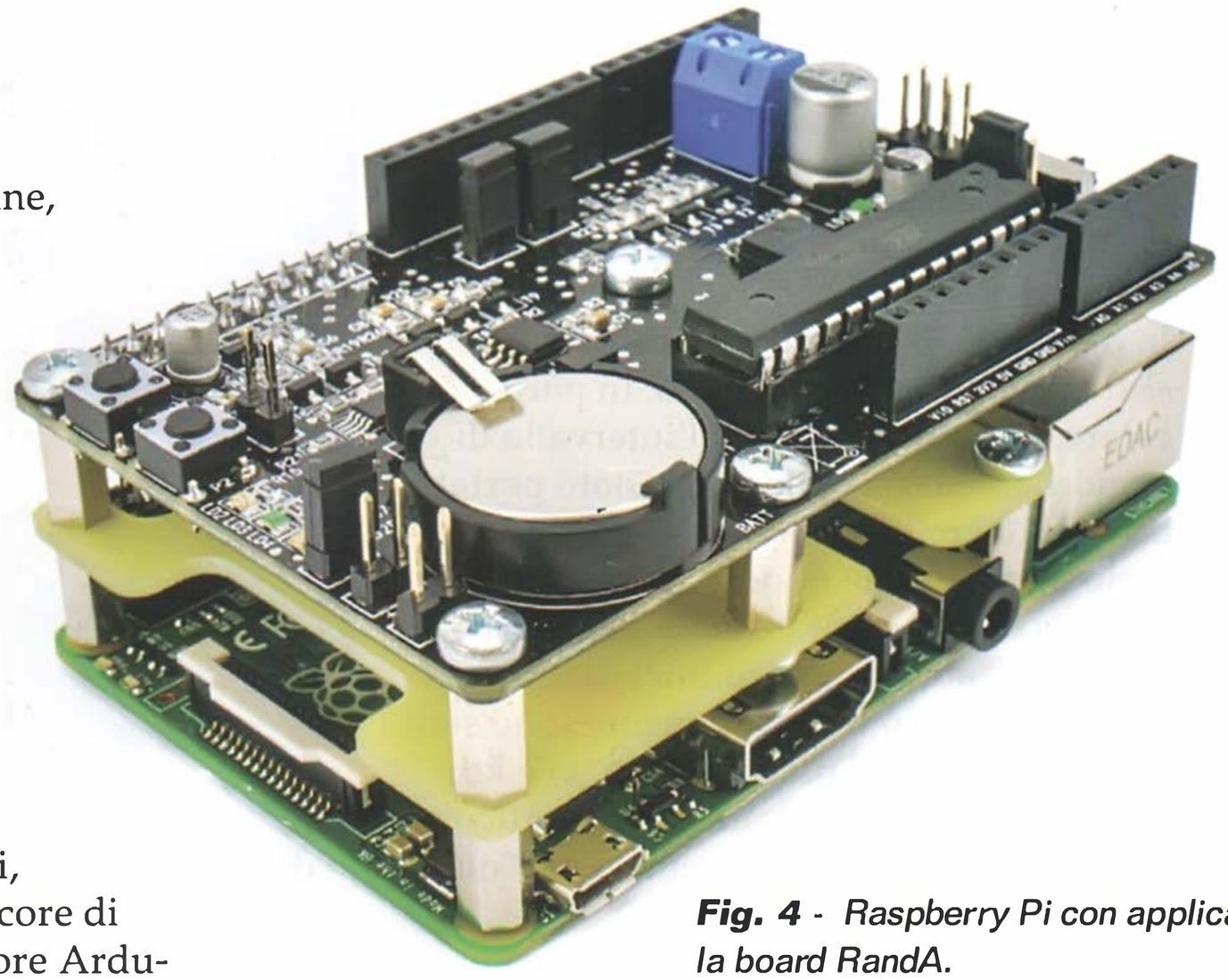


***Fig. 4** - Raspberry Pi con applicata la board RandA.*

OPENCV SU RASPBERRY PI

Avendo sviluppato il software per l'analisi delle immagini con Java, esso rimane identico sulla nuova piattaforma, rispetto alla prima applicazione (quella proposta in settembre), a patto di portare la libreria OpenCV sulla Raspberry Pi. Soprattutto è necessario portare sulla Raspberry Pi l'interfaccia nativa per la libreria Java di OpenCV. Infatti OpenCV realizza la versione Java tramite una libreria che interfaccia le routine compilate in C++; quindi c'è bisogno della libreria Java, che è unica per tutti i sistemi, ma anche della sua versione nativa (compilata) specifica.
Purtroppo nelle distribuzioni compilate previste per OpenCV non c'è quella per i processori ARM (e il processore della Raspberry Pi è un ARM...) pertanto è stato necessario compilare opportunamente i sorgenti di OpenCV, per poterlo utilizzare sulla board Raspberry Pi. Su Internet, alla pagina web *https://github.com/open-electronics/Tracking*, troverete queste librerie, che potrete utilizzare anche per altri progetti con OpenCV.
Ricordate che è possibile utilizzare OpenCV sulla Raspberry Pi, oltre che con Java, con C++ e Python.
Per utilizzare OpenCV sulla Raspberry Pi, oltre al downloading delle librerie specifiche, occorre installare alcune altre librerie dal repository di Raspbian; librerie che sono richieste dal software OpenCV. Sul sito è presente uno script che verifica la presenza di questi moduli e ne mostra anche un elenco.
A questo punto possiamo utilizzare il software descritto nel precedente articolo su questa nuova piattaforma; infatti lo sketch su Arduino rimane identico dialogando con Raspberry Pi tramite la porta seriale "/dev/ttySA".
La libreria Java per la gestione della seriale è già presente nel software per RandA in quanto utilizzata per altri scopi (come per esempio per l'IDE locale di Arduino); ma per comodità è stata inserita anche nel kit del software per il tracking.
Naturalmente bisogna configurare opportuna-

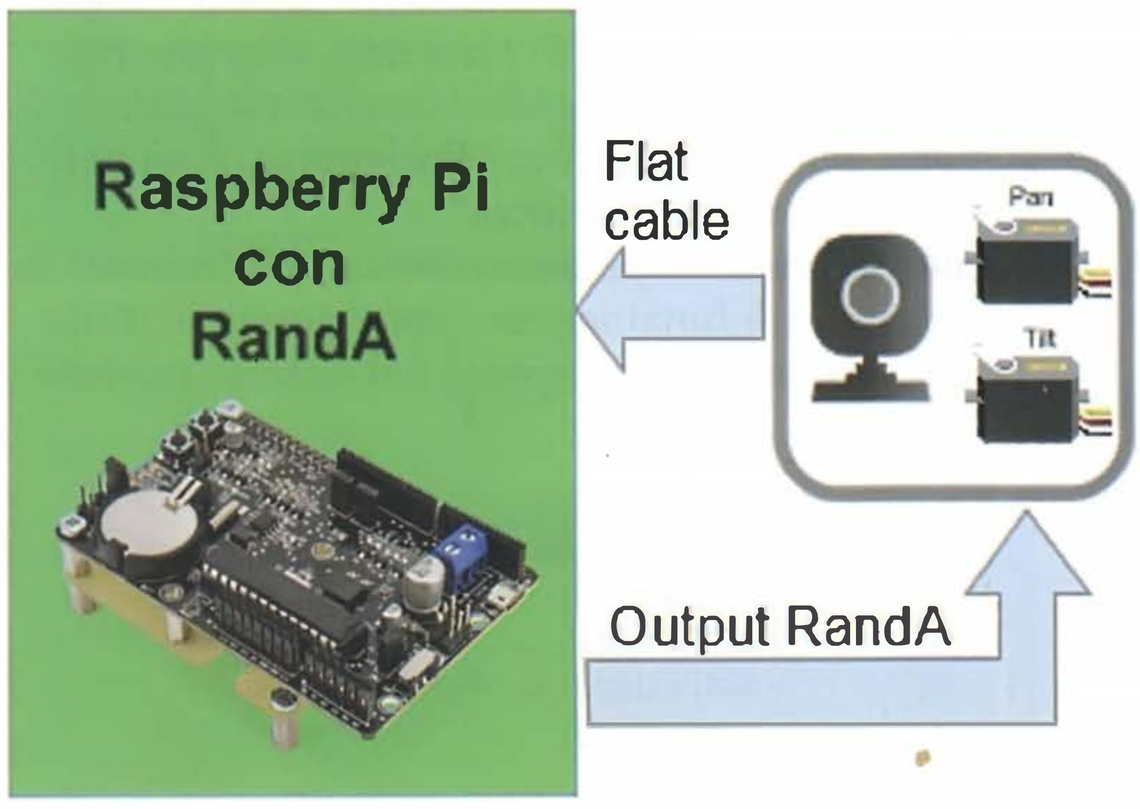


***Fig. 5** - Tracking con Raspberry Pi e RandA.*

mente il file testuale "tracking.proprieties", dove sono descritti i vari puntamenti e definizioni.
In ogni caso lo trovate sul nostro sito *www.elettronicain.it* già aggiornato. In particolare, trovate modificato l'intervallo di aggiornamento, che adesso è stato portato da 40 a 100 millisecondi. Ciò è stato necessario a causa delle minori "performance" del processore montato sulla Raspberry Pi rispetto ad un classico Intel di ultima generazione. Conseguentemente sono stati aggiornati anche i parametri kp, ki e kd; infatti, come detto nel precedente articolo, questi, per comodità, sono comprensivi dell'intervallo di aggiornamento, cioè:

kp=Kp ma ki=Ki*Tc e kd=Kd/Tc

Per questa ragione, aumentando Tc bisognerebbe ridurre ki ed aumentare kd; inoltre bisogna tener conto anche della ridotta inerzia del nuovo sistema meccanico, molto più leggero del precedente.

LA GESTIONE DELLA RASPICAM

La telecamera per la Raspberry Pi è pensata per essere collegata al connettore multipolare sulla scheda, tramite un "flat cable" (si tratta del connettore CSI); nel caso di utilizzo con RandA montata sulla Raspberry Pi (come è nella nostra applicazione), bisogna far passare il cavo piatto attraverso l'intaglio presente sulla schedina in vetronite nuda che fa da sostegno meccanico di RandA (riferitevi alla **Fig. 6**).
La telecamera, una volta montata, va abilitata utilizzando il menu del comando "sudo raspi-config". In particolare, utilizzando la voce "Interfacing Options" e poi "Pi Camera". Oppure, più semplicemente, utilizzando l'interfaccia grafica ed il menu generale alla voce "Preferenze/Raspberry Pi Configuration/Interfaces".
A questo punto, dopo aver effettuato il "reboot" la Raspicam sarà funzionante. Ora è possibile utilizzare il software già compreso nel sistema operativo Raspbian, ovvero le seguenti utility:

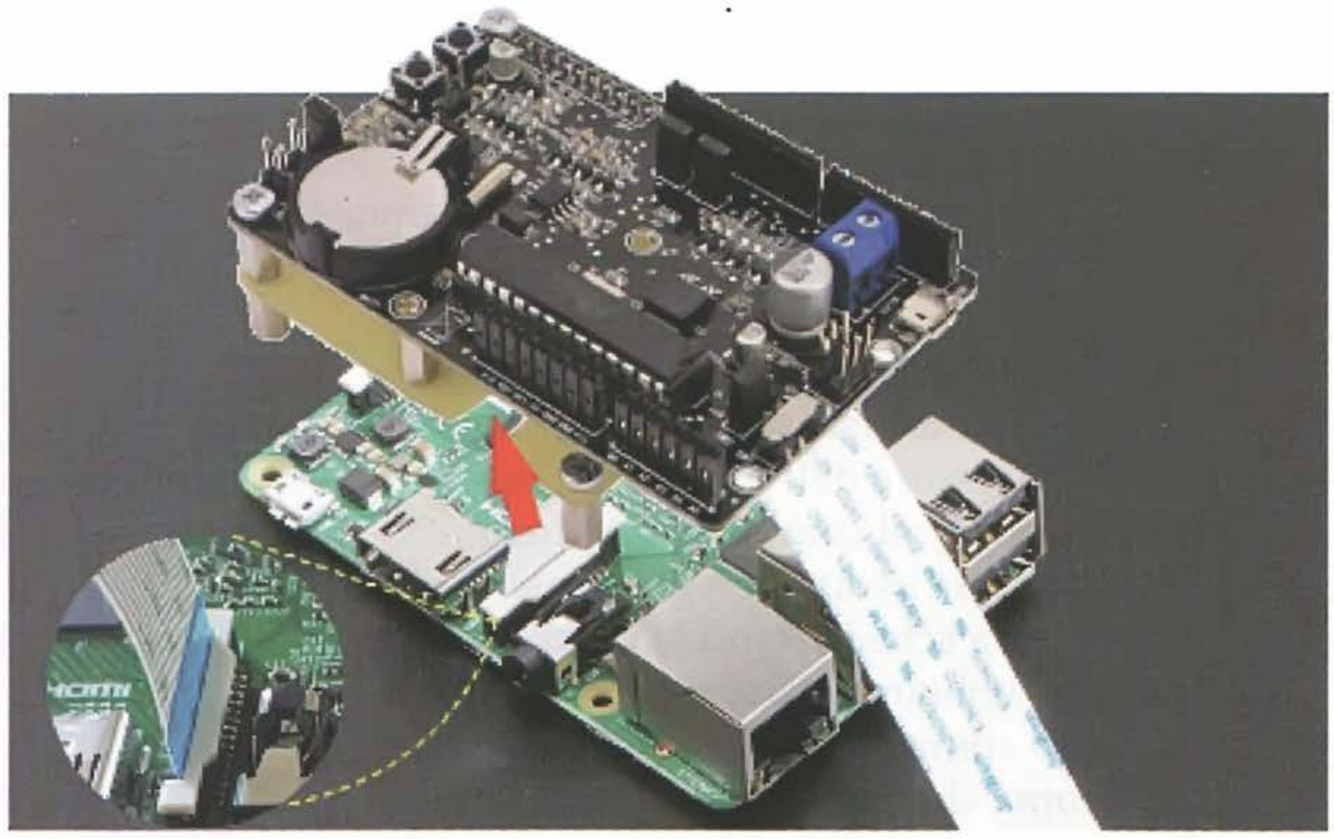


***Fig. 6** - Montaggio e cablaggio della Raspicam.*

- *raspistill* per scattare foto da memorizzare in formato jpg o "raw" anche in time-lapse;
- *raspivid* per salvare file video.

Per visualizzare ciò che è catturato dalla telecamera è possibile utilizzare semplici script Python mediante la sua libreria ("import picamera") oppure, più semplicemente, installando il potente programma multimediale **VLC media player** (per informazioni fate riferimento al sito *www.videolan.org*). Una volta installato (mediante il solito comando "sudo apt-get install vlc"), il software VLC va lanciato con un comando del tipo:

vlc v4l2:///dev/video0:width=320:height=240

Il comando indica a VLC che la periferica (vale a dire la telecamera per Raspberry Pi) corrisponde a "/dev/video0", che deve essere aperta con il drive "v4l2" e che la finestra corrispondente deve avere dimensioni pari a 320x240.

IL DRIVER "V4L2"

Le utility software "raspistill" e "raspivid" interfacciano direttamente la telecamera al sistema operativo Raspbian e quindi non fanno uso delle API (Application Program Interface) standard. Per

```
V=new VideoCapture();       // Istanza della classe della periferica di acquisizione (webcam)
if (!V.open(0)) {end("No PIcamera!");} // Apre la webcam (la prima : indice 0)
V.set(Videoio.CAP_PROP_FRAME_WIDTH, (double)dimx); //con dimx=320
V.set(Videoio.CAP_PROP_FRAME_HEIGHT, (double)dimy);//con dimy=240
V.set(Videoio.CAP_PROP_FPS, 60);
```

***Fig. 7** - Configurazione di Raspicam tramite OpenCV .*

poter utilizzare o implementare software multimediale per la Raspicam, sarebbe comunque opportuno poter contare su un driver ufficiale e standard come V4L (Video for Linux). Fortunatamente una versione aggiornata di questo driver "v4l2" è stata inclusa nel kernel Linux per Raspberry Pi già dalla fine del 2013. Il relativo modulo va però reso operativo caricandolo con il comando:

sudo modprobe bcm2835-v4l2

Se si vuole evitare di inserire il comando ad ogni riaccensione, questo può essere inserito, nel file di start-up "/etc/rc.local"; oppure come nome di modulo (cioè una riga con: bcm2835-v4l2) nel file "/etc/modules". In ambedue i casi occorrono i privilegi di "root". In alternativa potete inserire il comando nel file "/home/pi/pistartup.sh" che non ha bisogno dei privilegi e che viene fatto partire alla fine dello start-up. Il driver può essere configurato da console con il comando:

v4l2-ctl ...

I parametri sono numerosi, ma in Rete trovate un riferimento completo ad essi. Un parametro importante è:

-d ***nome-della-periferica***

Per esempio il comando:

v4l2-ctl -d /dev/video0 –list-ctrls

elenca tutti i comandi eseguibili sul device "/dev/video0" ed il loro valore attuale. Se non c'è il parametro "*-d nome-della-periferica*", la periferica considerata è proprio la prima, cioè "/dev/video0".
Il comando:

v4l2-ctl -c vertical_flip=1

ribalta verticalmente (sotto-sopra) l'immagine acquisita dalla periferica "/dev/video0" (il parametro *-c nome_proprietà=...* serve a configurare la telecamera).
La configurazione può, ovviamente, essere fatta da programma come si è visto nel lancio del programma VLC. Useremo proprio questa modalità nel tracking con Java+OpenCv ; infatti, per ridurre il carico di lavoro della routine di "face-detecting" è opportuno portare la finestra di acquisizione alle dimensioni di 320x240 pixel. Inoltre per aumentare la fluidità della visualizzazione è opportuno utilizzare una frequenza di "frame" a 60 fps ("frames per second").
Come si vede nell'estratto del software mostrato in **Fig. 7**, la configurazione dell'acquisizione è modificata tramite il metodo "set()" della classe "VideoCapture" (di OpenCV-Java), utilizzando codici enumerati nella classe "Videoio".

LO SKETCH PER ARDUINO

Come si è già detto, lo sketch su Arduino non cambia di una virgola, essendo RandA una Arduino collegata alla seriale "/dev/ttySA". Quello che cambieranno sono i valori letti dal programma Java sul file di configurazione "Tracking.proprieties" e passati come comandi di "setting" del PID implementato dallo sketch, tramite la seriale. I valori sono stati modificati per adeguare il PID alle diverse condizioni del sistema.
Nel file presente nell'archivio dell'applicazione completa troverete dei valori che sembrano dare un soddisfacente comportamento.

RIASSUMENDO

Bene, prima di procedere è il caso di riepilogare i dettagli salienti della nostra applicazione; partiamo dal software, che, corredato dei sorgenti, va scaricato dalla pagina web:

https://github.com/open-electronics/Tracking

Qui sono contenuti diversi archivi compressi e in particolare vi potete trovare l'archivio compresso contenente l'applicazione per il sistema Windows ("TrackingWin.zip"), di cui si è parlato al precedente articolo, e l'archivio relativo all'ambiente Raspberry Pi : "TrackingRasp.zip".
Quest'ultimo contiene il seguente materiale software:

- la cartella *libcvarm* contenente la libreria nativa necessaria al funzionamento della libreria Java per OpenCV (*libopencv_java320.so*);
- la cartella *libserarm* contenente la libreria nativa associata alla libreria Java *SerialLib.jar* (che è una estensione della *rxtxComm*);
- la cartella *TrackPID* contenente lo scketch Arduino (che è uguale a quello per Windows);
- la cartella con i file sorgenti Java;
- l'applicazione vera e propria, che è *Tracking.jar*;
- lo script di lancio (esecuzione) *Tracking.sh*;
- il file testuale di configurazione *Tracking.proprieties* e il file del modello "addestrato" per il riconoscimento del volto, ossia *haarcascade_frontalface_alt.xml*;

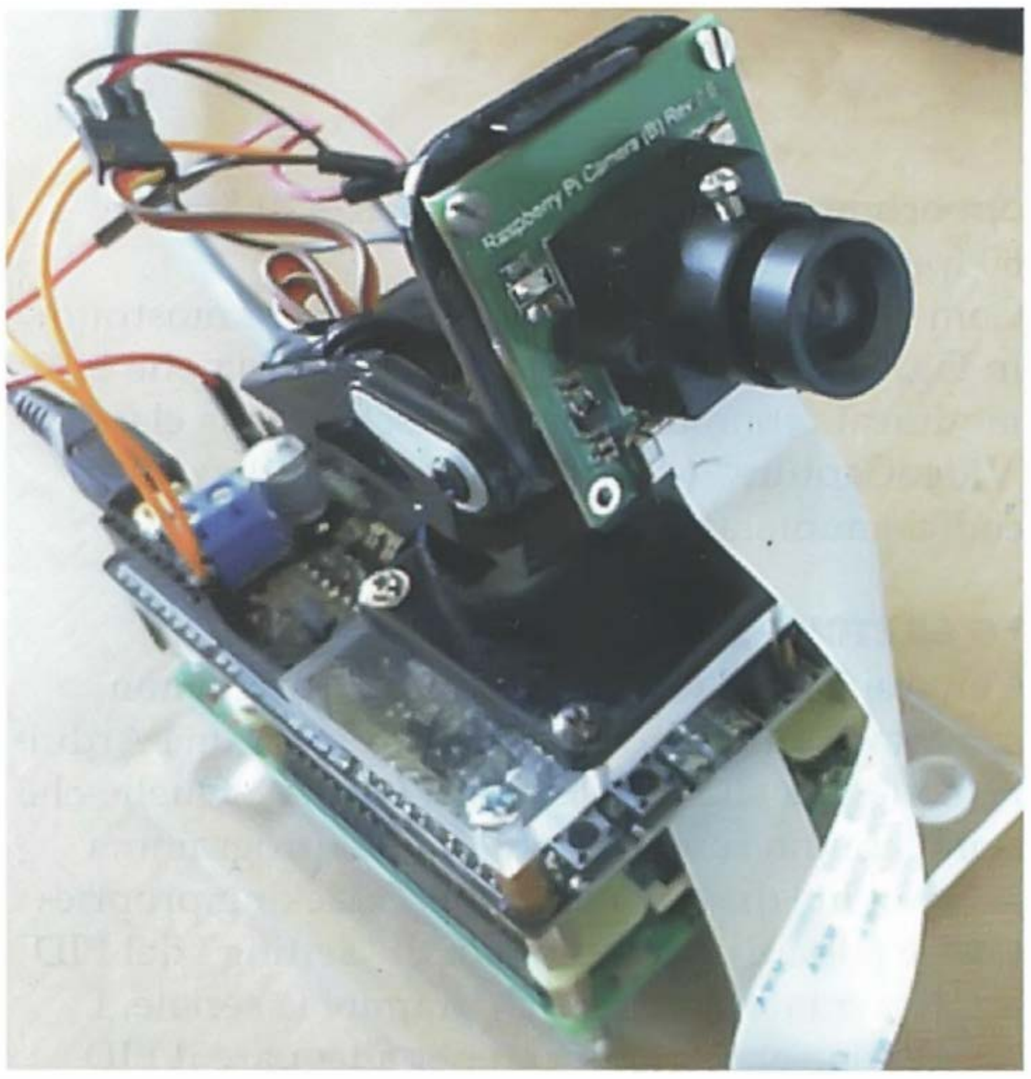


***Fig. 8** - Il sistema di tracking con RandA, al completo.*

L'archivio dev'essere decompresso in una cartella da cui lanciare lo script di avvio *Tracking.sh*.
Nel sito sono poi presenti anche altri file di possibile interesse come:

- un archivio di modelli addestrati per altre configurazioni oltre al viso umano;
- un archivio separato per la libreria Java con la controparte nativa per OpenCV e per la gestione della seriale (per ambiente Windows e per ambiente Raspberry Pi), da impiegare nel caso si vogliano sviluppare programmi Java con queste caratteristiche, senza dover scaricare l'applicazione Tracking;
- un archivio completo del risultato della compilazione di OpenCV su Raspberry Pi 3;
- uno script per verificare la presenza o meno di "package" necessari al funzionamento di OpenCV su Raspberry Pi.

