

Elettronica In

www.elettronica.in.it

€ 6,00 - Anno XXII - n. 202 - FEBBRAIO 2016

- Driver di potenza per RGBW
- Domotica Open Source con moduli UDOiT
- Red Radio Pitaya
- Termostato a microcontrollore
- STIMA: misurazione di parametri meteo e ambientali

Il cuscino di S. Valentino

Acquario con Fishino



Sensore di gas



- Conosciamo Open Scad
- Il Cebit accelera sulla strada della digitalizzazione

SVILUPPARE APP ANDROID CON Processing



Direttore responsabile:

Arsenio Spadoni
(Arsenio.Spadoni@elettronica.in.it)

Redazione:

Gabriele Daghetta, Paolo Gaspari, Boris Landoni,
Marco Magagnin, Davide Scullino,
Alessandro Sottocornola
(redazione@elettronica.in.it)

Grafica:

Alessia Sfulcini, Filippo Gori
(Alessia.Sfulcini@elettronica.in.it)

Ufficio Pubblicità:

Monica Premoli (0331-752668)
(Monica.Premoli@elettronica.in.it)

Ufficio Abbonamenti:

Elisa Guarniero (0331-752668)
(Elisa.Guarniero@elettronica.in.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
Esteri 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 3-05-1995.

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 2/7/2013.

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.elettronica.in.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.

Futura
Group
Edizioni

© 2016 FUTURA GROUP srl

Intelligenza umana vs intelligenza artificiale

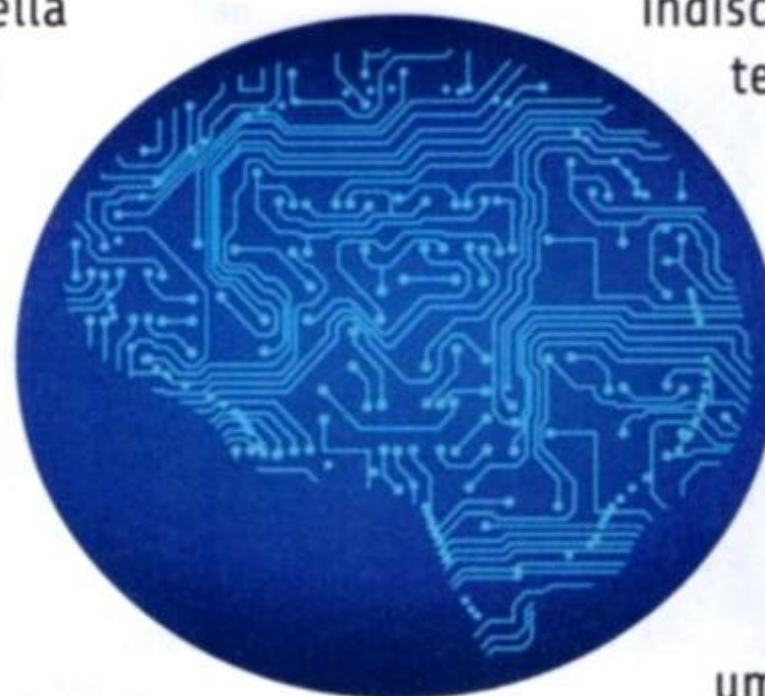
Qualcuno pensava che la prima legge di Moore che afferma che *"la complessità di un microcircuito, misurata ad esempio tramite il numero di transistori per chip, raddoppia ogni 18 mesi"* prima o poi dovesse essere smentita dai fatti. E invece no: la capacità computazionale di dispositivi, sistemi informatici e mezzi di trasmissione dei dati continua ad aumentare a dismisura. Presupposto a quella che viene definita intelligenza artificiale (AI) e che, di concerto, sta compiendo passi da gigante con applicazioni in tantissimi campi, dalle automobili senza conducente, alla diagnostica ed alla ricerca in campo medico. Tutte ricadute sicuramente positive che fanno seguito a quelle altrettanto positive che la scienza e la tecnologia ci hanno portato negli ultimi decenni. In fondo, per la maggior parte degli abitanti della Terra la durata e la qualità della vita, il livello di istruzione e il benessere non sono mai stati così alti. Certo, con una serie di problemi ancora da risolvere (come, ad esempio, l'inquinamento e lo sfruttamento delle risorse del pianeta) ma con un saldo, in ogni caso, sicuramente positivo.