Ora va fatta una precisazione: fin qui abbiamo scritto che il programma Java per il tracking è rimasto immutato nel passaggio dall'ambiente Windows a quello Raspberry Pi; in realtà non è del tutto vero, perché in questa versione abbiamo inseguito un'ottimizzazione per ovviare al minore rendimento della CPU ARM. Poiché su Raspberry Pi 3 è montata una CPU ARM quad-core, è stata creata un'ulteriore classe "ShowFrame.java" che realizza un thread separato ed ha il solo compito di mostrare l'immagine della telecamera ad intervalli regolari (30 millisecondi), con la speranza che la "Java Virtual Machine" utilizzi più "core" della CPU. Nelle prove siamo passati da un tempo di "FaceDetecting" di circa **40** millisecondi con CPU Pentium I7, a poco più di **100** millisecondi con ARM quad-core di Raspberry Pi 3, sebbene nel caso di I7 la finestra di acquisizione fosse più larga (640x480).

CONCLUSIONI

Per chi volesse approfondire il meccanismo di tracciamento del viso mediante la classe "CascadeClassifier" di OpenCV o il controller PID realizzato con Arduino, rimandiamo alla lettura del precedente articolo che li descrive più in dettaglio.
Con questi due articoli si è voluto suggerire la sperimentazione dei meccanismi di Intelligenza Artificiale con la grande libreria OpenCV, utilizzando nel contempo Arduino per il compito in cui riesce meglio: il controllo in tempo reale. Scegliendo Java come ambiente di sviluppo si è cercato di uniformare il codice per le diverse architetture Windows e Raspberry Pi. L'utilizzo di RandA ha poi permesso di compattare il sistema, nel caso di utilizzo dell'architettura Raspberry Pi 3.
La board del "lampone" ha svolto degnamente il compito affidatole, anche se con qualche affanno. Lasciamo a chi di voi se la sente, il tentativo di ottimizzare ulteriormente il software.
Ricordiamo, infine, che sul nostro sito è presente la versione compilata della libreria OpenCV per Raspberry Pi, per renderla disponibile anche a chi intenda utilizzare Raspberry Pi per altri compiti di Intelligenza Artificiale. ■

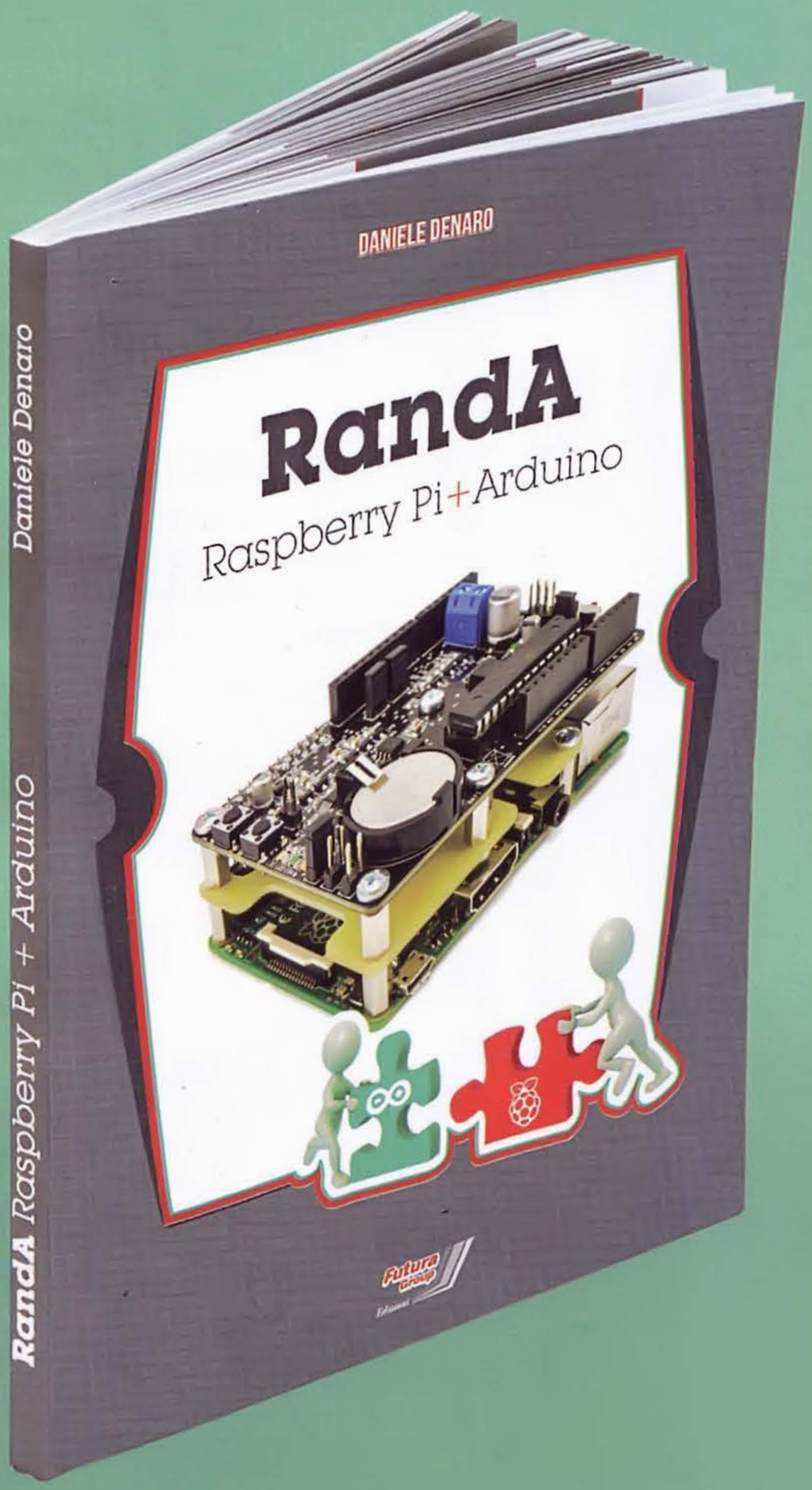
per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono di facile reperibilità, mentre il materiale elencato di seguito può essere acquistato presso Futura Elettronica. RANDA, la scheda di sviluppo Raspberry Pi e Arduino (cod. RANDA) è disponibile a Euro 39,00, la telecamera da 5 Megapixel per Raspberry Pi (cod. RPICAMERA-FOC) è in vendita a Euro 29,00, il servo analogico 1,2 Kg/cm (cod. SERVO206) al prezzo di Euro 7,50, il Pan-Tilt per servo 206 (cod. PANTILT206) costa Euro 4,00 e la board Raspberry Pi 3 Tipo B (cod. 8968660RS) è disponibile a Euro 44,00. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 • Fax: 0331-792287 - www.futurashop.it

GUIDA COMPLETA a RandA

La piattaforma che unisce Arduino e Raspberry Pi



€ 15,00

Cod. RANDABOOK

Questo libro vi permetterà di scoprire le potenzialità della scheda RandA, la board che unisce la semplicità d'uso di Arduino con la potenza del microcomputer Raspberry Pi. Vedrete come è semplice tramutare in realtà le vostre idee, anche le più complesse. Imparerete a gestire le periferiche Arduino da Raspberry Pi e viceversa, scoprirete come realizzare un web server e un sistema di allarme.

128 PAGINE

13 CAPITOLI

Acquistalo su *www.futurashop.it*

Futura Group srl - Via Adige, 11 - 21013 Gallarate (VA)
Tel: 0331-799775

Futura Group Edizioni

Ora i componenti non hanno più segreti!

Nuovi dispositivi professionali per misure e analisi precise.



Analizzatore di componenti *DCA55*

€ 69,00
cod. DCA55

Consente di analizzare una vasta gamma di componenti attivi, identificando la pinout e fornendo informazioni relative ai principali parametri.





€ 65,00
cod. ZEN50

Analizzatore di diodi *ZEN50*

Analizzatore multifunzione. Misura le caratteristiche di conduzione di svariati tipi di diodi.



€ 135,00
cod. ESR70

Misuratore di Capacità e ESR

Misura la capacità e l'ESR (Resistenza Equivalente Serie) di un condensatore con un valore compreso tra 1 uF e 22.000 uF. Dispone di avvisi acustici (per vari livelli ESR) e di una nuova gamma di misurazione (fino a 40 ohm).

Prezzi IVA inclusa



Analizzatore di semiconduttori *DCA75*

Analizza svariati tipi di semiconduttori. Può essere interfacciato al PC e, grazie al software in dotazione, fornisce un'analisi dettagliata del componente.

€ 165,00
cod. DCA75



Misuratore di induttanza, capacità, resistenza e impedenza *LCR45*

Strumento multifunzione. Misura induttanza, resistenza e impedenza arbitraria. Riconosce automaticamente il tipo di componente, e fornisce informazioni relative ai principali parametri.

€ 139,00
cod. LCR45

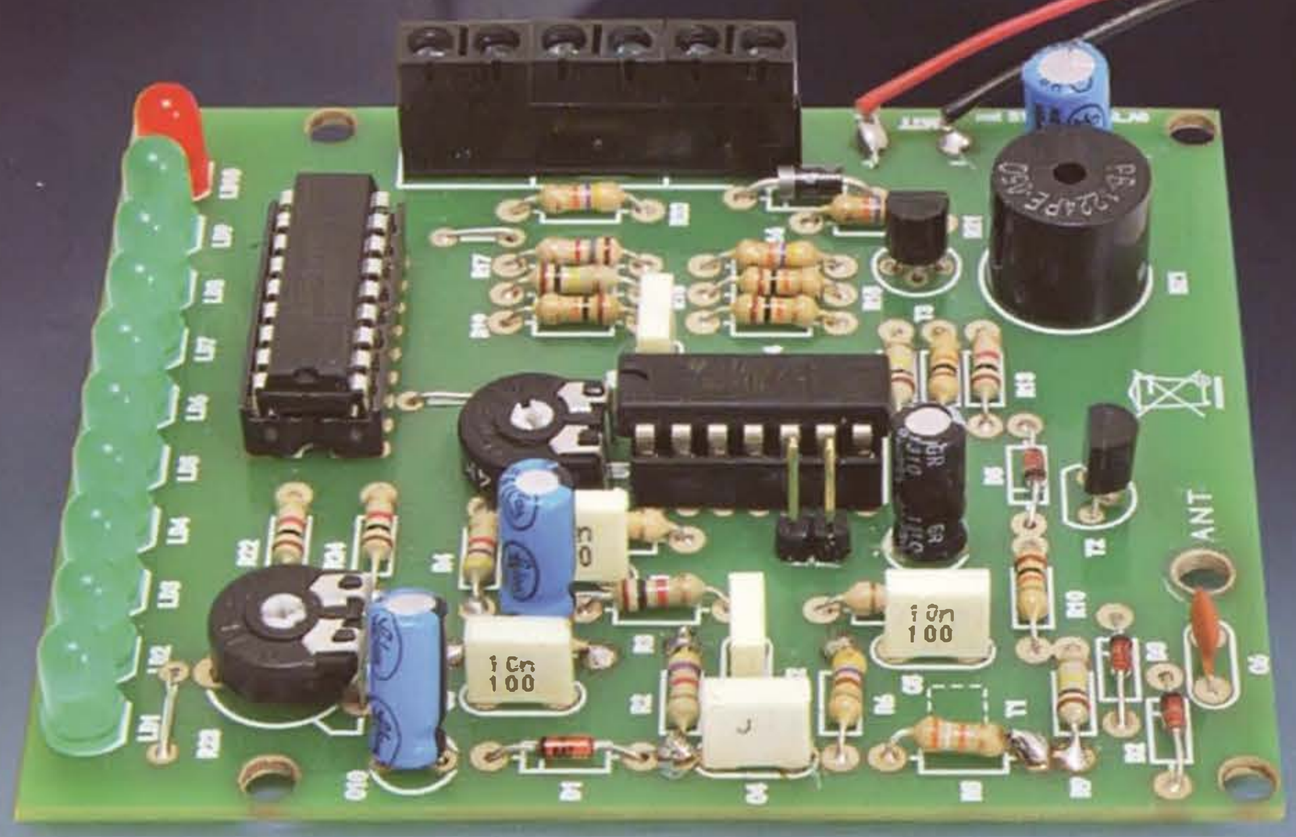
Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

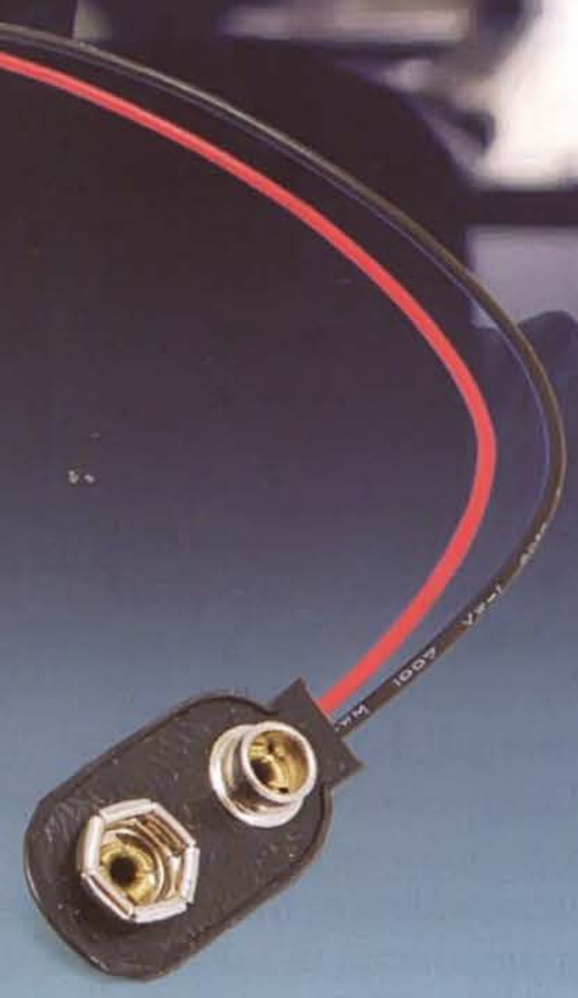
Prezzi IVA inclusa.

www.futurashop.it

RILEVATORE DI MICROSPIE

Ricevitore a banda larga e amplificazione diretta in grado di rilevare segnali RF di ogni genere e con qualsiasi modulazione, idoneo al rilevamento delle temutissime "cimici" negli ambienti dove la riservatezza è indispensabile.





di FRANCESCO DONI

Diciamolo pure: almeno una volta nella vita ci siamo sentiti sotto controllo ó abbiamo temuto di essere spiati e di avere in casa una microspia, ad esempio collegata al telefono o sotto al tavolino, come nei migliori film di spionaggio. In quei casi ci avrebbe fatto piacere avere tra le mani uno "spy detector", vale a dire un particolare radioricevitore in grado di dirci se nei dintorni c'è qualche trasmettitore attivo: trovare con esso un segnale radio è già un indizio; localizzarne la direzione, grazie a un indicatore del livello del segnale che sale puntando l'antenna da una parte, qualcosa di più. Poi, magari si tratta di un cellulare o di un sensore wireless dell'allarme di casa, ma è meglio averne la certezza. Per chi ha bisogno di ricercare eventuali radiospie o semplicemente microtrasmettitori utilizzati, ad esempio, dagli esaminandi nelle aule d'esame per collegarsi con amici "suggeritori" al di là del muro, abbiamo realizzato, nel "lontano" 2001, un *rilevatore di microspie*, che sostanzialmente era un ricevitore a larga banda in grado di captare e segnalare la presenza di onde elettromagnetiche emesse da

[Schema ELETTRICO]

un trasmettitore operante nelle immediate vicinanze.
Oggi vogliamo tornare sull'argomento, riproponendo una versione arricchita di quel progetto, dotata di un visualizzatore a barra di LED in aggiunta allo strumentino analogico (V-meter), che ormai è sempre più difficile da reperire in commercio, visto che i display vanno per la maggiore. Il visualizzatore è del tipo a barra di LED che si allunga e si accorcia. Lo strumento, che sia il vu-meter analogico o l'indicatore a LED servirà, una volta rilevata la presenza di un segnale, a cercarne l'origine, operazione che si compie semplicemente avvicinandosi o spostandosi nella direzione cui corrisponde il più alto livello di segnale captato.

SCHEMA ELETTRICO

Analizziamo dunque lo schema elettrico del nostro nuovo rilevatore: il circuito, pur essendo relativamente semplice, è costituito da più blocchi funzionali evidenziati nello schema a blocchi proposto nella **Fig. 1**.
Il primo stadio serve ad amplificare il debole segnale captato dall'antenna ricevente; essendo

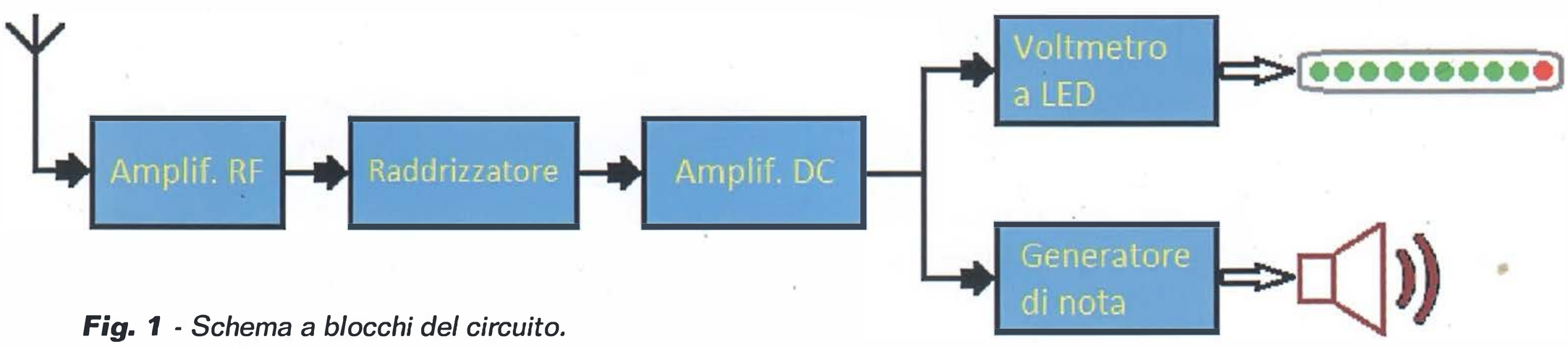
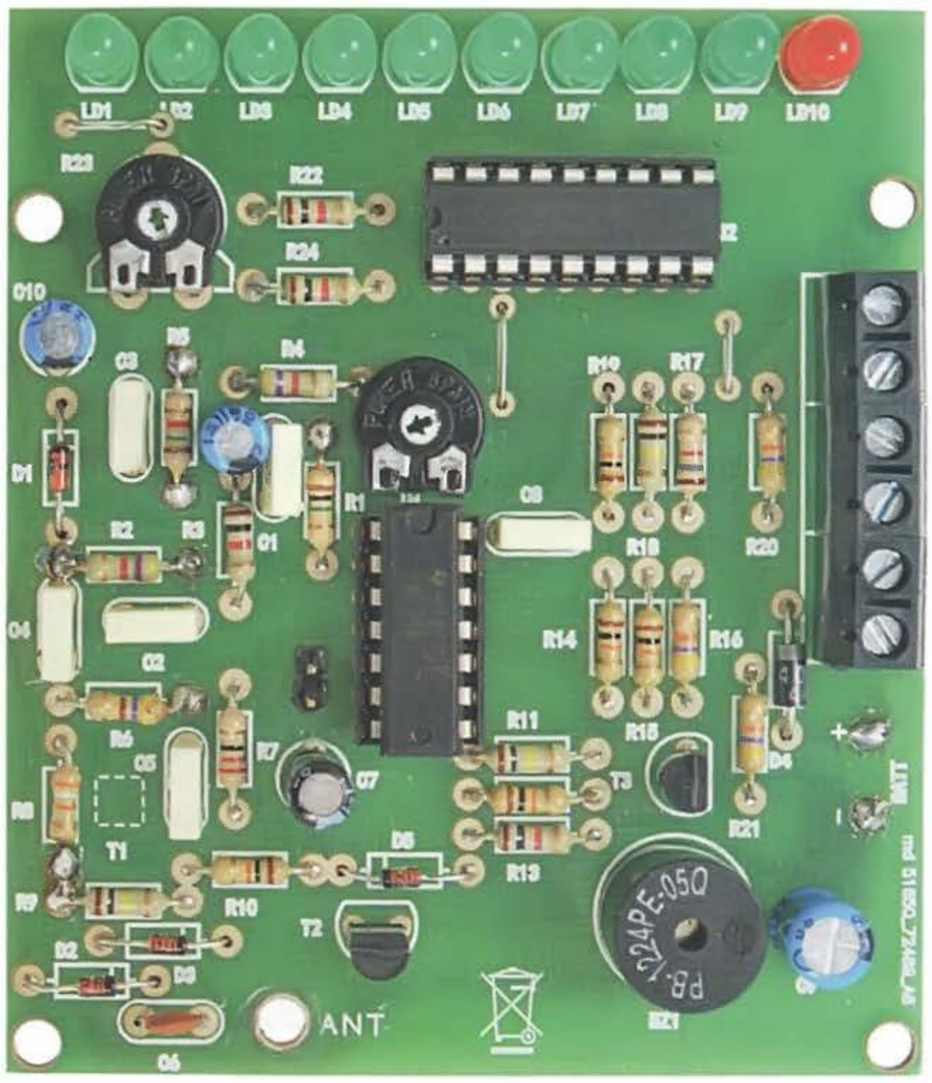


***Fig. 1** - Schema a blocchi del circuito.*

[piano di **MONTAGGIO**]



Elenco Componenti:

R1: 1 Mohm
R2: 4,7 kohm
R3: 1 kohm
R4: 4,7 kohm
R5: 1,5 kohm
R6: 470 ohm
R7: 22 ohm
R8: 39 kohm
R9: 470 kohm
R10: 10 kohm
R11: 100 kohm
R12: 10 kohm
R13: 1 kohm
R14: 10 kohm
R15: 10 kohm
R16: 47 kohm
R17: 1,8 kohm
R18: 100 kohm
R19: 10 kohm
R20: 47 kohm
R21: 47 kohm
R22: 1 kohm
R23: Trimmer 4,7 kohm MO
R24: 1 kohm
R25: Trimmer 4,7 Mohm MO
C1: 10 nF 100 VL poliestere
C2: 10 nF 100 VL poliestere
C3: 10 nF 100 VL poliestere
C4: 10 nF 100 VL poliestere
C5: 10 nF 100 VL poliestere
C6: 47 pF ceramico
C7: 22 μF 63 VL elettrolitico
C8: 10 nF 100 VL poliestere
C9: 100 μF 25 VL elettrolitico
C10: 10 μF 63 VL elettrolitico
C11: 33 μF 63 VL elettrolitico
U1: LM324N
U2: LM3914N
LD1: LED 5 mm verde
LD2: LED 5 mm verde
LD3: LED 5 mm verde
LD4: LED 5 mm verde
LD5: LED 5 mm verde
LD6: LED 5 mm verde
LD7: LED 5 mm verde
LD8: LED 5 mm verde
LD9: LED 5 mm verde
LD10: LED 5 mm rosso
T1: BFR93A
T2: BC547
T3: BC547
D1: BAT85
D2: 1N4148
D3: 1N4148
D4: 1N4007
D5: 1N4148
JP1: Strip maschio 2 vie
BZ1: Buzzer 12V
senza elettronica

Varie:
- Zoccolo 7+7
- Zoccolo 9+9
- Jumper
- Morsetto 2 poli (3 pz.)
- Clip batteria 9V
- Circuito stampato S1251 (74x85 mm)

CARATTERISTICHE TECNICHE

- **Frequenza ricevibile: 2 MHz ÷ 1 GHz**
- **Sensibilità regolabile**
- **Indicazione tramite barra di LED e strumento a lancetta**
- **Regolazioni: sensibilità in antenna e del vu-meter**
- **Tensione di alimentazione: 9V**
- **Corrente assorbita: con strumento a lancetta: 35 mA**
- **Corrente assorbita con vu-meter a LED: 130 mA**

tale segnale di ampiezza estremamente ridotta e ad alta frequenza, è necessario utilizzare un circuito a basso rumore (LNA, Low-Noise Amplifier) e con un'elevata banda passante, come richiesto per rintracciare trasmettitori che possono operare su una gran vastità di frequenze. Il segnale amplificato viene quindi raddrizzato e livellato, ovvero trasformato in una tensione continua di valore proporzionale all'ampiezza dell'onda in ingresso. Questa tensione continua viene poi ulteriormente amplificata per adattarne il livello a ciò che è ri-

chiesto per pilotare uno strumentino a lancetta (vu-meter analogico) oppure il voltmetro digitale a barra di LED da noi utilizzato per la visualizzazione.
La tensione in uscita dall'amplificatore viene anche sfruttata per pilotare un "generatore di nota", composto da un oscillatore controllato in tensione (VCO), che fa riprodurre al cicalino ad esso collegato un segnale acustico di frequenza proporzionale al valore della tensione applicata in ingresso e quindi all'ampiezza del segnale radio captato dall'antenna del circuito. Avendo chiaro il funzionamento di massima del circuito, passiamo ad esaminarne lo schema elettrico dettagliatamente, nello stesso ordine con cui abbiamo descritto i blocchi che lo compongono: il segnale captato dall'antenna viene applicato tramite il condensatore di disaccoppiamento C6 sulla base di T1, che rappresenta il "cuore" dello stadio amplificatore a radiofrequenza; le sue caratteristiche condizionano le prestazioni e la banda passante del circuito. È stato pertanto scelto il modello BFR93A, caratterizzato da una frequenza di transizione tipica (ovvero il punto in cui il suo guadagno in corrente scende, per via delle capacità interne, a 0 dB) di ben 6 GHz.
Il transistor è collegato in una configurazione ad emettitore comune: il segnale amplificato viene prelevato dal suo collettore. La corrente di polarizzazione di base del transistor viene ottenuta collegando R8 sul collettore del medesimo, anziché al positivo di alimentazione; in questo modo si ottiene un effetto di controreazione che fornisce un certo grado di stabilizzazione del punto di lavoro di T1, anche nei confronti di una variazione della tensione di alimentazione (tipicamente quella dovuta alla scarica della batteria), anche se questo porta ad una riduzione del guadagno dello stadio. Con i valori dello schema, la tensione di polarizzazione sul collettore di T1 si deve aggirare sui 4,5 volt.
A questa tensione viene sovrapposto il segnale RF amplificato e pronto ad essere raddrizzato, e di questo si occupa D1, che pure dev'essere un diodo ultrarapido in grado di poter commutare alla frequenza del segnale applicatovi. Nello schema, il condensatore C4 preleva dal collettore di T1 la sola componente di segnale (ed elimina la tensione continua di

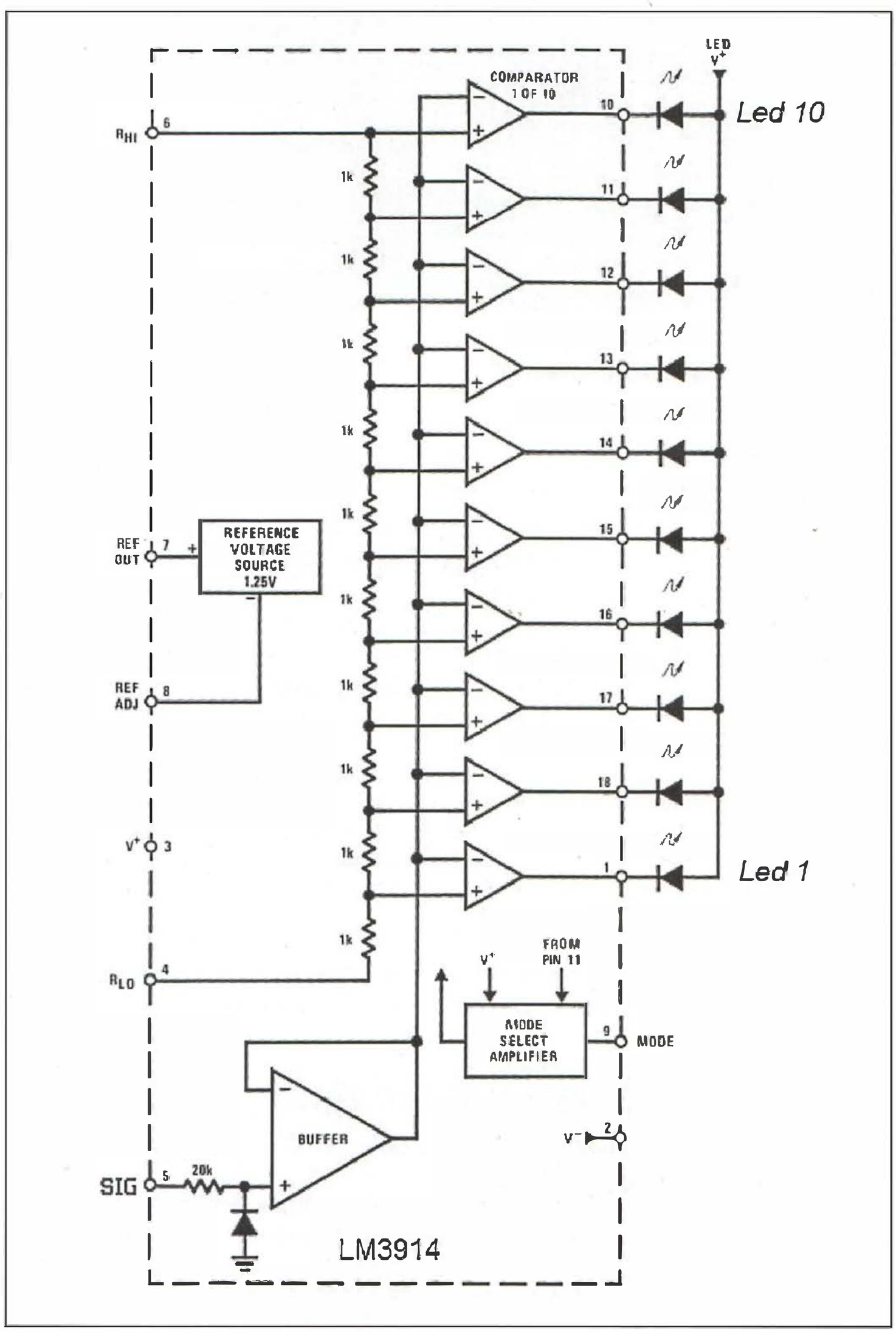


Fig. 2 *- Schema interno dell'integrato LM3914.*

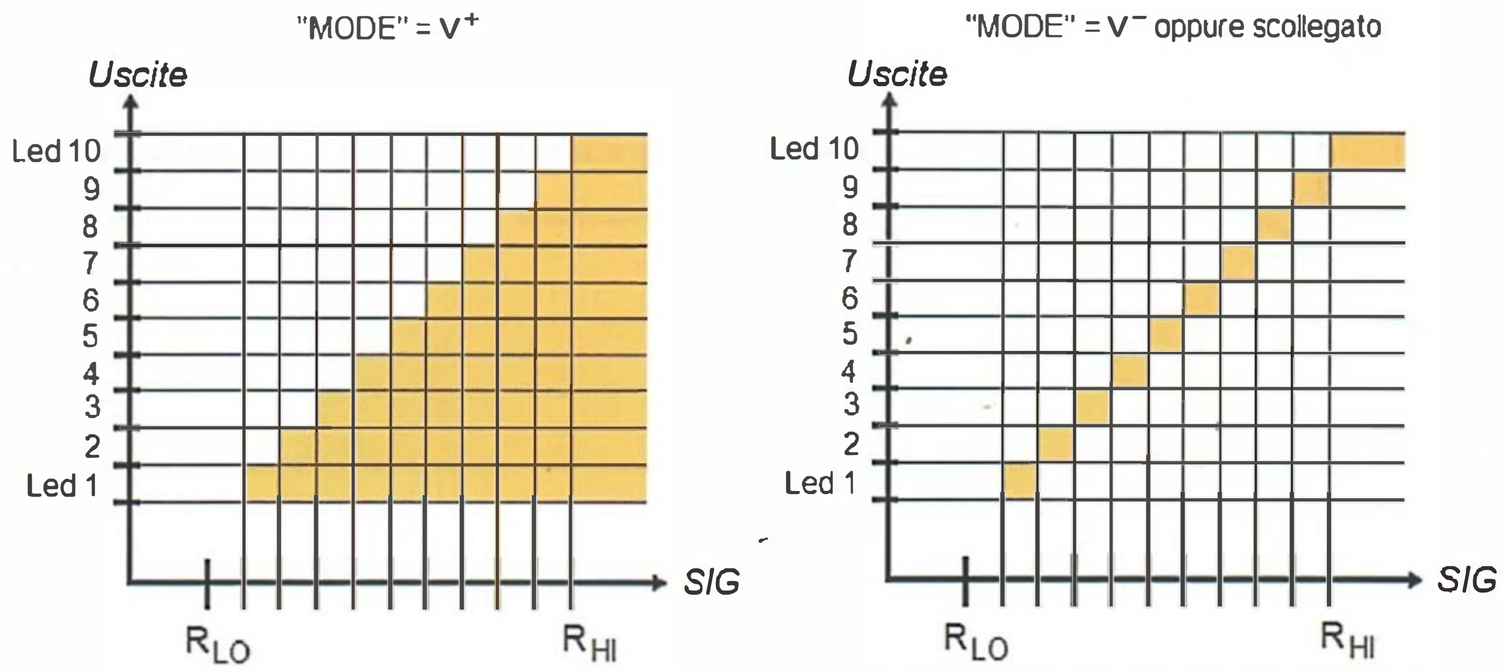


***Fig. 3** - Corrispondenza tra la tensione d'ingresso e accensione delle uscite.*

polarizzazione) e la applica a D1 che, così collegato, effettua un raddrizzamento a singola semionda negativa: infatti le semionde positive del segnale applicato sull'anodo mandano in conduzione il diodo, che le "cortocircuita" al potenziale (fornito dal partitore di tensione formato da R4 e R5) cui è collegato il catodo; tale potenziale è costante (anche per via delle capacità C3 e C11), e viene pertanto visto dal segnale (che è invece *tempo-variante*) come una massa virtuale. Nella realtà il diodo presenta una caduta di tensione diretta che ritarda il suo passaggio in conduzione durante le semionde positive e pertanto anche una parte del segnale positivo "passa oltre" lo stadio raddrizzatore, ma comunque il funzionamento del circuito non viene alterato in maniera rilevante.
Il segnale così raddrizzato entra nello stadio amplificatore cui fanno capo U1a ed i componenti ad esso collegati. Questo stadio ha un guadagno in tensione elevato, che in modulo corrisponde al rapporto dei valori dei resistori R1 + R25 ed R2; di fatto, il trimmer viene sfruttato per regolare quest'amplificazione e quindi la sensibilità dello strumento.
In questo stadio viene effettuato anche il filtraggio necessario a ricavare una tensione di uscita continua e proporzionale al segnale (raddrizzato) in ingresso a R2. A questo scopo vengono utilizzati i condensatori C1 e C2; in particolare, la costante di tempo di C1 x (R1 + R25) porta ad avere una frequenza di taglio massima di 15 Hz, quindi in uscita da U1a il segnale sarà continuo oppure variabile con la modulazione del segnale d'ingresso, ma solo se questa sarà sufficientemente lenta, e praticamente visibile ad occhio nudo (provando alcuni telecomandi si è osservata una "pulsazione" della barra di LED alla frequenza di pochi hertz, dovuta alla modulazione del segnale radio che emettevano).
La tensione continua così ricavata sull'uscita di U1a, che useremo per la visualizzazione, si compone di due parti: una fissa V_M ed una variabile V_S:

$$V_{U1a} = V_M + V_S$$

La parte fissa V_M corrisponde al valore della tensione fornita dal partitore R4-R5, che viene utilizzata come "massa virtuale" per il funzionamento dell'amplificatore: pertanto è la tensione (circa 2V a batteria carica) che si ritrova anche in uscita, in assenza di segnali in ingresso. A questa tensione viene poi sommato il contributo V_S che è invece proporzionale all'ampiezza del segnale RF captato dall'antenna. Quindi per gestire il voltmetro a barra di led oppure anche l'eventuale strumentino a lancetta, dovremo sottrarre al valore V_{U1a} la quantità VM fornita dal partitore R4-R5.
Per tale motivo, il VU-meter analogico viene collegato proprio tra l'uscita di U1a e il partitore.
Siccome tale strumento è tipicamente un microamperometro da 100÷200 µA f.s., esso è posto in serie al resistore R20 il quale ha lo scopo di effettuare la conversione tensione/corrente (quindi riducendone il valore si può aumentare la sensibilità dello strumento, e viceversa), in base alla formula:

$$I_S = (V_{U1a} - V_M) / R20 = V_S / R20$$

dove I_S è la corrente che scorrerà nello strumento ed è dovuta alla sola componente di segnale V_S.
Nella formula non abbiamo tenuto conto della resistenza interna dello strumento, essendo essa trascurabile rispetto al valore di R20.
Passiamo alla descrizione dell'indicatore a barra di LED, incentrato su U2; questo integrato, come visibile dallo schema a blocchi di **Fig. 2**, contiene dieci comparatori di tensione per confrontare il livello applicato all'ingresso dell'integrato ("*SIG*", piedino 5) con altrettanti valori progressivi che vanno da un minimo, applicato sul piedino 4 (Reference Low, R_{LO}), al massimo applicato sul piedino 6 (Reference High, R_{HI}). Come pure si nota dalla **Fig. 2**, le uscite di questi comparatori vengono riportate all'esterno dell'integrato e possono pilotare direttamente dei LED (collegati ad anodo comune).
In **Fig. 3** è mostrata la "tabella di verità", ovvero la relazione tra la tensione applicata sull'ingresso *SIG* e le corrispondenti uscite (LED) attivate. Si nota che le dieci soglie di attivazione, comprese

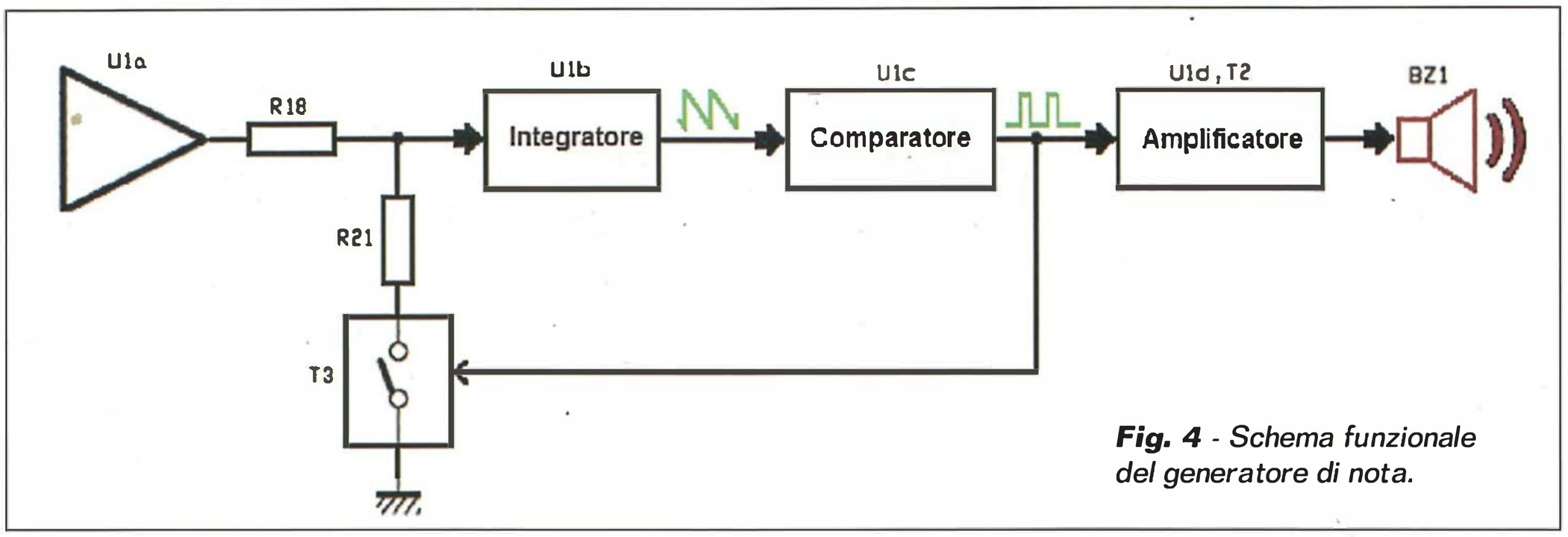


***Fig. 4** - Schema funzionale del generatore di nota.*

tra R_{LO} ed R_{HI}, sono equamente spaziate tra di loro: per tale motivo la visualizzazione è detta *lineare*. Dalla stessa figura si nota anche che vi sono due modalità di visualizzazione, dipendenti dalla tensione applicata sul piedino 9 ("MODE"):

1) Mode collegato a V+ → La visualizzazione è a *barra*: l'attivazione di un nuovo LED (in seguito ad un aumento di tensione in ingresso) non fa spegnere quelli precedenti;
2) Mode collegato a V- oppure scollegato → La visualizzazione è *a punto*: rimane acceso solo il LED relativo al livello di tensione in ingresso.

La prima modalità consente di avere una visualizzazione intuitivamente più similare al classico V-meter: più LED accesi significano maggiore intensità di segnale; la seconda modalità, per contro, garantisce che la corrente assorbita sia sempre quella di un solo LED, con il relativo allungamento dell'autonomia complessiva della batteria. Nel nostro circuito abbiamo comunque previsto l'interruttore S2 per dar modo all'utente di potersi impostare la visualizzazione che preferisce.
Oltre alle funzionalità sin qui discusse, l'integrato LM3914 contiene internamente anche un generatore di tensione, facente capo ai piedini 7 e 8, che sfruttiamo per generare la tensione di riferimento massima, ovvero quella corrispondente all'accensione dell'ultimo LED.
Tornando allo schema di **Fig. 2**, per far attivare la barra di LED in maniera proporzionale al segnale Vs colleghiamo U2 nel seguente modo:

- l'ingresso *SIG* è collegato sull'uscita di U1a, ove abbiamo visto che è presente la tensione $V_{U1a}=V_M+V_S$;
- il riferimento di tensione minima R_{LO} viene preso sul partitore R4-R5, che fornisce la tensione V_M: tale valore, pertanto, diventa lo "zero" dell'intervallo di misurazione e la visualizzazione rimane legata solo al livello di V_S;
- il riferimento di tensione massima R_{HI} (corrispondente all'accensione dell'ultimo led) si ricava dallo stabilizzatore interno di U2, collegando insieme i piedini 6 e 7; il partitore formato da R22 ed R23+R24 stabilisce il valore assoluto di tensione di uscita dello stabilizzatore; in particolare, agendo sul trimmer R23 si può impostare su R_{HI} una tensione da 2,5 a 6,2V; anche questa regolazione consente pertanto di variare la sensibilità della lettura.

L'ultimo stadio del circuito, che fa capo agli operazionali U1b, U1c e U1d e ai componenti di contorno, è il "generatore di nota": come già detto, il suo scopo è quello di fornire un breve impulso sonoro cadenzato, con frequenza crescente con il livello di segnale captato.
Per farvi meglio comprendere il funzionamento, ne proponiamo lo schema funzionale in **Fig. 4**. Il principio è quello dell'oscillatore controllato in tensione (VCO), anche se il nostro circuito non genera un segnale sinusoidale ma bensì un'onda quadra, e sarebbe dunque più corretto chiamarlo "multivibratore astabile controllato in tensione". Per avere la frequenza di quest'onda legata all'ampiezza del segnale RF captato, la tensione di controllo viene prelevata sempre sull'uscita di U1a. Questa tensione viene applicata all'integratore realizzato con l'operazionale U1b ed i componenti al contorno: sull'uscita di quest'ultimo viene pertanto generato un segnale proporzionale alla funzione integrale del segnale applicato all'ingresso. Supponendo che tale segnale rimanga costante, l'uscita di U1b è una tensione che varia linearmente nel tempo, ovvero è una rampa, la cui ripidità è proporzionale al livello di tensione d'ingresso, quindi V_{U1a}.
Specifichiamo che l'integratore è invertente, quindi la tensione V_{U1a} genera sull'uscita di U1b una rampa di tipo discendente.
Il successivo stadio è un comparatore con isteresi, realizzato con

l'operazionale U1c, che riceve questo segnale a rampa sull'ingresso invertente. Sull'ingresso non invertente si fornisce invece una tensione di riferimento (chiamiamola V_{RIF}) ricavata tramite le reti resistive formate da R11, R14, R15 e R19; in particolare, la resistenza R11 fa sì che questa tensione sia condizionata anche dalla tensione di uscita di U1c; poiché questi è configurato per funzionare come comparatore (non vi è reazione tra uscita e ingresso invertente), l'uscita potrà solo assumere due stati, zero (tensione nulla) e uno (tensione positiva), a seconda che la tensione sull'ingresso invertente superi quella dell'ingresso non invertente, o viceversa. Per la presenza di R11, in corrispondenza di questi due stati avremo anche due valori possibili della tensione V_{RIF}: chiamiamoli VTH+ e VTH- (con $V_{TH}+ > V_{TH}-$). Riassumendo:

- Uscita U1c a zero logico → $V_{RIF} = V_{TH-}$
- Uscita U1c a uno logico → $V_{RIF} = V_{TH+}$

In definitiva, la tensione in arrivo dall'integratore viene confrontata con la V_{RIF} che può assumere uno di questi due livelli. Durante la rampa discendente, che è la situazione che stavamo analizzando, il comparatore si trova nello stato zero; pertanto, $V_{RIF} = V_{TH-}$: quando la rampa scende al disotto di questo livello, il comparatore commuta nello stato uno e la sua uscita assume il massimo valore positivo. Questo fa sì che la V_{RIF} applicata sull'ingresso non invertente salga a V_{TH}^{+}; inoltre il transistor T3, collegato sull'uscita tramite R12, passa in conduzione. A sua volta T3, tramite R21, abbassa la tensione d'ingresso all'integratore ad un livello stabilito dalla partizione di R21 e R18 della tensione V_{U1a}. I valori delle resistenze sono stati scelti in modo che, quando T3 è in conduzione, la tensione sul piedino invertente di U1b venga ad essere inferiore a quella presente sul piedino non invertente: in questa situazione la rampa in uscita a U1b inverte la pendenza, con la tensione che inizia quindi a salire (partendo dal punto a cui era arrivata nell'istante in cui T3 è passato in conduzione, quindi il livello VTH-). Il comparatore U1c però non "ri-commuta" subito a zero perché la V_{RIF} è salita, come visto, a V_{TH}^{+} (che è maggiore): questo stato di rampa ascendente permane quindi fin tanto che essa non raggiunge questo nuovo livello. Quando ciò succede, il comparatore commuta nuovamente nello stato 0, riportando $V_{RIF} = V_{TH}^{-}$ e spegnendo T3; la tensione in uscita dall'integratore torna a scendere ed il ciclo si ripete.