E con prospettive di sviluppo ancora più incoraggianti, grazie proprio a questa inarrestabile avanzata tecnologica.

Ma che per qualcuno potrebbe aprire scenari inquietanti. Nell'ultimo anno infatti, proprio da parte di alcuni dei più convinti sostenitori delle

ricadute positive della tecnologia, da Elon Musk a Bill Gates, sono venuti appelli sui potenziali rischi dello sviluppo senza freni dell'intelligenza artificiale. Pur senza tirare in ballo il computer Hal 9000 di *2001: Odissea nello Spazio*, Musk ha più volte ripetuto che, come nel caso della manipolazione del genoma umano, è necessario una più approfondita conoscenza di quali saranno le conseguenze del dispiegamento indiscriminato di simili

tecnologie. E proprio per questo ha dato vita, insieme a numerosi altri imprenditori della Silicon Valley, all'iniziativa OpenAI con l'obiettivo di far progredire l'AI a beneficio dell'intera umanità. Uno spazio di attenzione e riflessione sulle più avanzate e, potenzialmente pericolose, applicazioni dell'intelligenza artificiale. Avendo ben presente che lo sviluppo tecnologico e la ricerca scientifica hanno un senso solo se vanno a beneficio dell'intera umanità. Un'iniziativa, quella di Elon Musk, che dimostra che, se anche l'intelligenza artificiale diventerà realtà, dovrà sempre fare i conti con l'intelligenza umana.




Arsenio Spadoni

(Arsenio.Spadoni@elettronica.in.it)

Sommario

Febbraio 2016
n. 202



IN COPERTINA:

La nostra proposta tecnologica per la festa degli innamorati, un cuore elettronico luminescente da regalare al tuo partner.

ELENCO INSERZIONISTI

- Area Rebus
- Fiera di Novegro
- Fiera di Scandiano
- Futura Elettronica
- M2M
- Microchip
- National Instruments
- RM Elettronica
- Robot & Makers
- www.mdsrl.it

ARTICOLI

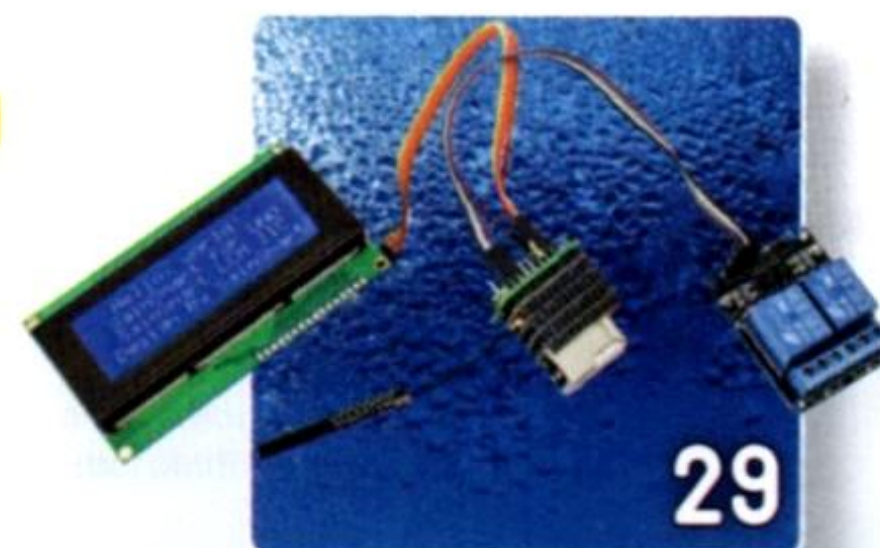
23 DIDATTICA TERMOSTATO A MICROCONTROLORE

Utilizziamo un PIC10F322 in una scheda demo che permette di testarne le funzionalità realizzando, nello specifico, un controllo di temperatura.



29 NETWORKING STIMA: MISURAZIONE DI PARAMETRI METEO E AMBIENTALI

Tre nuovi moduli per l'acquisizione e trasmissione al server RMAP dei parametri rilevati, utilizzabili anche per applicazioni domotiche come, ad esempio, chiudere una tenda da sole in caso di pioggia.