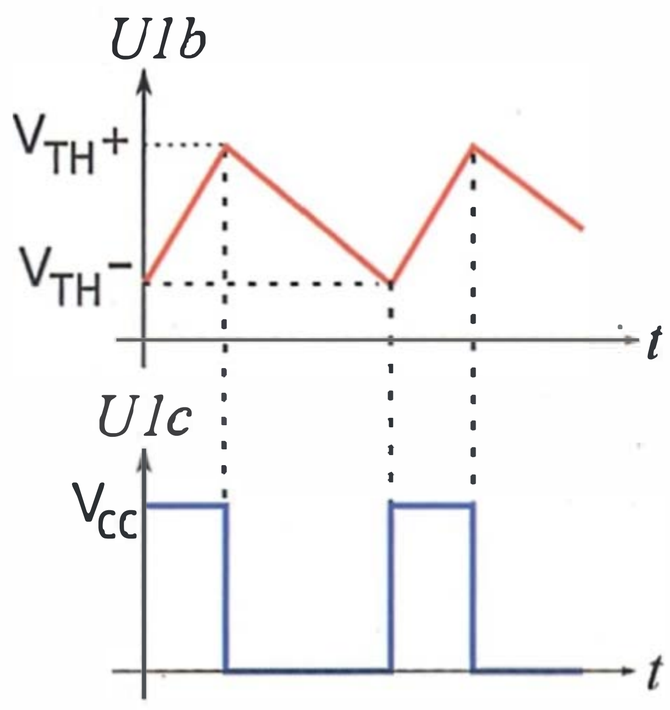


***Fig. 5** - Forme d'onda in uscita dall'integratore e dal comparatore.*

La Fig. 5 riporta i grafici delle forme d'onda in uscita da U1b (integratore) e U1c (comparatore) correlate tra loro.
Invece la Fig. 6 riporta il risultato di una simulazione (eseguita con il programma LTspice), che

Scegliere e dirigere l'antenna

La captazione del campo elettromagnetico presente nell'ambiente può essere effettuata (se non occorrono particolari sensibilità) anche utilizzando uno spezzone di filo rigido: il nostro circuito prevede infatti un singolo ingresso per elementi "monopolo", in cui ci si affida alla massa del circuito (o del terreno stesso) come piano di riferimento. Se tale piano fosse ideale la lunghezza del conduttore preferibile sarebbe, in base alla teoria delle antenne, da scegliersi di poco inferiore ad un quarto della lunghezza d'onda del segnale che si vuole captare; ricordiamo che la lunghezza d'onda si ricava dalla formula:

$$\lambda = c / f$$

dove λ (lambda) è la lunghezza d'onda espressa in metri, c la velocità della luce nel vuoto (circa $3 \cdot 10^8$ metri al secondo) e f è la frequenza dell'onda (in Hertz). Tuttavia, poiché la frequenza di trasmissione delle "cimici" non viene normalmente resa nota (soprattutto alla persona intercettata!), consigliamo di impiegare un'antenna telescopica a filo ed andare per tentativi, effettuando misurazioni regolando l'antenna su lunghezze diverse. In base a quanto detto sopra, per rilevare microspie modulate in FM (nella gamma "commerciale" 88÷108 MHz) la lunghezza dovrà essere di circa 75 cm, mentre con dispositivi operanti in UHF (dai 400 MHz in su) bisognerà ridurre la lunghezza a 20 cm, o anche meno per frequenze più elevate. Tenete, infine, presente che la sensibilità dello strumento è legata anche alla posizione dell'antenna rispetto a quella della sorgente irradiante: pur considerando che l'antenna rettilinea ha un diagramma di radiazione di tipo radiale (con sensibilità massima per i segnali diretti perpendicolarmente al suo asse), la bontà della captazione dipende da come è posizionata a sua volta l'emittente, dal tipo di antenna che utilizza (in quanto determina la polarizzazione del campo), ecc. Pertanto per fare una buona ricerca bisogna anche cercare di posizionare la "stilo" lungo diverse orientazioni (in orizzontale, in verticale e anche con diverse inclinazioni intermedie).

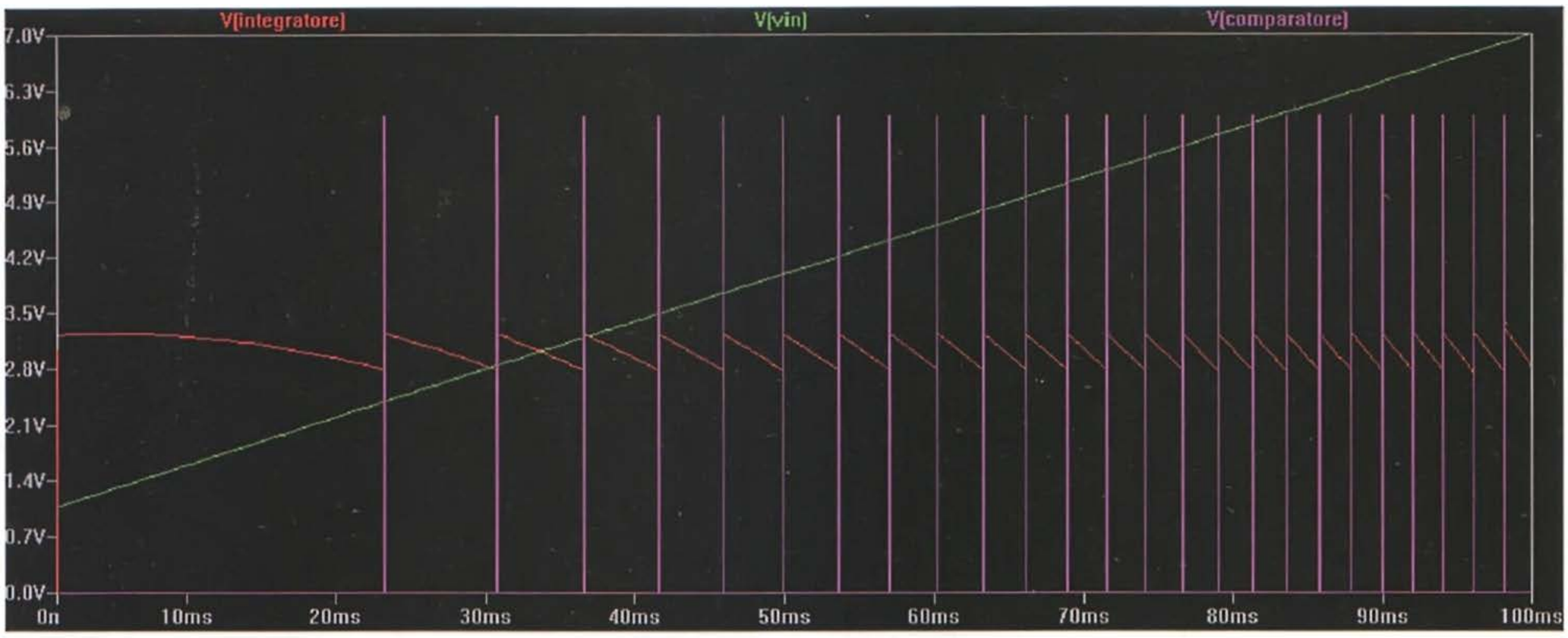


Fig. 6 *- Simulazione condotta con LTspice.*

mostra l'andamento delle tensioni all'uscita dell'integratore (in rosso) e del comparatore (in lilla), per valori crescenti della tensione d'ingresso Vin (corrispondente all'uscita di U1a, in verde); si nota come, con l'aumentare di essa, aumentino pure le frequenze dei due segnali. Infine, come mostrato nello schema a blocchi, l'onda quadra in uscita al comparatore U1c viene amplificata da U1d e T2 per poter pilotare il cicalino piezoelettrico BZ1. Il ponticello JP1 è stato inserito per consentire all'utilizzatore di disattivarlo, nel caso in cui non si desiderasse avere anche il segnale sonoro.

L'alimentazione del circuito può essere fornita da una batteria da 9V del tipo 6LR61 (alcalina) che permette un'autonomia in funzionamento continuo di circa cinque ore, se si utilizza la modalità di visualizzazione "a punto", ovvero con un solo led acceso per volta. Alternativamente si può anche fornire al circuito una tensione compresa tra 8 e 12V, prelevata per esempio da un alimentatore stabilizzato in grado di erogare una corrente di almeno 50mA per la visualizzazione a punto, oppure di 200 mA se si opta per l'alimentazione a barra. Il diodo D4 serve solo a proteggere la scheda da indesiderate inversioni di polarità.

MONTAGGIO

Il circuito non presenta particolari difficoltà di montaggio, in quanto vi sono quasi esclusivamente componenti a montaggio a foro passante (tranne T1, che è del tipo a montaggio superficiale e che conviene lasciare per ultimo). L'unica accortezza è come sempre il rispetto dei versi d'inserzione dei componenti polarizzati (diodi e condensatori elettrolitici *in primis*), seguendo il piano di montaggio mostrato in queste pagine.

Inoltre è consigliabile non saldare direttamente i circuiti integrati U1 e U2, ma fare uso degli appositi zoccoli. I led predisposti sul circuito stampato sono del tipo tondo da 5mm, ma si possono utilizzare anche di altro tipo (e colore).

Una volta completato il montaggio di tutti i componenti tradizionali, si può girare la scheda sul lato inferiore e saldarvi T1 sulle piazzole predisposte (**Fig. 7**). Poiché il componente è molto piccolo, suggeriamo di aiutarsi con delle presselle o delle piccole pinzette per tenerlo, e di utilizzare un saldatore con punta fine (massimo 1mm di diametro) e con temperatura non superiore a 400°C (se si può regolare); in ogni caso, cercare di non insistere tenendo poggiata la punta per più di tre-quattro secondi sulla saldatura.

Per quanto riguarda l'antenna, si può iniziare saldando uno spezzone di filo di rame, nudo oppure isolato, (per la cui lunghezza si veda il riquadro "La scelta dell'antenna") nella piazzola ANT, oppure collegarvi un'antenna stilo vera e propria, avvitandola nel foro predisposto. Si possono quindi effettuare

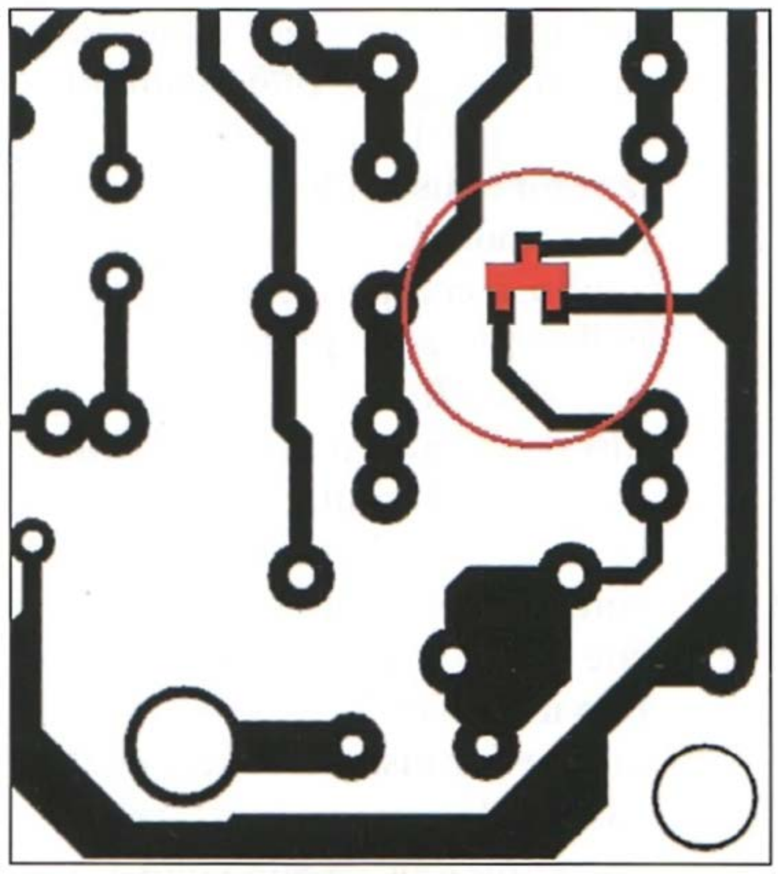


Fig. 7 *- Posizione corretta per il montaggio del componente T1.*

le connessioni sui morsetti per portare l'alimentazione, i fili per connettere gli interruttori volanti S1 ed S2, e (volendo) il vu-meter. Infine, dopo avere inserito gli integrati U1 e U2 nei rispettivi zoccoli, il circuito è pronto al funzionamento.
Avendone la possibilità, si potrebbe effettuare una piccola verifica collegandolo ad una sorgente di alimentazione di circa 10 V stabilizzati e, con S1 ed S2 chiusi, verificare che l'assorbimento non superi i 50 mA (più circa 12 mA aggiuntivi per ciascun LED acceso).

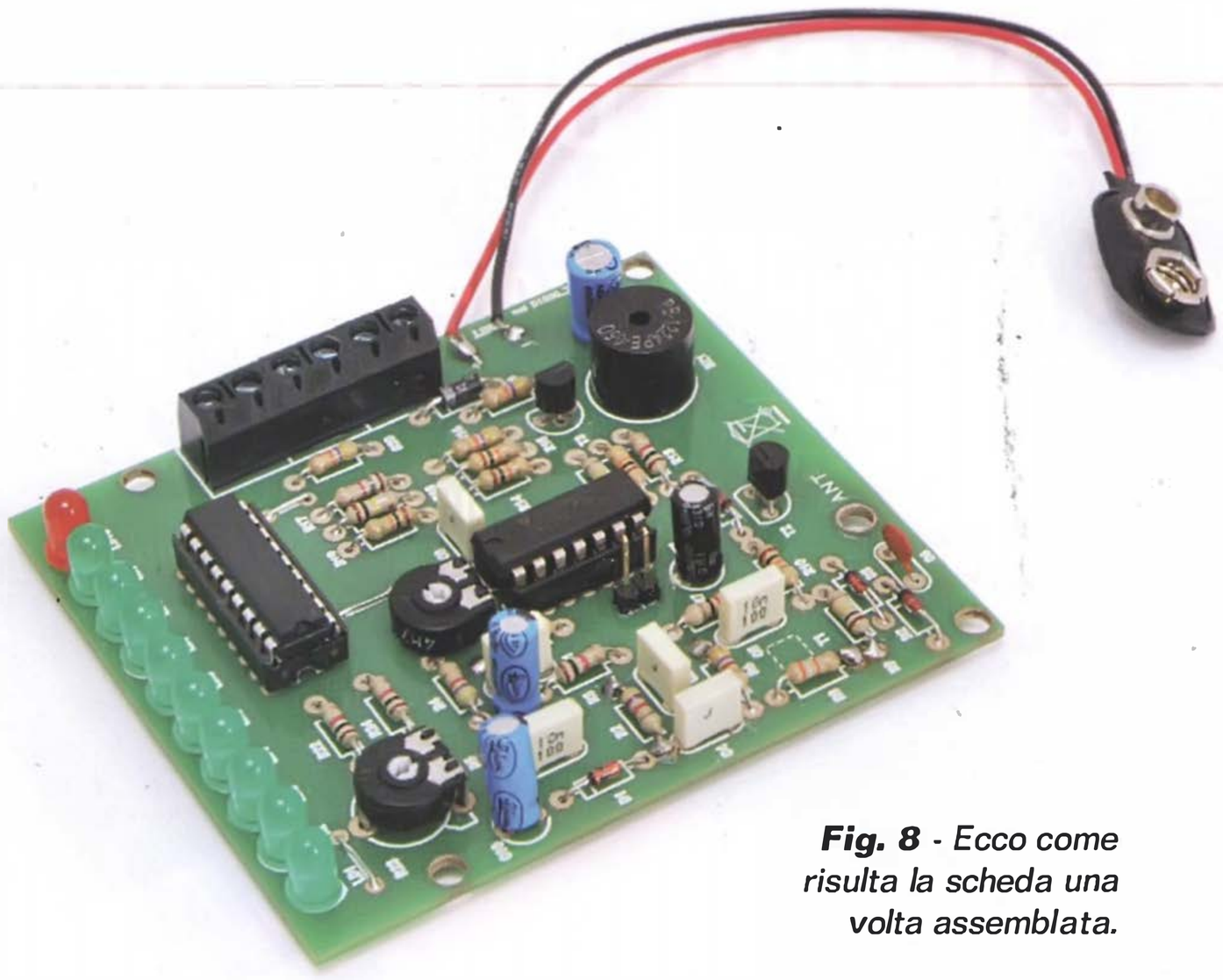


***Fig. 8** - Ecco come risulta la scheda una volta assemblata.*

UTILIZZO

Il circuito non ha bisogno di taratura, sebbene abbia due trimmer da regolare nell'utilizzo: R25 permette di variare il guadagno dell'amplificatore U1a, in modo da aumentarlo se non rilevate segnale, oppure diminuirlo se lo strumento è già a fondo scala magari per via di un elevato campo elettromagnetico "di fondo" (dovuto alla presenza di varie fonti RF in zona). Il trimmer R23 è invece utile per impostare il livello di fondo-scala dello strumento, ovvero il livello di segnale cui corrisponde l'accensione dell'ultimo LED della scala: questa regolazione consente di aumentare ulteriormente la sensibilità del rilevatore.
Il circuito ha una sensibilità sufficiente per rilevare trasmettitori con potenze di anche solo poche decine di milliwatt, però va considerato che, trattandosi di un ricevitore a larga banda (da pochi MHz ad oltre un GHz), esso non potrà mai avere le prestazioni di un circuito che faccia uso di amplificatori sintonizzati. Questo significa che, ad esempio, il tipico telecomando a radiofrequenza che vi apre il cancello di casa a decine di metri potrà essere "visto" dal nostro circuito solo se lo azionate a 50 cm di distanza dall'antenna, o meno. Il ricevitore dell'apricancello lo riceve a distanza maggiore perché ha un circuito accordato sulla sola frequenza del radiocomando...
Un altro svantaggio dell'elevata larghezza di banda è la difficoltà di utilizzo in ambienti molto "inquinati elettromagneticamente", ad esempio nei laboratori o comunque nelle stanza in cui siano in funzione apparecchi che emettono, intenzionalmente o meno, campi RF (telefonini cellulari e cordless, dispositivi WiFi, forni a microonde, ecc.); in tali ambienti lo strumento potrebbe andare a fondo scala anche senza rilevare una reale microspia nelle vicinanze. In tal caso bisogna agire sul trimmer R25 per ridurre l'amplificazione, anche se questo si riflette ovviamente in un degrado delle prestazioni di rilevazione. Viceversa, se anche in vicinanza di un trasmettitore RF non si accendesse alcun LED, bisognerà incrementare il guadagno aumentando la resistenza di R25.

CONCLUSIONI

Chiudiamo con due ultime raccomandazioni: utilizzate batterie cariche, ovvero tali che la loro tensione sia sempre superiore a 8V (durante il funzionamento: misurarle a circuito spento non vale!) e, nel caso si ricerchino delle microspie, provare a parlare ad alta voce o comunque a produrre del rumore acustico: per risparmiare energia, infatti, tali dispositivi in assenza di rumore tipicamente non trasmettono alcun segnale e non sono quindi intercettabili.

per il MATERIALE

Tutti i componenti presentati nel progetto sono di facile reperibilità; il master del circuito stampato può essere scaricato dal sito della rivista (*www.elettronicain.it*).
Per chi non vuole cimentarsi nell'autocostruzione può acquistare presso Futura Elettronica apparecchiature commerciali in grado di operare con le stesse finalità di quella presentata nel progetto, come ad esempio il misuratore di campi elettromagnetici (cod. VTEMF) disponibile al prezzo di Euro 46,00.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775
http://www.futurashop.it

LA TENSIONE DI RETE OVUNQUE TU SIA!

Nuovi inverter economici e potenti

Inverter da 12 Vdc a 220 Vac con forma d'onda d'uscita sinusoidale modificata dotati di protezione da sovraccarico, sovratensione d'ingresso, cortocircuito e allarme batteria scarica. Dispongono di uscita USB supplementare (5 V – 500 mA), presa di tipo Schuko e cavi per il collegamento alla batteria.

Ideali su camper, camion, roulotte, per alimentare TV, VCR, PC, impianti HI-FI e molto altro!

300 W



€ 39,00
Cod. FR670M300

600 W



€ 54,00
Cod. FR670M600

1500 W



€ 148,00
Cod. FR670M1500

Prezzi IVA inclusa.

...e per le tue batterie scegli FIAMM!