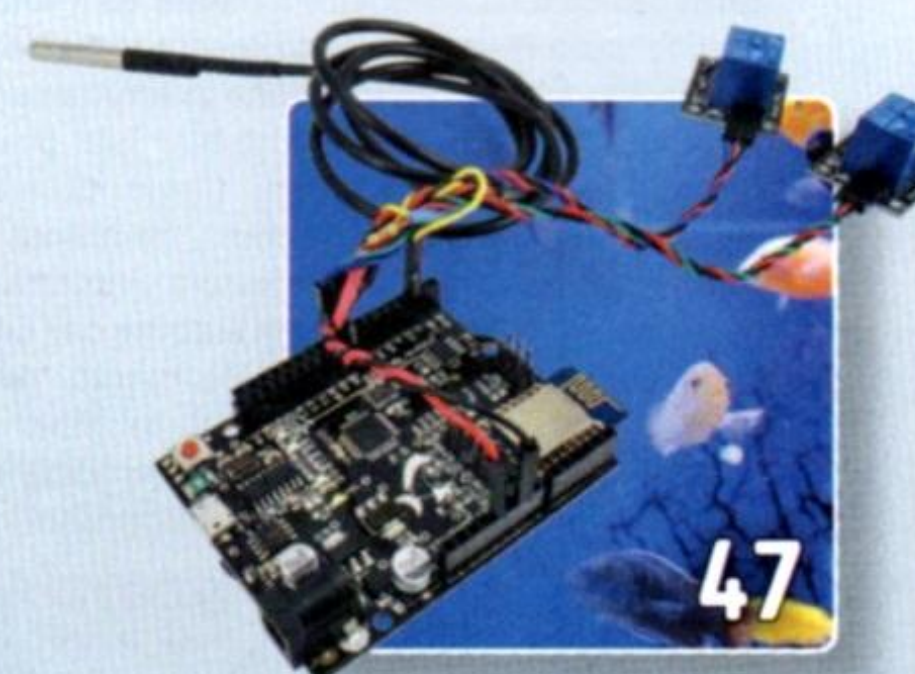
39 APPLICAZIONI RED RADIO PITAYA

Un'insolita applicazione della scheda nata come strumento di misura riconfigurabile, che ci mostra come grazie alla FPGA si può realizzare una Software Defined Radio.



47 ARDUINO L'ACQUARIO CONNESSO

Rendiamo smart e connesso il tradizionale acquario grazie all'impiego di una scheda Fishino.



GADGET

IL CUSCINO DI S. VALENTINO

Ricamiamo un cuscino con un messaggio diretto alla persona amata fatto di filo elettroluminescente

pilotato da una scheda che lo farà accendere creando un effetto suggestivo.



71

63

LIGHTING

DRIVER DI POTENZA PER RGBW

Permette di pilotare i nuovissimi diodi luminosi a 4 colori, variandone facilmente la luminosità e la colorazione.



63

83

EVENTI

IL CEBIT ACCELERA SULLA STRADA DELLA DIGITALIZZAZIONE

Il tema conduttore della prossima edizione del CeBIT in programma ad Hannover dal 14 al 18 marzo 2016 sancisce la definitiva trasformazione della manifestazione tedesca nel più importante evento mondiale dedicato alla digitalizzazione.



83

89

DIDATTICA

CONOSCIAMO OPENS CAD

Concludiamo la panoramica sul software libero di modellazione ed editing tridimensionale con la creazione di modelli divisi in parti da assemblare. Seconda ed ultima puntata.



89

La tiratura di questo numero è stata di 18.500 copie.



Mensile associato all'USPI, Unione Stampa Periodica Italiana

99

LABORATORIO

SENSORE DI GAS

Aggiungiamo, con questa piccola breakout board, un efficiente rivelatore di gas a sistemi di allarme generici, tecnologici e antifurto.



99

RUBRICHE

- 01 EDITORIALE
- 05 ELETTRONICA INSIEME
- 12 EVENTI
- 15 APPUNTAMENTI
- 19 COMPONENTI & SISTEMI
- 114 SCIENZA & TECNOLOGIA
- 125 RINNOVABILI

107

DOMOTICA

DOMOTICA OPEN SOURCE CON I MODULI UD0IT

Dotiamo il gateway di UD0IT, basato su Raspberry Pi, di un display touch screen e di un contenitore adatto a fargli fare bella figura in qualunque ambiente.