Batteria ricaricabile ermetica 12 V - 1,2 Ah



€ 18,00
Cod. NP12-12

Batteria ricaricabile ermetica 12 V - 2 Ah



€ 19,50
Cod. NP21-12

Batteria ricaricabile ermetica 12 V - 7,2 Ah



€ 32,00
Cod. NP72-12

FUTURA ELETTRONICA®

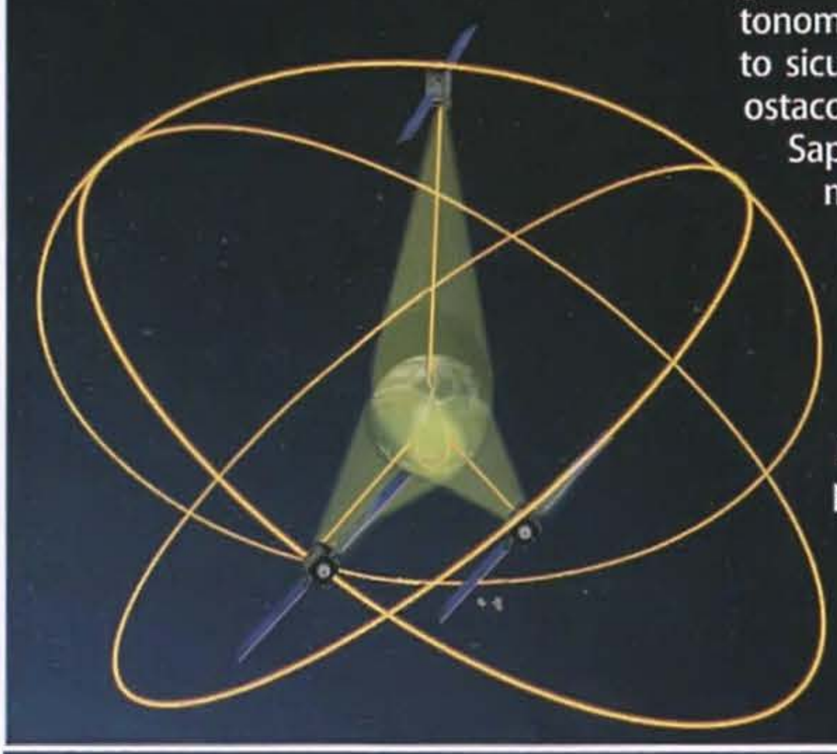
www.futurashop.it

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche di questi prodotti e acquisti on-line su *www.futurashop.it*

Sapcorda, un sistema di localizzazione satellitare con precisione centimetrica

Le società tedesche Bosch e Geo++, la svizzera U-blox e la giapponese Mitsubishi Electric hanno dato vita alla joint venture *Sapcorda Services* con lo scopo di fornire servizi di posizionamento globale di precisione centimetrica basati su reti GNSS, sfruttando anche le reti cellulari e Internet per la correzione degli errori. I dirigenti delle quattro aziende hanno affermato che per rispondere alla richiesta di precisione centimetrica per applicazioni come la guida autonoma è necessario creare servizi di correzione in tempo reale.



Il sistema di posizionamento globale GPS degli Stati Uniti presenta una precisione di circa 10 metri, mentre il corrispondente sistema europeo Galileo, con 18 dei suoi 24 satelliti attualmente in orbita, ha un'accuratezza di 1 metro. Tuttavia, anche l'accuratezza di un metro non è sufficiente quando si tratta di garantire, per esempio, che i veicoli autonomi mantengano un posizionamento sicuro nella corsia ed evitino alcuni ostacoli.

Sapcorda, in sostanza, intende aumentare la precisione del sistema Galileo in Europa utilizzando stazioni di riferimento terrestre in grado di controllare gli errori del segnale satellitare, in particolare gli errori causati dalla ionosfera e dalla troposfera, che possono alterare il segnale radio. Le stazioni invieranno i dati via Internet a un centro di controllo che invierà i dati di correzione al satellite e agli utenti attraverso la rete di telefonia mobile.

Lo schema è simile al sistema satellitare giapponese Quasi-Zenith Satellite System (QZSS) che mira a fornire servizi di posizionamento e di navigazione di precisione centimetrica in Giappone. Il sistema impiega 1.300 stazioni di riferimento per correggere costantemente gli errori del satellite. I dati corretti vengono quindi compressi e trasferiti ad una velocita di 2 Mbit/sec al sistema satellitare e agli utenti.

Il segmento spaziale del progetto dovrebbe essere simile a quello giapponese con quattro satelliti in orbita geostazionaria.

Tre dei quattro satelliti mantengono un'orbita geosincrona inclinata che traccia una figura asimmetrica nord-sud a forma di otto sopra il Giappone. L'orbita si allarga a sud, verso l'Australia, mentre il suo percorso si restringe in Giappone. Questa disposizione garantisce che almeno un satellite sia sempre visibile nel cielo sopra il Giappone, da cui il nome Quasi-Zenith Satellite System.

www.sapcorda.com

Europen XFEL, inaugurato il supermicroscopio europeo

È stato inaugurato il primo settembre ad Amburgo lo European XFEL, il super microscopio europeo realizzato anche grazie al contributo italiano con l'INFN. Si tratta di un laser a elettroni liberi che permetterà di visualizzare la struttura di macromolecole in azione, girando dei film molecolari.

A differenza delle tradizionali tecniche per determinare la struttura di una macromolecola attraverso i raggi X, lo European XFEL riesce ad analizzare anche macromolecole non cristallizzate. Questo è un vantaggio notevole in quanto, oltre a permettere di visualizzare la struttura di macromolecole difficili da cristallizzare, consente di scattare fotografie di macromolecole in momenti diversi della loro vita. E raccogliendo molti scatti diversi della stessa molecola, si può ottenere un filmato del movimento della macromolecola in attività, sia questa un enzima, un virus o un catalizzatore.

Il laser è lungo 2,1 km. Genera raggi X accelerando gli elettroni liberi lungo un tunnel di 1,7 km e curvando il percorso di questi elettroni attraverso una serie di magneti. La radiazione emessa è molto intensa, i campioni colpiti emettono molti fotoni che permettono di rivelare la struttura atomica della macromolecola analizzata. La radiazione è così intensa che i campioni sono poi distrutti. L'INFN ha contribuito in modo essenziale alla realizzazione dell'acceleratore di elettroni sviluppando al laboratorio LASA di Milano guidato da Carlo Pagani alcuni degli elementi chiave: tra questi la sorgente fotoemissiva di elettroni freddi e intensi, i moduli superconduttivi di accelerazione e il sistema di terza armonica per la linearizzazione del fascio. Lo European XFEL è costato complessivamente 1,2 miliardi di euro e l'Italia ha contribuito al progetto con un finanziamento, stanziato dal MIUR e mediato dall'INFN, di circa 40 milioni.

I primi esperimenti del laser a elettroni liberi europeo partiranno a breve. E in occasione dell'inaugurazione del più intenso laser di raggi X al mondo, cinque fasci di luce laser hanno illuminato il cielo di Amburgo fino al 3 settembre.

www.infn.it

Un'estate di successi per SpaceX

Lancio dopo lancio, il vettore Falcon 9 della società SpaceX di Elon Musk, si va affermando come uno dei razzi più affidabili (e più economici) nella storia della conquista dello spazio.

Nei mesi scorsi Falcon 9 ha eseguito una serie di lanci tutti coronati da successo, sia per quanto riguarda la regolare messa in orbita del paylod, sia per quanto riguarda il rientro sulla terra del primo stadio del vettore, quello con i 9 motori Merlin.

L'ultimo, in ordine di tempo, riguarda la messa in orbita del veicolo militare X37-B, un mini shuttle progettato dall'Air Force americana con caratteristiche simili a quelle della storica navetta della NASA.

In precedenza un Falcon 9 aveva lanciato con successo il satellite di osservazione della Terra Formosat-5 per conto della National Space Organization di Taiwan.

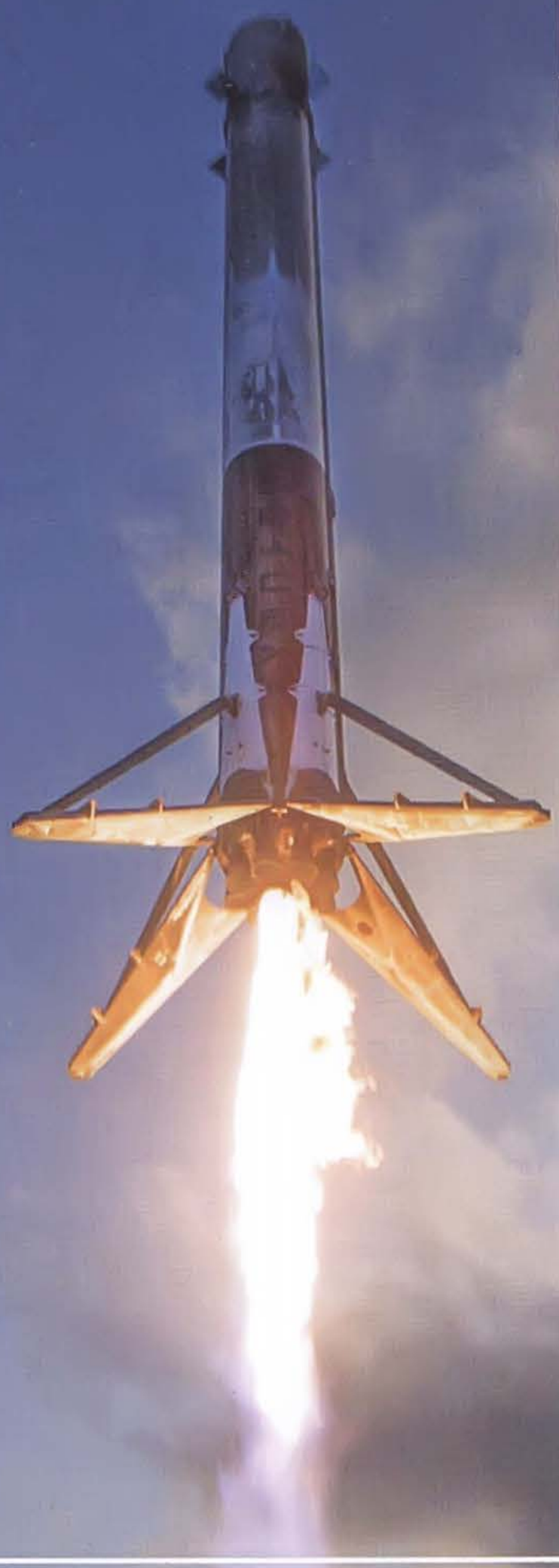
Formosat-5 è il primo programma spaziale progettato, sviluppato e gestito interamente dalla National Space Organization di Taiwan; scopo della missione è quello di promuovere esperimenti scientifici e di ricerca in ambito spaziale e migliorare le capacità tecnologiche di Taiwan per future missioni spaziali – continuando a fornire agli utenti immagini del globo alla pari del suo predecessore Formosat-2 (lanciato nel maggio 2004).

Tra i principali strumenti a bordo, il Remote Sensiting Instrument (RSI) un sistema di telecamere che osserverà la Terra e sarà in grado di raggiungere una risoluzione di 2 metri in bianco e nero e di 4 metri a colori, e lo strumento Advanced Ionospheric Probe costruito dalla National Central University di Taiwan.

Ad agosto un Falcon 9 aveva portato nello spazio la navicella Dragon (anch'essa realizzata da SpaceX) destinata a portare rifornimenti e nuovi strumenti scientifici alla Stazione Spaziale Internazionale.

Un lancio particolarmente importante per il nostro paese perché a bordo della capsula Dragon si trovavano gli esperimenti previsti per la missione dell'Agenzia Spaziale Italiana VITA. i quattro esperimenti che compongono la cosiddetta "ASI Biomission" (CORM, MYOGRAVITY, NANOROS, SERISM), le cartucce per la raccolta dei campioni biologici per l'esperimento IN-SITU, la giacca per la radioprotezione per l'esperimento PERSEO e le etichette che saranno utilizzate congiuntamente alla app caricata sull'i-Pad di bordo per l'esperimento ARAMIS. Anche in questo caso tutto ha funzionato senza intoppi e gli strumenti sono ora a disposizione di Paolo Nespoli, il veterano italiano dello spazio, che si trova a bordo della ISS dalla fine di luglio.

www.spacex.com



Arrivano le nanoantenne, mille volte più piccole di quelle tradizionali

Sono antenne "Nems", dall'acronimo inglese "sistemi nanoelettromeccanici", e promettono di abbattere uno fra i limiti che più condizionano i progettisti nel continuo processo di miniaturizzazione dei dispositivi elettronici: il rapporto fra dimensione dell'antenna, appunto, e lunghezza d'onda del segnale che devono ricevere o trasmettere. Rapporto che, anche per le antenne più compatte, difficilmente può scendere al di sotto di un decimo. Affidandosi a un sistema di concezione completamente diversa, descritto sull'ultimo numero di Nature Communications, e basato su membrane risonanti a specifiche frequenze acustiche, un team di ricercatori della Northeastern University di Boston, nel Massachusetts, è invece riuscito a sviluppare antenne con dimensioni indipendenti dalla lunghezza d'onda.

Non più vincolate alla lunghezza d'onda del segnale che debbono ricevere, le antenne si basano su membrane risonanti a frequenze acustiche specifiche. Le dimensioni di queste antenne di nuova concezione sembrano essere

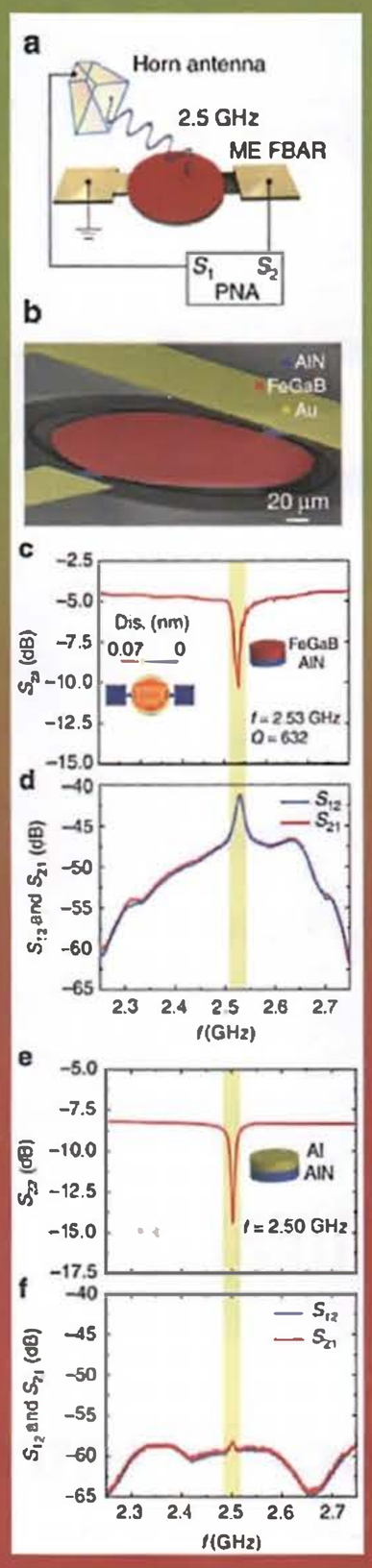
OPTSAT-3000: un satellite all'avanguardia per la Difesa italiana

È stato lanciato con successo il satellite OPTSAT-3000 del Ministero della Difesa Italiano. Il lancio è stato effettuato da Arianespace dallo spazioporto europeo di Kourou, in Guyana francese, con il lanciatore europeo VEGA, realizzato da AVIO.

Il satellite si è separato dal razzo vettore 42 minuti dopo il lancio e il primo segnale di telemetria è stato acquisito circa cinque ore dopo. L'intero sistema viene fornito da Leonardo (ex Finmeccanica) attraverso Telespazio (Leonardo 67%, Thales 33%), primo contraente alla guida di un gruppo

uno o due ordini di grandezza inferiori a quelle delle attuali antenne compatte. Le antenne tradizionali sono strutture metalliche dimensionate per essere risonanti a una specifica lunghezza d'onda della radiazione, e ovviamente questo vincolo progettuale limita moltissimo la loro miniaturizzazione. La nuova antenna progettata e sviluppata dal gruppo di Nian Sun è una membrana risonante non a una lunghezza d'onda elettromagnetica specifica, bensì ad una frequenza acustica specifica. La membrana è composta da un film sottile di materiale magnetoelettrico che, vibrando, cambia le sue proprietà magnetiche, accoppiando la vibrazione acustica con la radiazione trasmessa o ricevuta. Gli autori hanno trovato che la frequenza acustica di risonanza della membrana può essere controllata dal disegno geometrico e hanno dimostrato il funzionamento di questa tecnologia.

www.nature.com





Flip-flop qubits: nuova architettura per i computer quantistici

In uno studio pubblicato recentemente da Nature Communications, i ricercatori dell'Università australiana del Nuovo Galles del Sud descrivono una innovativa architettura per il calcolo quantistico, basata su nuovi qubits flip-flop, che promette di rendere più semplice ed economica la produzione su larga scala di chip quantistici. Il nuovo design utilizza un processore quantistico al silicio nel quale gli atomi possono essere introdotti senza la precisione necessaria con altri approcci. Allo stesso modo è possibile posizionare i quantum bits (o qubits) alla distanza di centinaia di nanometri e mantenere inalterate le caratteristiche per quanto riguarda l'entanglement. Il progetto è stato sviluppato da un team di ricercatori guidati dall'italiano Andrea Morello, Program Manager presso il Centre of Excellence for Quantum Computation and Communication Technology (CQC2T) della UNSW, che ha dichiarato che la tecnologia messa a punto è alla portata dei processi produttivi attuali.
Al nuovo progetto hanno lavorato anche Guilherme Tosi, Fahd Mohiyaddin, Vivien Schmitt e Stefanie Tenberg del CQC2T, e Rajib Rahman e Gerhard Klimeck della statunitense Purdue University. Quello che Morello e il suo team hanno messo a punto è un nuovo modo per definire uno "spin qubit" che impiega sia l'elettrone che il nucleo dell'atomo. Ma la cosa più importante è che questo nuovo qubit può essere controllato utilizzando segnali elettrici, anziché magnetici. I segnali elettrici sono molto più facili da distribuire e localizzare all'interno di un chip elettronico.
"*Con questa tecnica è anche possibile realizzare più facilmente grandi array di qubits, spingendo la distanza tra ciascun elemento a 10-20 nanometri o 50 atomi*" ha dichiarato uno dei ricercatori.

newsroom.unsw.edu.au

internazionale di aziende tra cui Israel Aerospace Industries (IAI), che ha realizzato il satellite nell'ambito di un accordo di cooperazione internazionale tra Italia e Israele, e OHB Italia, responsabile dei servizi di lancio. Composto da un satellite in orbita LEO (Low Earth Orbit) eliosincrona e da un segmento di terra per il controllo in orbita, l'acquisizione e il processamento dei dati, OPTSAT-3000, fornirà immagini ad alta risoluzione di ogni area del pianeta, permettendo all'Italia di acquisire una capacità nazionale autonoma di osservazione della Terra dallo Spazio con sensore ottico ad alta risoluzione.
OPTSAT-3000 interopererà a partire dal 2018 con il sistema di satelliti radar COSMO-SkyMed di seconda generazione – anch'esso sviluppato dall'industria italiana, con Leonardo e le sue joint venture Thales Alenia Space (Thales 67%, Leonardo 33%) e Telespazio –, integrando dati ottici e radar per fornire alla Difesa italiana informazioni di estrema precisione, risoluzione e completezza e strumenti d'analisi e operativi allo stato dell'arte.
Dopo il successo del lancio, sono ora iniziate le operazioni di verifica e test del sistema, che sarà quindi gestito interamente dal segmento di terra italiano, articolato su tre siti operativi: il Centro Interforze di Telerilevamento Satellitare (CITS) di Pratica di Mare (Roma), il Centro Interforze di Gestione e Controllo SICRAL (CIGC SICRAL) di Vigna di Valle (Roma) e il Centro Spaziale del Fucino (L'Aquila) di Telespazio.

www.leonardocompany.com

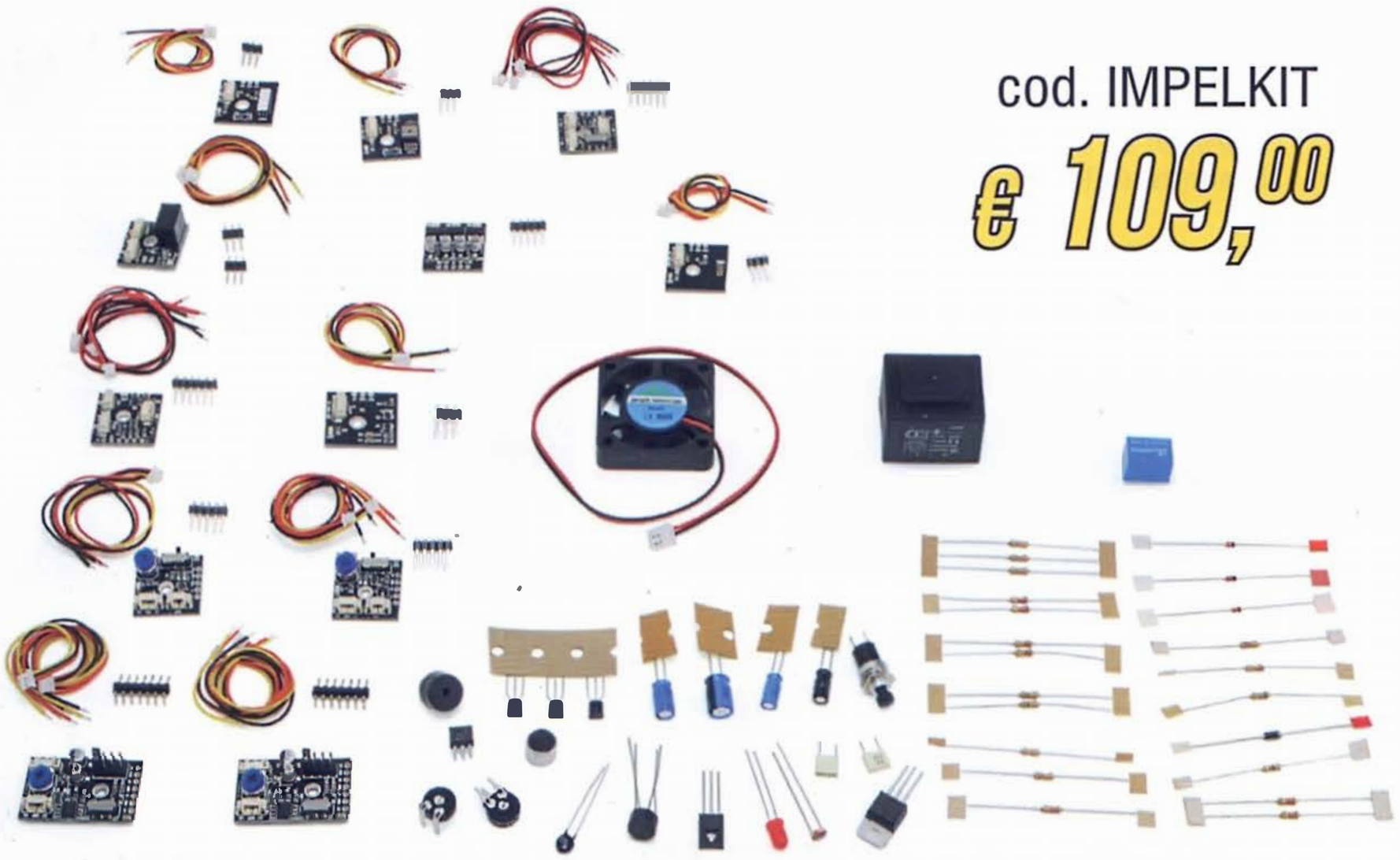
Impara l'elettronica facilmente!

Un kit pensato per chi si avvicina per la prima volta all'elettronica e vuole impararla in maniera semplice ed intuitiva.

Prezzo IVA inclusa.