107

117

DIDATTICA

SVILUPPARE APP ANDROID CON PROCESSING

Impariamo a scambiare dati tra uno smartphone e Arduino, utilizzando la connessione Bluetooth. Quarta puntata.



117

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@elettronica.in.it

Ricarica wireless long range

Possiedo uno smartphone Samsung S6 Edge+ che ricarico in modalità wireless appoggiando il telefono sull'apposito charger. Volevo sapere se è possibile aumentare la portata di questo sistema, magari a qualche metro di distanza. Ho letto infatti che esistono dei sistemi di ricarica wireless che funzionano all'interno di una stanza, semplicemente tenendo il cellulare in tasca. Mi potete dare qualche suggerimento?

Marco Riva - Roma

Il tuo telefono utilizza una tecnologia che, dopo alcuni anni di esperimenti e test, possiamo considerare già consolidata. In questo caso il trasferimento di energia avviene mediante induzione elettromagnetica utilizzando due bobine, secondo lo standard Qi

definito dal Wireless Power Consortium (WPC). Il principio di funzionamento è simile a quello di un comune trasformatore con un primario che trasferisce per induzione l'energia su un avvolgimento secondario. In questo caso, tuttavia, per ottenere un

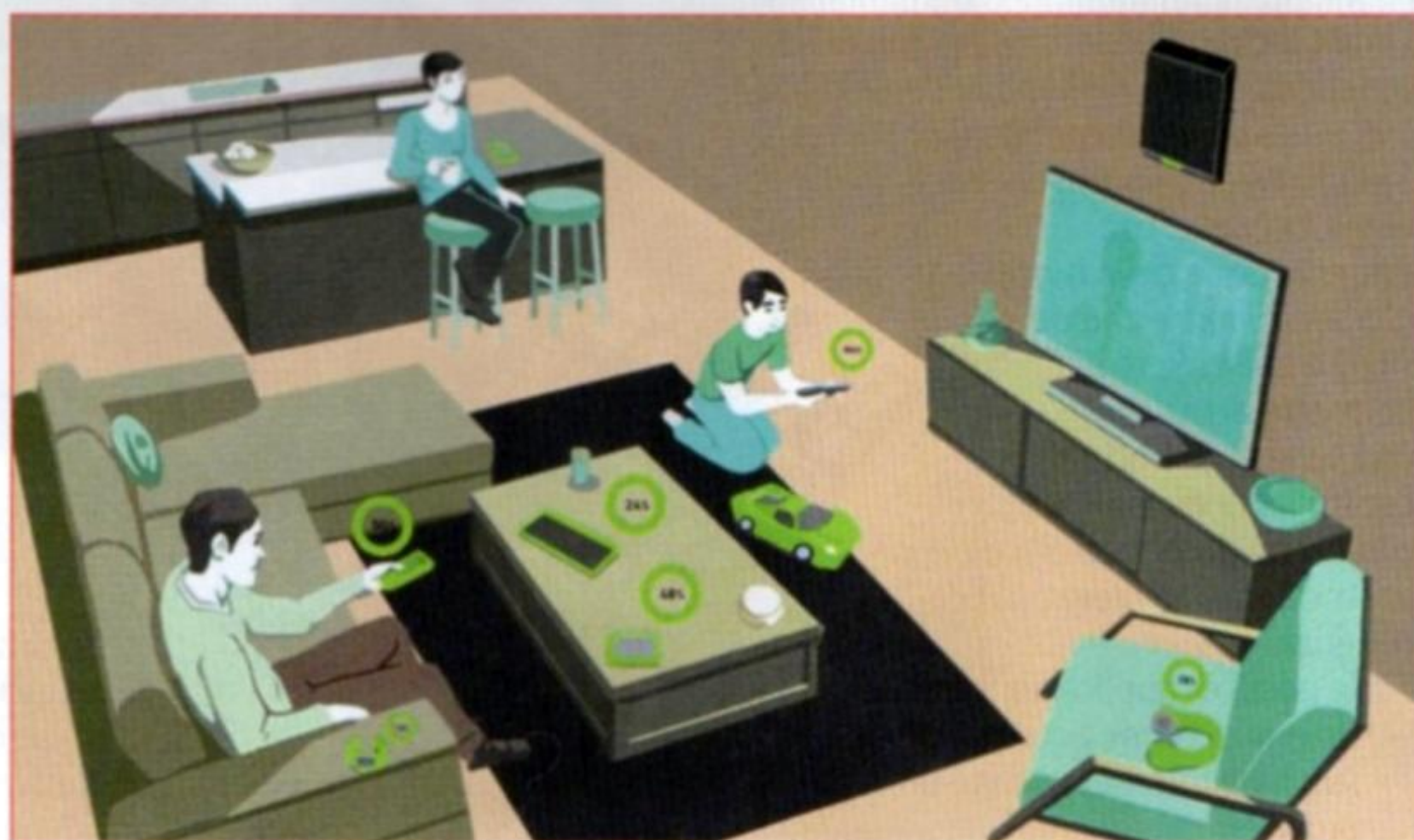
elevato rendimento, vengono utilizzate frequenze molto più alte rispetto ai 50 Hz di rete. Come avrai potuto constatare, l'efficienza è decisamente buona e il telefono si ricarica in un paio d'ore circa. Inoltre, il consumo a vuoto del Charger è irrisorio e le radiazioni elettromagnetiche emesse, al di là dell'area di lavoro, sono bassissime. In più, il sistema Qi prevede un canale di comunicazione tra il Charger e lo smartphone per ottimizzare i parametri di ricarica. Per tutti questi motivi non è in alcun modo possibile modificare il dispositivo per aumentare la distanza di funzionamento.