Kit didattico per sperimentazione Elettronica

Contiene tutti i componenti necessari per realizzare gli esperimenti descritti nel libro "Impara l'elettronica sperimentando*".



cod. IMPELKIT

€ 109,00



*Il libro è compreso nel kit

FUTURA ELETTRONICA®

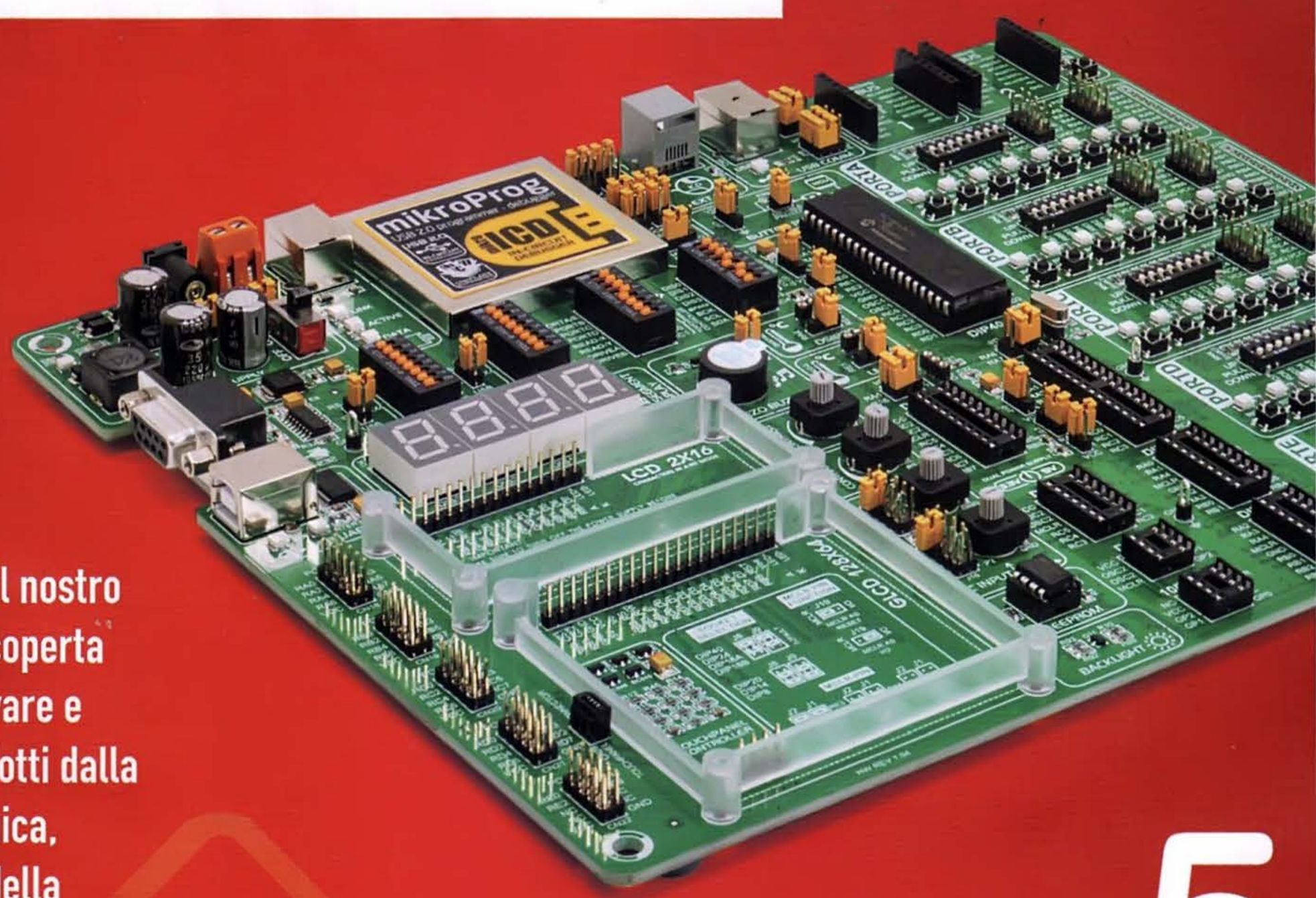
www.futurashop.it

Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Caratteristiche tecniche di questo prodotto e acquisti on-line su *www.futurashop.it*

L'ambiente

MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS



Proseguiamo il nostro viaggio alla scoperta dei tool hardware e software prodotti dalla MikroElektronica, occupandoci della gestione dei display LCD alfanumerici con la nostra piattaforma di sviluppo. Quinta puntata.

di FRANCESCO FICILI

5

Nella prima parte del nostro corso ci siamo occupati di introdurre l'ambiente di programmazione e di iniziare a compiere le prime, semplici esercitazioni, utilizzando le periferica di base (GPIO e Timers) e le periferiche analogiche (CCP e ADC) a bordo del microcontrollore. In questa quinta puntata passiamo ad esaminare i display alfanumerici ed il supporto fornito da MikroC per gestirli con la demoboard di mikroE che abbiamo scelto di adottare per questo corso.
Gli schermi LCD rappresentano uno degli strumenti indispensabili per dotare le nostre applicazioni embedded di un'interfaccia HMI (Human-Machine Interface). Come certamente saprete, gli schermi a cristalli liquidi sono attualmente, considerando tutte le possibili varianti, tra i display più diffusi sul mercato e vengono utilizzati per le più svariate applicazioni; la loro diffusione ha reso oramai obsoleti gli schermi CRT (a tubo catodico), che avevano dominato per diversi decenni il mercato dei display, nelle sue varie declinazioni. Le applicazioni di questi dispositivi sono innumerevoli: basti pensare ai

Fig. 1 – Visual TFT.



Fig. 2 – Esempio di display LCD.



Fig. 3 – Pinout di un modulo LCD basato su controller HD44780.



nostri televisori o ai monitor dei PC (sia fissi che laptop), a dispositivi di taglia più piccola, come riproduttori MP3 o stereo, o anche ai nostri smartphone, che spesso fanno uso degli ultimi ritrovati di questa tecnologia, come gli IPS o anche i Retina di Apple (quando non vengono utilizzati gli antagonisti OLED e AMOLED).
In ambito embedded, storicamente, prima del diffondersi di tecnologie più avanzate e di visualizzatori grafici, si è fatto uso massivo di display LCD basati sul controller HD44780 (detti anche LCD alfanumerici). Negli ultimi anni si stanno rapidamente diffondendo display LCD in tecnologia LED TFT o OLED con controller grafico integrato ed interfaccia seriale (generalmente SPI o UART), relegando l'uso dei display basati su HD44780 ad applicazioni a basso costo con interfaccia HMI minimale: si pensi ad esempio ai termostati, alle radiosveglie o alle vending machine, in questo tipo di contesti l'uso di LCD basati su questo controller risulta ancora un'opzione valida.
In questa puntata ci occuperemo proprio di tali dispositivi, che sono nativamente supportati da MikroC (con opportune librerie) e dalla EasyPic V7, ma si tenga presente che Mikroe ha recentemente lanciato sul mercato un prodotto, denominato Visual TFT, che permette di creare facilmente interfacce grafiche (GUI) per display in tecnolgia TFT dotati di controller FT90x (una famiglia di controller grafici sviluppata dalla FTDI inc.).

I diplay LCD con controller HD44780

Come accennato nel paragrafo precedente, una famiglia molto nota di display LCD per applicazioni embedded è quella basata sul controller HD44780; quest'ultimo è un ASIC, prodotto dalla Hitachi sul finire degli anni '80, in grado di controllare display LCD capaci di rappresentare solo caratteri (display alfanumerici). In commercio si trovano moduli costituiti dalla coppia display/controller di varie dimensioni (8x1, 8x2, 16x2 20x4, ecc.).
A causa della sua diffusione il controller HD44780 è considerato lo standard "de facto" per il controllo dei display LCD, tanto che in genere si parla di "Hitachi compatibile" in riferimento ad un qualsiasi controller che utilizzi questo genere di interfaccia.
In Fig. 2 è riportato un tipico esempio di display

MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS

LCD 16x2 con controller Hitachi-compatibile. Display di questo tipo possono avere diverse combinazioni di colore di sfondo e caratteri (ad esempio verde-nero, blu-bianco, ecc.), e sono capaci di visualizzare i caratteri standard ASCII, i simboli giapponesi Kana ed alcuni simboli grafici essenziali.
Il controller HD44780 supporta il trasferimento parallelo con 8-bit di dati e 3 linee di controllo. L'interfaccia, tramite opportuna configurazione, può funzionare anche in modalità 4-bit (nibble), nel qual caso sono utilizzate le linee dati D4-D7. I segnali di controllo sono:

- RS (Register Select): selezione comandi/dati (RS=0, invio comandi – RS=1, invio dati);
- R/W (Read/Write): selezione modalità (R/W=1, lettura – R/W=0, scrittura); l'uso di questo pin è opzionale, dato che il display è spesso utilizzato solo in modalità scrittura e quindi il pin R/W viene collegato fisso a GND;
- E (Enable): segnale di sincronismo (clock) che indica al controller che è possibile campionare i segnali presenti sul bus (triggerato sul fronte di discesa).

In Fig. 3 è riportato uno schema che evidenzia la tipica interfaccia di un display LCD basato su HD44780. Oltre al bus sono presenti anche i pin di alimentazione (Vss e Vdd), il contrasto (Vee) e il pin per la gestione della retroilluminazione (LedA, LedK).
In Fig. 4 è riportata una tipica connessione di un display LCD basato su HD44780,

Fig. 4 – Esempio di collegamento di un display HD44780.

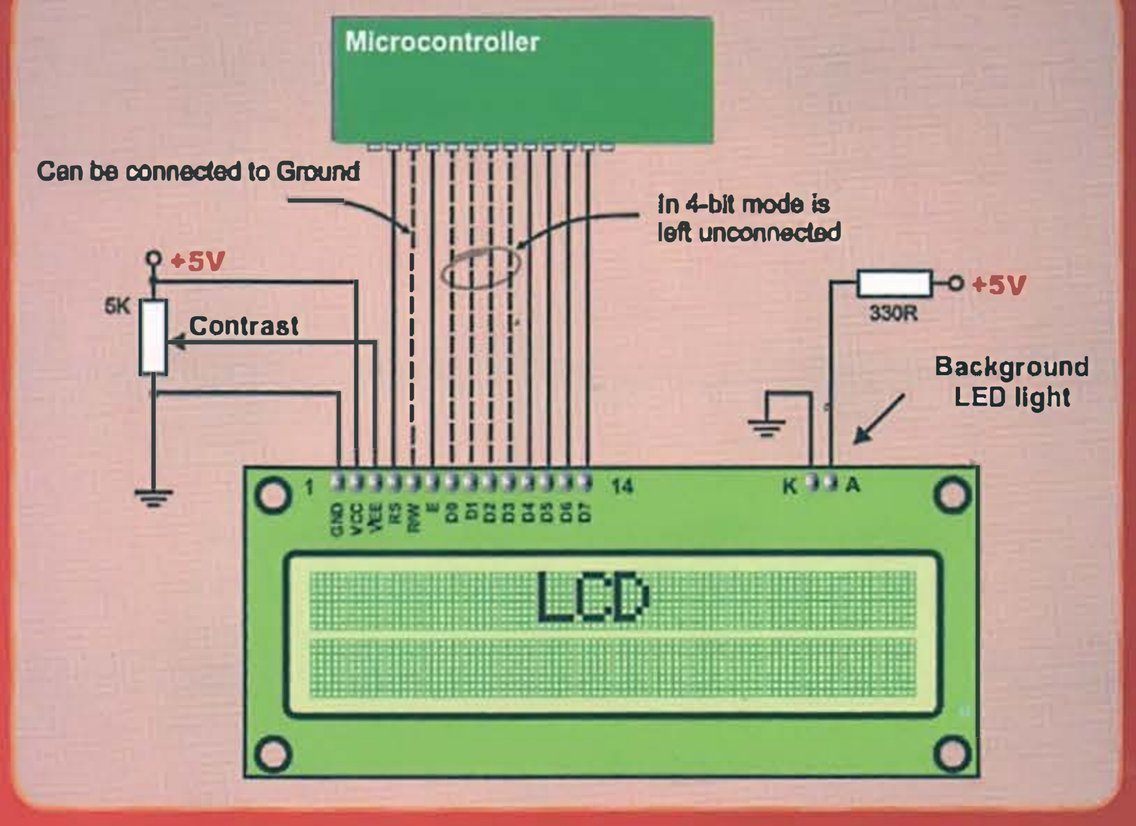
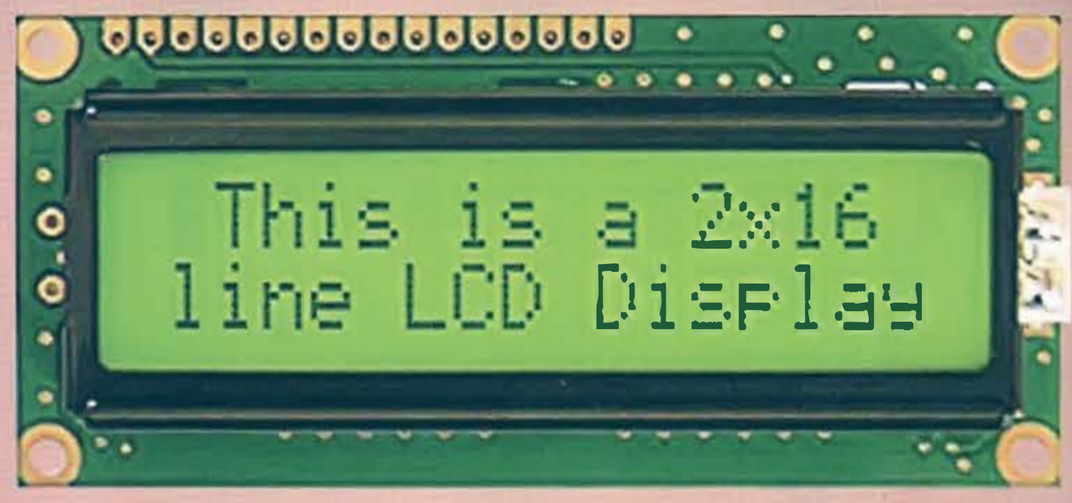


Fig. 5 – Display HD44780 in funzione.



dove il contrasto è stato collegato ad un potenziometro e la retroilluminazione è collegata fissa a 5V, in modo da essere sempre attiva. Configurazioni alternative possono prevedere il controllo del contrasto e della retroilluminazione direttamente

Variabili da definire	Descrizione	Esempio
extern sfr sbit LCD_RS:	Register Select line.	**sbit** LCD_RS **at** RB4_bit;
extern sfr sbit LCD_EN:	Enable line.	**sbit** LCD_EN **at** RB5_bit;
extern sfr sbit LCD_D7;	Data 7 line.	**sbit** LCD_D7 **at** RB3_bit;
extern sfr sbit LCD_D6;	Data 6 line.	**sbit** LCD_D6 **at** RB2_bit;
extern sfr sbit LCD_D5;	Data 5 line.	**sbit** LCD_D5 **at** RB1_bit;
extern sfr sbit LCD_D4;	Data 4 line.	**sbit** LCD_D4 **at** RB0_bit;
extern sfr sbit LCD_RS_Direction;	Register Select direction pin.	**sbit** LCD_RS_Direction **at** TRISB4_bit;
extern sfr sbit LCD_EN_Direction;	Enable direction pin.	**sbit** LCD_EN_Direction **at** TRISB5_bit;
extern sfr sbit LCD_D7_Direction;	Data 7 direction pin.	**sbit** LCD_D7_Direction **at** TRISB3_bit;
extern sfr sbit LCD_D6_Direction;	Data 6 direction pin.	**sbit** LCD_D6_Direction **at** TRISB2_bit;
extern sfr sbit LCD_D5_Direction;	Data 5 direction pin.	**sbit** LCD_D5_Direction **at** TRISB1_bit;
extern sfr sbit LCD_D4_Direction;	Data 4 direction pin.	**sbit** LCD_D4_Direction **at** TRISB0_bit;

Tabella 1 – Elenco delle variabili globali da definire.

MikroElektronika DEVELOPMENT TOOLS I COMPILERS I BOOKS

dal microcontrollore al quale il display è collegato. In Fig. 5 è riportata l'immagine di un display LCD in funzione.

La libreria LCD MikroC

La libreria LCD di MikroC fornisce una serie di API per la gestione di display LCD basati su controller HD44780, utilizzando un'interfaccia dati a 4-bit. È anche possibile creare un set di caratteri custom, utilizzando il tool "LCD Custom Character".
Le API disponibili sono:

- Lcd_Init,
- Lcd_Out,
- Lcd_Out_Cp,
- Lcd_Chr,
- Lcd_Chr_Cp,
- Lcd_Cmd.

Prototipo	void **Lcd_Init** (void).
Ritorno	Nessuno
Descrizione	Inizializza il modulo LCD.
Requisiti	Le seguenti variabili globali: • LCD_D7: Data bit 7 • LCD_D6: Data bit 6 • LCD_D5: Data bit 5 • LCD_D4: Data bit 4 • LCD_RS: Register Select (data/instruction) signal pin • LCD_EN: Enable signal pin • LCD_D7_Direction: Direction of the Data 7 pin • LCD_D6_Direction: Direction of the Data 6 pin • LCD_D5_Direction: Direction of the Data 5 pin • LCD_D4_Direction: Direction of the Data 4 pin • LCD_RS_Direction: Direction of the Register Select pin • LCD_EN_Direction: Direction of the Enable signal pin Devono essere definite prima di usare questa funzione.
Esempio	• Inizializzazione modulo LCD: *// Lcd pinout settings* **sbit** LCD_RS **at** RB4_bit; **sbit** LCD_EN **at** RB5_bit; **sbit** LCD_D7 **at** RB3_bit; **sbit** LCD_D6 **at** RB2_bit; **sbit** LCD_D5 **at** RB1_bit; **sbit** LCD_D4 **at** RB0_bit; *// Pin direction* **sbit** LCD_RS_Direction **at** TRISB4_bit; **sbit** LCD_EN_Direction **at** TRISB5_bit; **sbit** LCD_D7_Direction **at** TRISB3_bit; **sbit** LCD_D6_Direction **at** TRISB2_bit; **sbit** LCD_D5_Direction **at** TRISB1_bit; **sbit** LCD_D4_Direction **at** TRISB0_bit; ... Lcd_Init();

Tabella 2 – Dettagli della funzione Lcd_Init.

Prototipo	void **Lcd_Out** (char row, char column, char *text)
Ritorno	Nessuno
Descrizione	Stampa un testo sull'LCD partendo da una posizione specificata. Parametri: • Row: posizione nella riga • Column: posizione nella colonna • Text: testo da stampare
Requisiti	Il modulo LCD deve essere inizializzato.
Esempio	Scrittura del testo "Hello!" sulla prima riga a partire dalla terza colonna: Lcd_Out(1, 3, "Hello!");

Tabella 3 – Dettagli della funzione Lcd_Out.

Prototipo	void **Lcd_Out_Cp** (char *text)
Ritorno	Nessuno
Descrizione	Stampa un testo sull'LCD partendo dalla posizione corrente. Parametri: • Text: testo da stampare
Requisiti	Il modulo LCD deve essere inizializzato.
Esempio	Scrittura del testo "Here!" dalla posizione corrente del cursore: Lcd_Out_Cp("Here!");

Tabella 4 – Dettagli della funzione Lcd_Out_Cp.

Prototipo	void **Lcd_Chr** (char row, char column, char out_char)
Ritorno	Nessuno
Descrizione	Stampa un carattere sull'LCD partendo da una posizione specificata. Parametri: • Row: posizione nella riga • Column: posizione nella colonna • Out_char: carattere da stampare
Requisiti	Il modulo LCD deve essere inizializzato.
Esempio	Scrittura del carattere 'i' sulla prima riga e dalla terza colonna: Lcd_Chr (1, 3, 'i');

Tabella 5 – Dettagli della funzione Lcd_Chr.

L'ambiente
MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS

Prototipo	void **Lcd_Chr_Cp** (char out_char)
Ritorno	Nessuno
Descrizione	Stampa un carattere sull'LCD partendo dalla posizione corrente. Parametri: • Out_char: carattere da stampare
Requisiti	Il modulo LCD deve essere inizializzato.
Esempio	Scrittura del carattere 'i' dalla posizione corrente del cursore: Lcd_Chr_Cp('i');

Tabella 6 - Dettagli della funzione Lcd_Chr_Cp.

Prototipo	void **Lcd_Cmd** (char out_char)
Ritorno	Nessuno
Descrizione	Invia un comando all'LCD. Parametri: • Out_char: comando da inviare.
Requisiti	Il modulo LCD deve essere inizializzato.
Esempio	Pulisce il display: Lcd_Cmd(_LCD_CLEAR);

Tabella 7 - Dettagli della funzione Lcd_Cmd

La libreria ha una dipendenza con una libreria di valori costanti denominata Lcd_Constants; questa serve ad effettuare le corrette associazioni tra i pin di interfaccia tra microcontrollore e display, come riportato in Tabella 1. Invece, dalla Tabella 2 alla Tabella 7 sono riportati in forma tabellare i dettagli delle singole funzioni della libreria LCD.
L'elenco dei comandi disponibili nella libreria stessa è riportato nella Tabella 8.

Esempio Pratico: sensore di temperatura

Nell'esempio pratico che vi proponiamo in questa puntata vogliamo mettere insieme un po' di concetti, allo scopo di realizzare qualcosa di più complesso ed anche più vicino alla realtà rispetto a quanto esposto nelle puntate precedenti; ci proponiamo, infatti, di realizzare un termometro che faccia uso di un sensore di temperatura allo stato solido (l'MPC9701A) e che visualizzi la temperatura rilevata su un display LCD alfanumerico.
Come di consueto, utilizzeremo la EasyPIC V7 come scheda di sviluppo, prototipazione e sperimentazione, ma il progetto è facilmente replicabile costruendo un PCB dedicato o anche un prototipo su basetta millefori o breadboard.
Per prima cosa analizziamo la componente software, ossia il progetto MikroC.
Rispetto a quanto visto in precedenza, vogliamo seguire un percorso da utenti più esperti, rispetto a quanto fatto in passato, quindi al termine della

Comando	Descrizione
_LCD_FIRST_ROW	Muove il cursore sulla prima riga.
_LCD_SECOND_ROW	Muove il cursore sulla seconda riga.
_LCD_THIRD_ROW	Muove il cursore sulla terza riga.
_LCD_FOURTH_ROW	Muove il cursore sulla quarta riga.
_LCD_CLEAR	Cancella il display.
_LCD_RETURN_HOME	Riporta il cursore alla posizione 'home' (prima riga prima colonna).
_LCD_CURSOR_OFF	Spegne il cursore.
_LCD_UNDERLINE_ON	Sottolinea il cursore.
_LCD_BLINK_CURSOR_ON	Attiva il blink del cursore.
_LCD_MOVE_CURSOR_LEFT	Muove il cursore di una posizione a sinistra.
_LCD_MOVE_CURSOR_RIGHT	Muove il cursore di una posizione a destra.
_LCD_TURN_ON	Accende il display.
_LCD_TURN_OFF	Spegne il display.
_LCD_SHIFT_LEFT	Slitta il contenuto del display a sinistra di una posizione.
_LCD_SHIFT_RIGHT	Slitta il contenuto del display a destra di una posizione.

Tabella 8 - Elenco dei comandi supportati.

MikroElektronika DEVELOPMENT TOOLS | COMPILERS | BOOKS

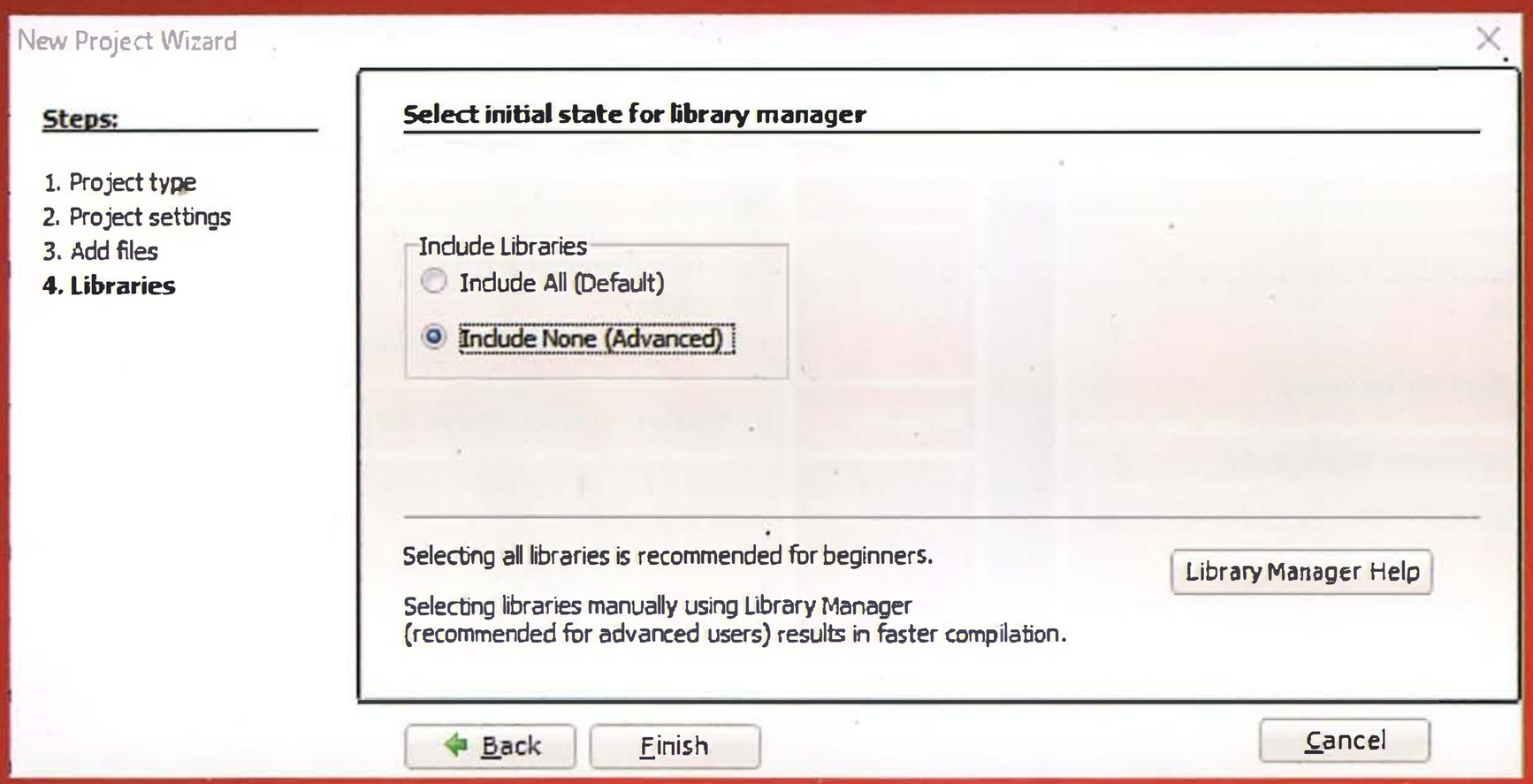


Fig. 6 – Opzione Include None relativamente alle librerie MikroC.

procedura guidata, scegliamo di non importare tutte le librerie (si fa con un clic sul pulsante d'opzione Include None della finestra di dialogo, come mostra la Fig. 6), ma andremo poi a scegliere noi quelle che ci interessano realmente.
Per la realizzazione del progetto che ci proponiamo le librerie che ci interessano sono (considerando anche le dipendenze) cinque:

- ADC;
- Conversions;
- C_String;
- Lcd;
- Lcd_Constants.

La librerie ADC ci occorre per le gestione del convertitore, mentre Conversion (con la sua dipendenza C_String) ci serve per poter convertire il valore numerico letto dall'ADC in una stringa ASCII. Infine La libreria Lcd (con la sua dipendenza Lcd_Constants) ci occorre per controllare il display LCD alfanumerico. In Fig. 7 è riportato il dettaglio delle librerie da selezionare nel library manager.
Prima di passare all'analisi del listato, diamo uno sguardo più da vicino al sensore di temperatura, in modo da capire come interfacciarlo correttamente.
Il sensore di temperatura che abbiamo scelto per questo esempio è l'MCP9701A, prodotto da Microchip Technology; si tratta di un trasduttore di temperatura con uscita analogica, accuratezza di ±2 °C su un range di temperatura di -40 ÷ 125 °C. Questo dispositivo fa parte di una famiglia di trasduttori che comprende anche gli MCP9700/9700A e l'MPC9701. L'MCP9701A presenta un coefficiente di temperatura di 19,5 mV/°C e una tensione di

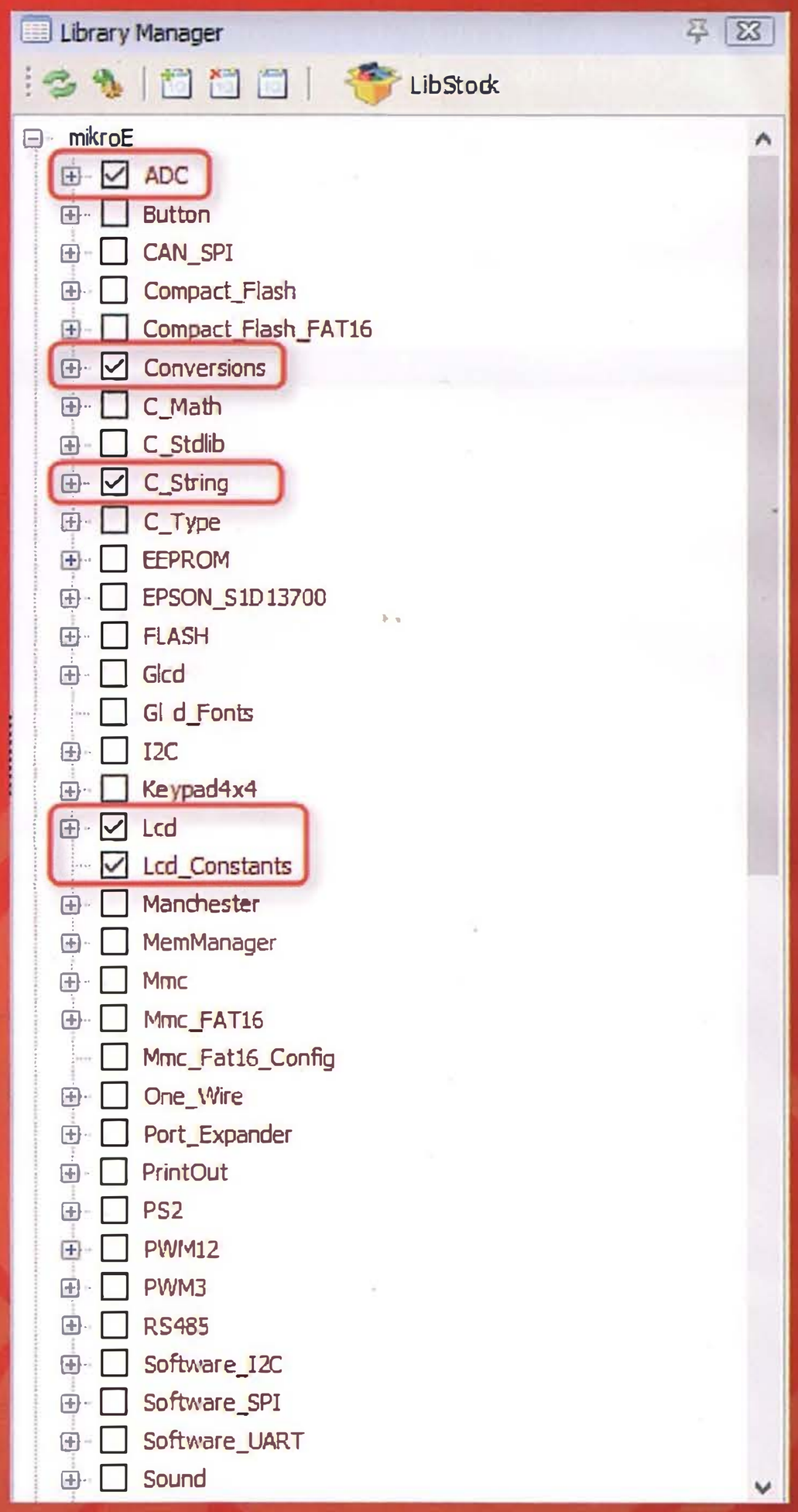


Fig. 7 – Le librerie da selezionare in Library Manager.

MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS

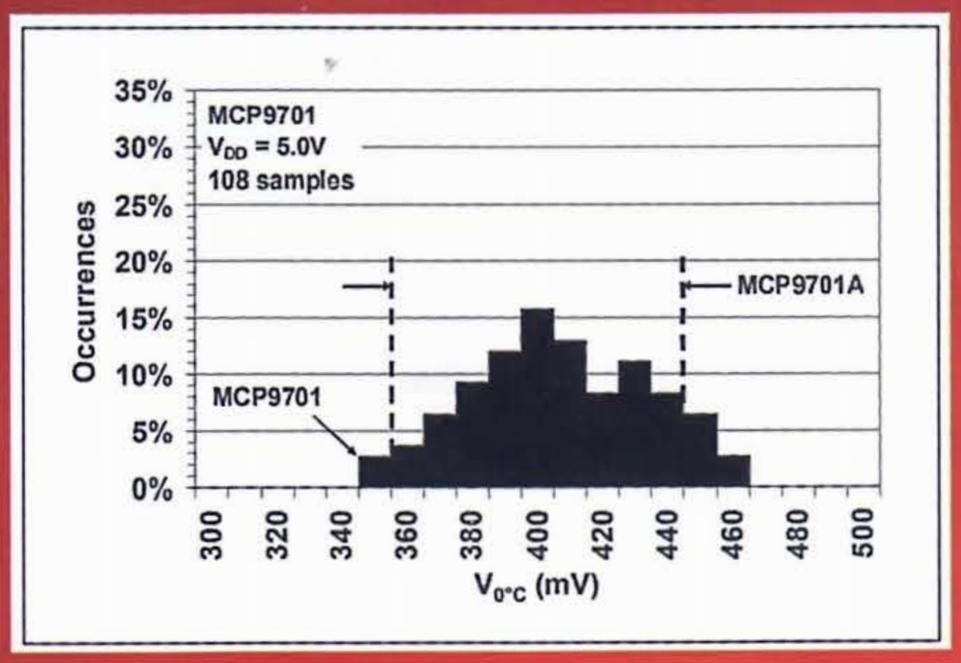


Fig. 8 – Tensione di uscita a 0° C.

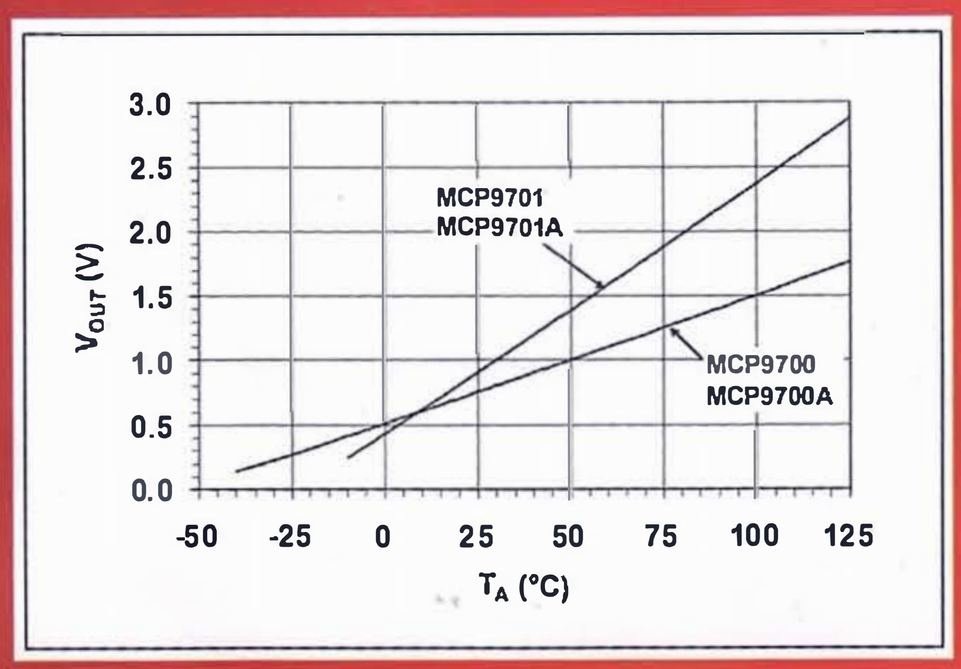


Fig. 9 – Tensione di uscita vs. temperatura ambiente.

uscita, a 0 °C, di circa 500 mV. In Fig. 8 e Fig. 9 sono riportati i grafici relativi alla distribuzione del valore di tensione di uscita a 0° e della curva di tensione di uscita/temperatura ambiente.
La funzione di trasferimento per ottenere la temperatura risulta essere:

$$Ta = \frac{(Vout - Vo°)}{Tc}$$

dove:
TA = temperatura ambiente;
Vout = tensione di uscita del sensore;
Vo° = tensione di uscita a 0°;
TC = coefficiente di temperatura.

In base a quanto riportato nel datasheet e sui grafici stimiamo Vo° pari a 450 mV e Tc pari a 19,5 mV. Dovremo poi trasformare questa formula in codice C sul nostro programma (operazione relativamente semplice che vedremo poi, in dettaglio, durante l'analisi del listato corrispondente).

Il firmware dell'applicazione

Passiamo ora alla descrizione dettagliata del codice del nostro programma, riportato nel Listato 1, in maniera da comprenderlo a fondo. La prima parte del codice associa gli alias definiti nella libreria Lcd_Constants, ai registri dei pin connessi all'LCD. Questo passaggio è assolutamente necessario per fare in modo che la libreria lavori sui registri corretti. Per effettuare le associazioni si usano le

Listato 1

```
/* Lcd module connections */
sbit LCD_RS at LATB4_bit;
sbit LCD_EN at LATB5_bit;
sbit LCD_D4 at LATB0_bit;
sbit LCD_D5 at LATB1_bit;
sbit LCD_D6 at LATB2_bit;
sbit LCD_D7 at LATB3_bit;

sbit LCD_RS_Direction at TRISB4_bit;
sbit LCD_EN_Direction at TRISB5_bit;
sbit LCD_D4_Direction at TRISB0_bit;
sbit LCD_D5_Direction at TRISB1_bit;
sbit LCD_D6_Direction at TRISB2_bit;
sbit LCD_D7_Direction at TRISB3_bit;

/* Constants */
const unsigned short VREF = 5;

/* Variables */
unsigned int TempAdc = 0;
float Temperature = 0;
char TempTxt[15];

void main()
{
  /* Init ADC module */
  ADC_Init();
  /* Init Lcd module, clear display and cursor blink off */
  Lcd_Init();
  Lcd_Cmd(_LCD_CLEAR);
  Lcd_Cmd(_LCD_CURSOR_OFF);

  /* Write first string */
  Lcd_Out(1, 1, "Temperature :");
  /* Display "C" for Celsius */
  Lcd_Chr(2,6,'C');

  while (1)
  {
    /* Get Adc sample from ch6 */
    TempAdc = ADC_Get_Sample(6);
    /* Convert temparature sample */
    Temperature = ((TempAdc * VREF * 0.97) - 450)/19.5;
    /* Convert float to string */
    FloatToStr_FixLen(Temperature, TempTxt, 4);
    /* Write Temperature string */
    Lcd_Out(2,1,TempTxt);
    /* Delay 500ms */
    Delay_ms(500);
  }
}
```

Listato 1 – Il programma TermometroLcd.

MikroElektronika DEVELOPMENT TOOLS | COMPILERS | BOOKS

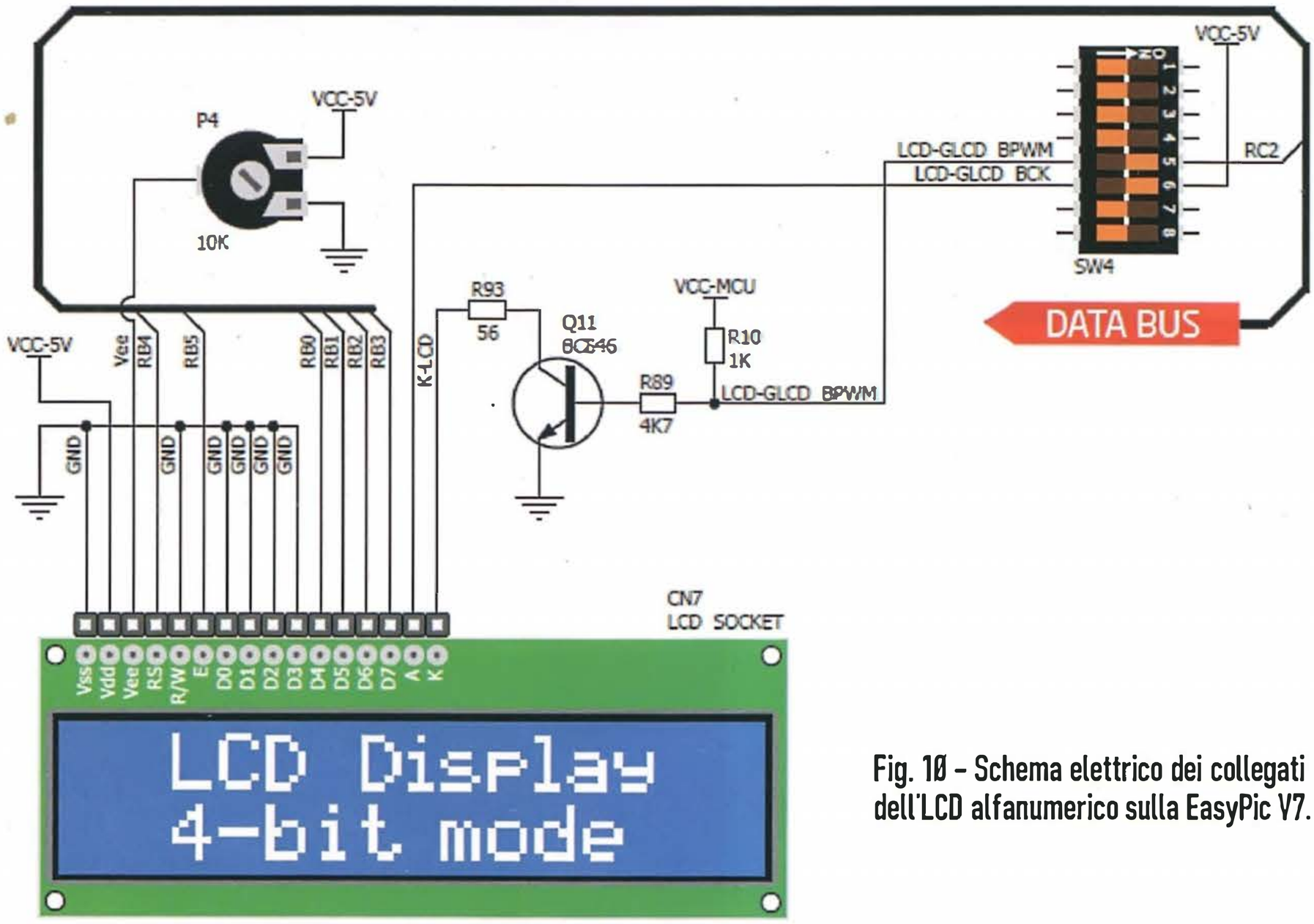


Fig. 10 – Schema elettrico dei collegati dell'LCD alfanumerico sulla EasyPic V7.

keyword sbit ed at:

- sbit: è un tipo di dato che permette di puntare ad uno specifico bit di un registro;
- at: permette di definire un alias ad una variabile o un registro.

Ad esempio, definendo sbit B4 at PORTB.B4 L'alias B0 andrà a puntare esattamente al bit4 del registro PORTB.

Una volta fatte le corrette associazioni per i pin dell'LCD viene definita una costante VREF (che servirà per la formula di conversione della temperatura) e le tre variabili globali:

- TempAdc: lettura del valore numerico in count del canale analogico collegato al sensore di temperatura;
- Temperature: valore numerico in virgola mobile della temperatura;
- TempTxt: array di caratteri per la memorizzazione del valore di temperatura in formato ASCII (per la visualizzazione sul display).

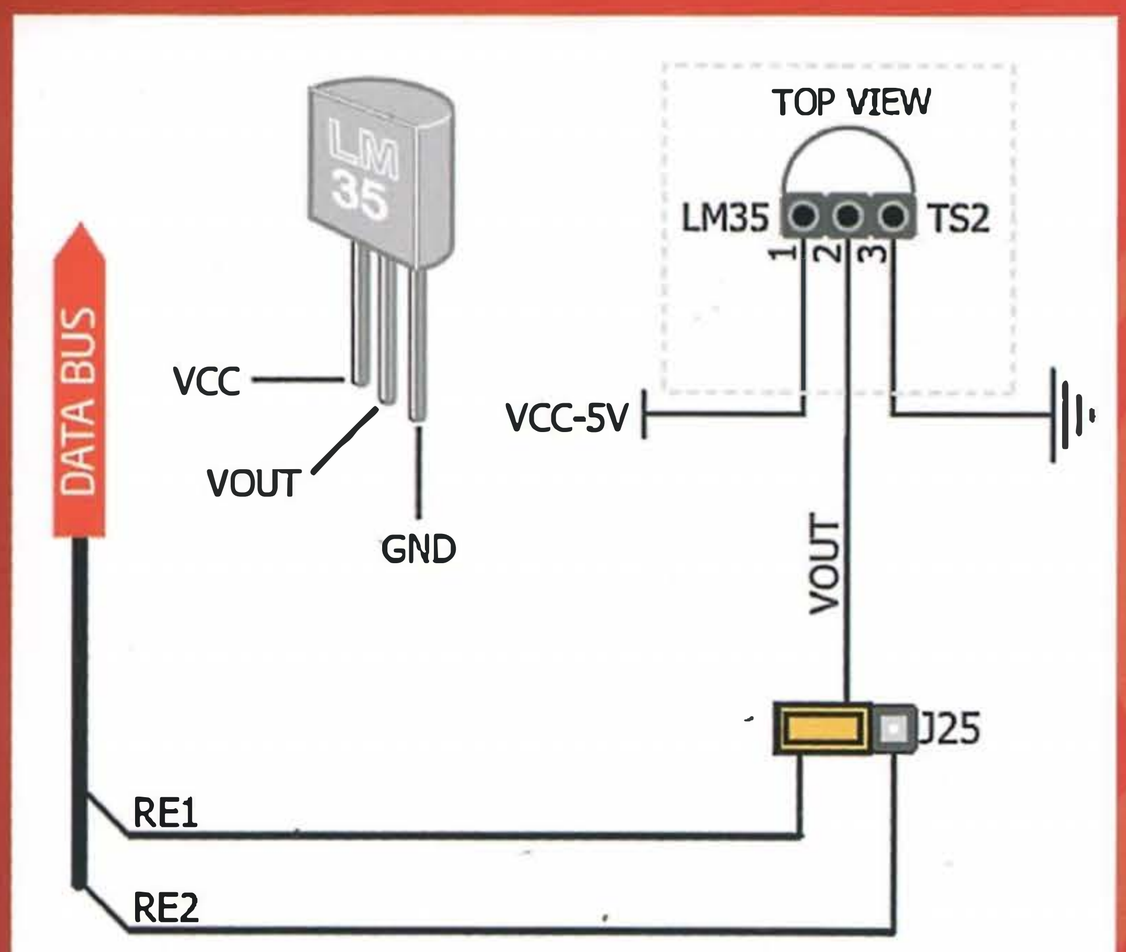


Fig. 11 – Schema elettrico di collegamento del sensore di temperatura alla EasyPic V7.

Dopo aver dichiarato le variabili necessarie, inizia il programma vero e proprio, costituito da una prima fase di inizializzazione e dal solito ciclo infinito. Durante l'inizializzazione viene:

- inizializzato il modulo ADC;
- inizializzato il modulo LCD;
- ripulito il display da eventuali caratteri casuali o residui presenti;
- selezionata l'opzione "Cursor Blink Off", per fare in modo che il cursore non lampeggi;
- visualizzata la stringa "Temperature :" sulla prima riga del display;

MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS

- visualizzato il carattere "C" sulla seconda riga, alla sesta colonna (è l'unità di misura dei gradi Celsius).

Dopo la fase di inizializzazione, nel ciclo infinito viene acquisito il canale 6 del convertitore analogico/digitale, tramite la funzione ADC_ Get_Sample; il risultato dell'acquisizione viene convertito, tramite la seguente istruzione:

```
Temperature = ((TempAdc * VREF * 0.97) - 450)/19.5
```

Come si può vedere, l'istruzione replica esattamente la formula; il calcolo (TempAdc * VREF * 0.97) non è altro che il valore della tensione di uscita del trasduttore in mV.
A questo punto il valore della variabile float Temperature, viene convertito in ASCII usando la funzione della libreria Conversions FloatToStr_ FixLen, che converte un float in ASCII e permette anche di troncare la lunghezza della stringa (in questo caso troncata alla prima cifra decimale). Dopo la conversione, la stringa viene stampata per mezzo della funzione Lcd_Out (già vista in precedenza) e viene applicato un ritardo di 500 ms. Il codice corrispondente è quello riportato nel Listato 1.
Possiamo collaudare il nostro programma per mezzo della EasyPic V7, con l'ausilio di un paio di componenti esterni. In particolare i componenti da reperire sono:

- Sensore di temperatura MCP9701A,
- Display LCD.

Entrambi reperibili sul sito di Futura Elettronica. Il display va posizionato nell'apposito socket della EasyPic V7, mentre il sensore di temperatura va posizionato nel socket denominato LM35, alla sinistra del socket sul quale è alloggiato il micro. In Fig. 10 e Fig. 11 sono riportati gli schemi elettrici delle due sezioni.

Il sensore di temperatura è indicato come LM35 perché questo è uno dei modelli più comunemente utilizzati, ma il socket presente sulla EasyPic V7 è adatto ad ospitare qualsiasi dispositivo compatibile, come appunto l'MCP9701A.
Infine occupiamoci dei settaggi relativi a jumper e dip-switch ed al contrasto dell'LCD. L'LCD necessita del settaggio del contrasto, che può essere fatto tramite il trimmer T4 (evidenziato dalla label "LCD Contrast"); inoltre, se si vuole abilitare la retroilluminazione bisogna agire sul pin BCK del dip-switch SW4, portandolo in posizione '+5V'. Per quanto riguarda il sensore di temperatura l'unico settaggio necessario è relativo al jumper J25, che deve essere chiuso in posizione RE1 (in modo da connettere l'uscita del sensore di temperatura al canale analogico AN6).
A questo punto non resta che programmare il micro, utilizzando il progetto scaricabile sul nostro sito *www.elettronicain.it*. Una volta programmato correttamente il PIC, dovreste ottenere un risultato simile a quello riportato in Fig. 12.

Conclusioni

In questa quinta puntata siamo passati all'analisi dei display alfanumerici ed abbiamo combinato le conoscenze apprese in questa puntata e nelle puntate precedenti per la realizzazione del primo programma non totalmente didattico.
Nella prossima ci occuperemo delle periferiche di comunicazione e di come l'uso di queste ultime è supportato da MikroC.

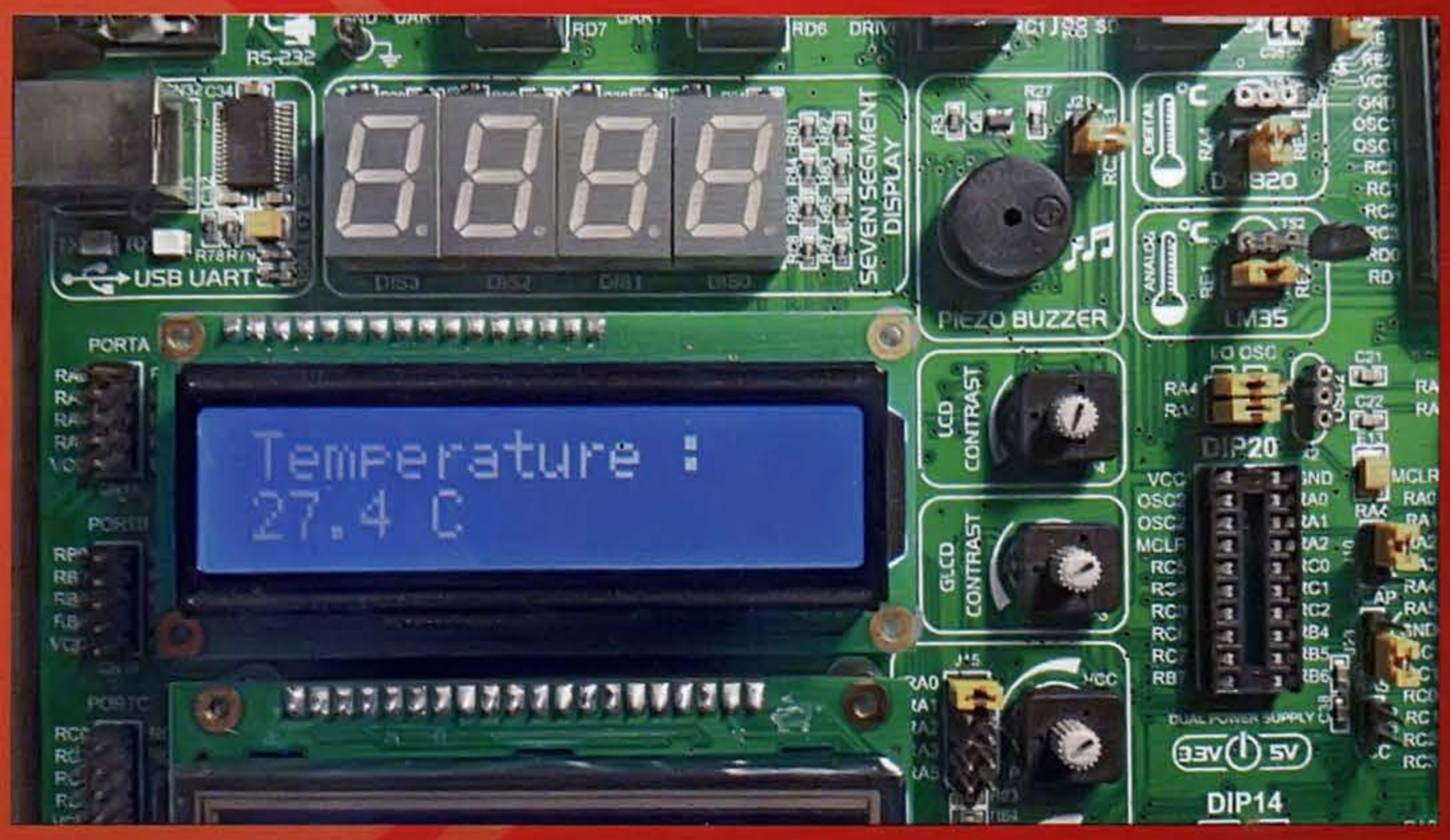


Fig. 12 – Termometro con LCD.

L'ambiente
MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS

ENEL avvia la costruzione del più grande parco eolico del Perù

Enel, attraverso la controllata Enel Green Power Perù ha dato il via ai lavori di costruzione di Wayra I1, il suo primo parco eolico in Perù, situato nel distretto di Marcona, regione di Ica. Il parco, nel quale è in corso l'installazione delle prime turbine eoliche, avrà una capacità totale di 132 MW e, una volta completato, sarà il più grande del Paese.

Il Gruppo Enel investirà circa 165 milioni di dollari USA nella costruzione del nuovo parco eolico, un investimento che si iscrive nel piano strategico del Gruppo. Il parco, che dovrebbe entrare in esercizio nella prima metà del 2018, venderà l'energia generata nel quadro di un contratto di fornitura ventennale firmato con il Ministero dell'energia e delle miniere del Perù.

Il nuovo impianto eolico, che comprende 42 turbine eoliche da oltre 3 MW ciascuna, sarà in grado di generare circa 600 GWh all'anno, pari al consumo annuale di oltre 480.000 famiglie peruviane ed evitando l'emissione in atmosfera di quasi 288.000 tonnellate di CO_2 all'anno. L'energia prodotta dalla centrale verrà immessa nella rete di trasmissione peruviana (SEIN) attraverso la sottostazione di Poroma.

In linea con il modello di Creazione del Valore Condiviso (CSV) adottato dal Gruppo Enel per conciliare lo sviluppo del business con le esigenze delle comunità locali, EGPP ha avviato un piano finalizzato a favorire le opportunità di generazione di reddito nelle aree in prossimità dell'impianto. Il piano, che comprende corsi di formazione imprenditoriale per le donne e supporto per la creazione delle loro imprese, è attuato in coordinamento con le autorità e le comunità locali.



Il Gruppo Enel è presente nel settore peruviano della generazione tramite EGPP (anche impegnata nella costruzione dell'impianto solare da 180 MW di Rubí e l'impianto idroelettrico da 20 MW di Ayanunga), Enel Generación Perú e Enel Generación Piura, che contano su una capacità installata totale di circa 2 GW. Il Gruppo opera anche nel settore della distribuzione del Paese con Enel Distribución Perú, che serve quasi 1,4 milioni di clienti nella regione di Lima.

www.enelgreenpower.com

Anche il camion diventa elettrico

Mentre molti produttori di camion svelano i propri progetti relativi a nuovi modelli a propulsione elettrica, ed anche Tesla annuncia un primo prototipo, è la Cummins, una delle più note e antiche società statunitensi del settore ad anticipare tutti presentando il primo prototipo di motrice alimentato a batterie. Il mezzo si chiama Urban Hauler Tractor, detto anche AEOS.

L'azienda lo definisce come "un trattore a trazione integrale urbana di classe 7". Si tratta di un camion di 18.000 libbre con una capacità massima di carico di 44.000 libbre e con un'autonomia di 100 miglia con una singola carica (160 km) garantita da una pacco batterie da 140 kWh. Un camion, dunque per impiego cittadino e metropolitano, che però con batterie più potenti (questa è solamente la prima versione) o con "range extender" diesel potrebbe essere utilizzato anche per i lunghi tragitti.

Cummins afferma che il veicolo può essere completamente ricaricato in un'ora, e che le consegne inizieranno nel 2019; tutto ciò in attesa di Semi, la motrice di Tesla che dovrebbe avere un'autonomia doppia rispetto al prototipo AEOS.

Continuano intanto anche i test sul fronte della guida autonoma con la start-up Otto (acquisita da Uber) a tirare la volata e con tutti i più grandi produttori, da Scania a Mercedes, a inseguire. E su questo fronte non è escluso che la motrice di Tesla non presenti importanti novità.

www.cummins.com



Acquasmart, la start-up torinese contro l'emergenza idrica

39%: è questa la cifra che indica l'enorme quantità di acqua potabile che gli acquedotti italiani perdono, in media, secondo l'analisi fornita dal Blue Book della Fondazione Utilitatis. Fino al 75% è invece la percentuale della riduzione delle perdite idriche nelle reti urbane proposta dalla soluzione di AquaSmart, la start up torinese creata per gestire in automatico e in remoto le infrastrutture idriche.
Come funziona? Con due componenti principali: il primo è un contatore intelligente brevettato dai ricercatori del Politecnico di Torino, da sostituire agli attuali contatori residenziali e in grado di fornire i dati di pressione e portata della rete idrica in tempo reale; il secondo è un innovativo software gestionale (basato su "machine learning" e reti neurali) che identifica la posizione delle perdite idriche interpretando i dati raccolti dai nuovi contatori e, a differenza di altre soluzioni concorrenti, è stata anche eliminata la necessità di realizzare lunghi e costosi modelli della rete.
Il risultato? I tempi di attivazione sono più brevi e gli investimenti richiesti ai gestori idrici per installare la soluzione sono equiparabili alla spesa odierna per sostituire e gestire gli attuali contatori residenziali.
Garantendo però benefici per tutti.
I gestori idrici diminuiscono le perdite e quindi i costi di gestione.
Non ci sono più prospettive di eventuali razionamenti.
I cittadini pagano solo l'acqua effettivamente consumata e hanno la possibilità di monitorare, tramite una applicazione, i loro consumi in tempo reale.
Al momento AquaSmart è in procinto di effettuare il prossimo passo, ovvero identificare uno o più gestori idrici italiani per testare sul campo il nuovo contatore e il software (raccolta e gestione dati) con l'obiettivo di rendere disponibile la soluzione completa nel 2018.

www.polito.it



Solare fotovoltaico, il GSE pubblica il Rapporto Statistico 2016

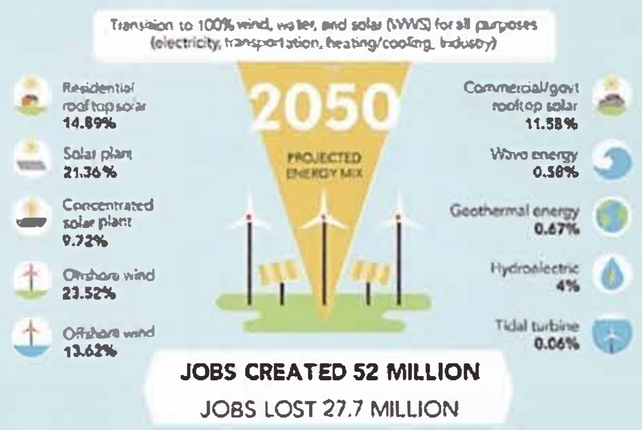
Il Gestore Servizi Energetici (GSE) ha pubblicato il dettagliato Rapporto Statistico sul Solare Fotovoltaico 2016 in Italia. Il documento evidenzia come a fine 2016 risultano in esercizio in Italia 732.053 impianti fotovoltaici, per una potenza installata di 19.283 MW (+382 MW rispetto a fine 2015), che nel 2016 hanno prodotto 22,1 TWh di energia (-0,8 TWh rispetto all'anno precedente).
Gli impianti installati nel corso dell'ultimo anno sono per la grande maggioranza di piccole dimensioni e hanno aderito al meccanismo dello Scambio sul Posto.
Oltre alle analisi sull'evoluzione della potenza installata e dell'energia prodotta a livello nazionale, regionale e provinciale, il Rapporto contiene approfondimenti su vari aspetti tra cui la diffusione del fotovoltaico nei diversi settori di attività, gli autoconsumi, il rapporto tra produzione, autoconsumi e consumi delle utenze per gli impianti in Scambio sul Posto.
Il Documento può essere scaricato gratuitamente dal sito del Gestore.

www.gse.it

100% da fonti rinnovabili nel 2050? L'Italia ce la può fare

Lo evidenzia un studio condotto da 27 ricercatori delle Università di Stanford, Berkeley, Berlino e Aarhus pubblicato nell'ultimo numero della rivista scientifica "Joule" che quantifica anche i benefici, per l'Italia e per gli altri 139 paesi presi in esame.
Lo studio analizza la possibile evoluzione del sistema energetico di 139 paesi sulla base della domanda dei settori trasporti, riscaldamento/raffrescamento, industria, agricoltura, foreste e pesca, giungendo alla conclusione che lo scenario tutto rinnovabili "Wws" ("wind, water and sunlight") è raggiungibile all'80% già nel 2030 e al 100% nel 2050. Rispetto allo scenario "business-as-usual" (Bau), sottolinea lo studio, il Wws permette una riduzione della domanda energetica del 42,5% grazie al maggior tasso di elettrificazione, alle attività di estrazione, trasporto e raffinazione evitate e alla più alta efficienza negli usi finali, con benefici di costi, dell'occupazione e della salute.
Per il nostro paese, i ricercatori stimano un carico di domanda complessivo al 2050 di 240,5 GW nello scenario Bau, derivante per il 33,3% dal settore trasporti, 25,8% dal residenziale, 25,7% dall'industria, 13,5% dal terziario e 1,7% da agricoltura e pesca. Se fosse invece raggiunto il Wws, il carico non supererebbe i 134,9 GW (-43,9% rispetto al Bau), con residenziale al 32,3%, industria (25,5%), trasporti (20,4%), terziario (19,2%) e agricoltura e pesca (2,5%). Il costo dell'energia (Lcoe) in Italia scenderebbe. Tale costo pari a 9,68 cent $/kWh nel 2013, si attesterebbe nel 2050 a 10,80 cent $/kWh nel Bau e a 7,66 cent $/kWh nel Wws. Quest'ultimo scenario permetterebbe un risparmio pro capite di 382 $ l'anno, che sale a 7.733 $ l'anno considerando anche i minori costi climatici (-3.870 $/anno) e sanitari legati all'inquinamento (-3.481 $/anno).
A livello globale, lo scenario Wws permetterebbe di creare 24,3 milioni di posti di lavoro permanenti aggiuntivi e di evitare la morte prematura di 3,5 milioni di persone all'anno entro il 2050. In termini economici si avrebbe un risparmio sui costi dell'inquinamento di 22.800 miliardi $/anno e climatici di 28.500 mld $/anno.

www.iea.org



I sensori e i satelliti aiutano gli agricoltori

Un importante progetto finanziato dall'UE ha realizzato con successo un progetto pilota che ha riunito i dati provenienti da sensori e satelliti al fine di favorire i raccolti, definendo accuratamente i livelli di fertilizzanti necessari.
IOF 2020 (Internet of Food and Farm 2020), finanziato dall'UE, ha l'obiettivo ambizioso di realizzare l'agricoltura di precisione e, grazie a un progetto pilota, sta riuscendo in tale proposito. Il frumento raccolto a Boigneville, 100 km a sud di Parigi, in Francia, ad agosto di quest'anno sarà stato coltivato con i dati forniti da sensori. L'azoto è un problema specifico in agricoltura: l'eccesso di azoto defluisce dal suolo nei corsi d'acqua, per finire poi nei corpi idrici. Ciò può portare all'eutrofizzazione – fioriture di alghe che consumano l'ossigeno, influenzando le risorse ittiche e la biodiversità. Benché le immagini satellitari possano indicare i livelli di azoto nelle colture, i livelli nel suolo non vengono rivelati, ed è qui che entrano in gioco i sensori.
Il progetto pilota sui campi di frumento di IOF 2020 combina le informazioni di sensori sul terreno con i dati satellitari, al fine di indicare agli agricoltori la quantità di azoto o acqua necessaria per i loro campi.

www.iof2020.eu

400 km di autonomia con la nuova Nissan LEAF 100% elettrica

Il 2017 è un anno importante per Nissan, a 70 anni dalla prima auto elettrica: Tama. Era il 1947 e la piccola Tama riusciva già a percorrere fino a 100 km a una velocità massima di 35 km/h. Ma è solo negli anni 2000 che Nissan ha iniziato a puntare fortemente sul progetto Zero Emission.
Fino alla nuova Nissan Leaf presentata recentemente alla stampa, in grado di percorrere quasi 400 km con una ricarica grazie alle batterie agli ioni di litio da 40 kWh ed al nuovo propulsore elettrico che migliora l'efficienza energetica e innalza i valori di potenza e di coppia per prestazioni dinamiche ottimizzate.
Oltre all'efficienza energetica, il nuovo propulsore elettrico offre notevoli prestazioni di guida, erogando una potenza di 110 kW, ossia 150 CV. La coppia è stata innalzata a 320 Nm, con un conseguente miglioramento dell'accelerazione.
Per chi desidera spingersi oltre con la nuova LEAF, Nissan introdurrà una versione con un motore più potente e una maggiore capacità della batteria a partire dalla fine del 2018 e a un prezzo più elevato. Questa versione consentirà una autonomia di percorrenza ancora più estesa ampliando la scelta dei clienti in base alle loro esigenze.
La nuova Nissan LEAF sarà commercializzata in Europa a partire da gennaio 2018.

www.nissan.it



Proteggi la tua casa in modo semplice, efficace e veloce!

Tutto sotto controllo ovunque e in ogni momento!

GSM-GPRS WiFi





€ 219,00
Cod. **FR653**

Prezzi IVA inclusa

Centralina d'allarme wireless WiFi GPRS/GSM

Crea facilmente e velocemente un efficace impianto d'allarme grazie alla nuova centralina antifurto wireless programmabile e controllabile da smartphone. Tramite l'apposita App gratuita per Android e iOS è possibile configurare e gestire da remoto il funzionamento della centralina in ogni suo dettaglio sfruttando la rete GSM-GPRS o Wi-Fi. Consente di monitorare fino a 32 zone e di memorizzare fino ad 8 numeri telefonici a cui inviare SMS e chiamate in caso di allarme; può essere attivata/disattivata tramite radiocomando, chiave TAG RFID o smartphone. La batteria di backup ricaricabile incorporata garantisce il funzionamento anche in assenza di alimentazione principale.



VASTA GAMMA DI ACCESSORI!

Trasmettitore per contatti magnetici



€ 8,00
Cod. FR575MAGN

Presa wireless



€ 19,00
Cod. FR653PRESA

Sensore PIR wireless effetto tenda



€ 16,00
Cod. FR575TENDAPIR

Rilevatore wireless per fughe di gas



€ 22,00
Cod. FR575GAS

Transponder a portachiavi



€ 7,00
Cod. FR575TAG

Sensore PIR wireless



€ 16,00
Cod. FR575PIR

Sirena wireless da esterno



€ 79,00
Cod. FR653WSIR

Radiocomando



€ 8,00
Cod. FR575RAD

FUTURA ELETTRONICA®
Futura Group srl
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775

Prezzi IVA inclusa.

Caratteristiche tecniche e vendita on-line su: www.futurashop.it