I sistemi a cui ti riferisci utilizzano la tecnica della risonanza magnetica, con frequenze di funzionamento molto più elevate e particolari tecniche di localizzazione del dispositivo da ricaricare. Siamo ancora nel campo della sperimentazione anche se Ener-

gous, una delle società più attive in questo settore, è già quotata in Borsa. WattUp, la tecnologia di Energous, si basa su trasmettitori RF che generano fasci di onde radio sulla banda dei 5,8 GHz, una sorta di router che irradiano energia anziché dati. Un ricevitore all'interno dello smartphone sintonizzato sulla stessa frequenza converte il segnale RF in energia elettrica che ricarica la batteria.

Il tutto viene controllato da un software che "gira" in background su ciascun telefonino e che ottimizza le prestazioni di ricarica. La portata del sistema (con i trasmettitori RF che possono essere inseriti dentro elettrodomestici o accessori d'arredamento) è di circa 4,6 metri (15 piedi) con la possibilità per ciascun trasmettitore di gestire fino a 12 smartphone. Il sistema è in grado di trasferire una discreta quantità di energia (4 W alla distanza di 1,5 metri e 1 W a 4,5 metri) con un'efficienza massima del 25%.

Questo ed altri sistemi simili, tuttavia, sollevano importanti interrogativi riguardanti la sicurezza: anche se le radiazioni non ionizzanti non sono (teoricamente) pericolose per il corpo umano, che bisogno c'è di aumentare il livello di inquinamento elettromagnetico delle nostre case e dei nostri uffici quando - in ogni caso - esiste una facile alternativa per ricaricare i nostri smartphone e i nostri dispositivi portatili?



Oltre la nuvola c'è anche la ... nebbia

Alla parola cloud e al suo significato nel settore informatico sono già abituato. Tuttavia sento sempre più spesso l'espressione Fog e Fog Computing. Cosa si intende esattamente con questi termini?

Lorenzo Triggiano - Bari

Se coi termini Cloud e Cloud Computing si intende quell'insieme di risorse di archiviazione e elaborazione disponibili on demand e fruibili via Internet, gratuitamente o a pagamento, il termine Fog (nebbia) Computing indica un livello intermedio tra l'utilizzatore e il cloud, in grado di fornire anch'esso capacità di elaborazione e archiviazione ma con una latenza molto più bassa, quasi in modalità real-time. Una tecnologia di particolare importanza per le reti IoT e un paradigma che si sta affermando nelle piattaforme tecnologiche



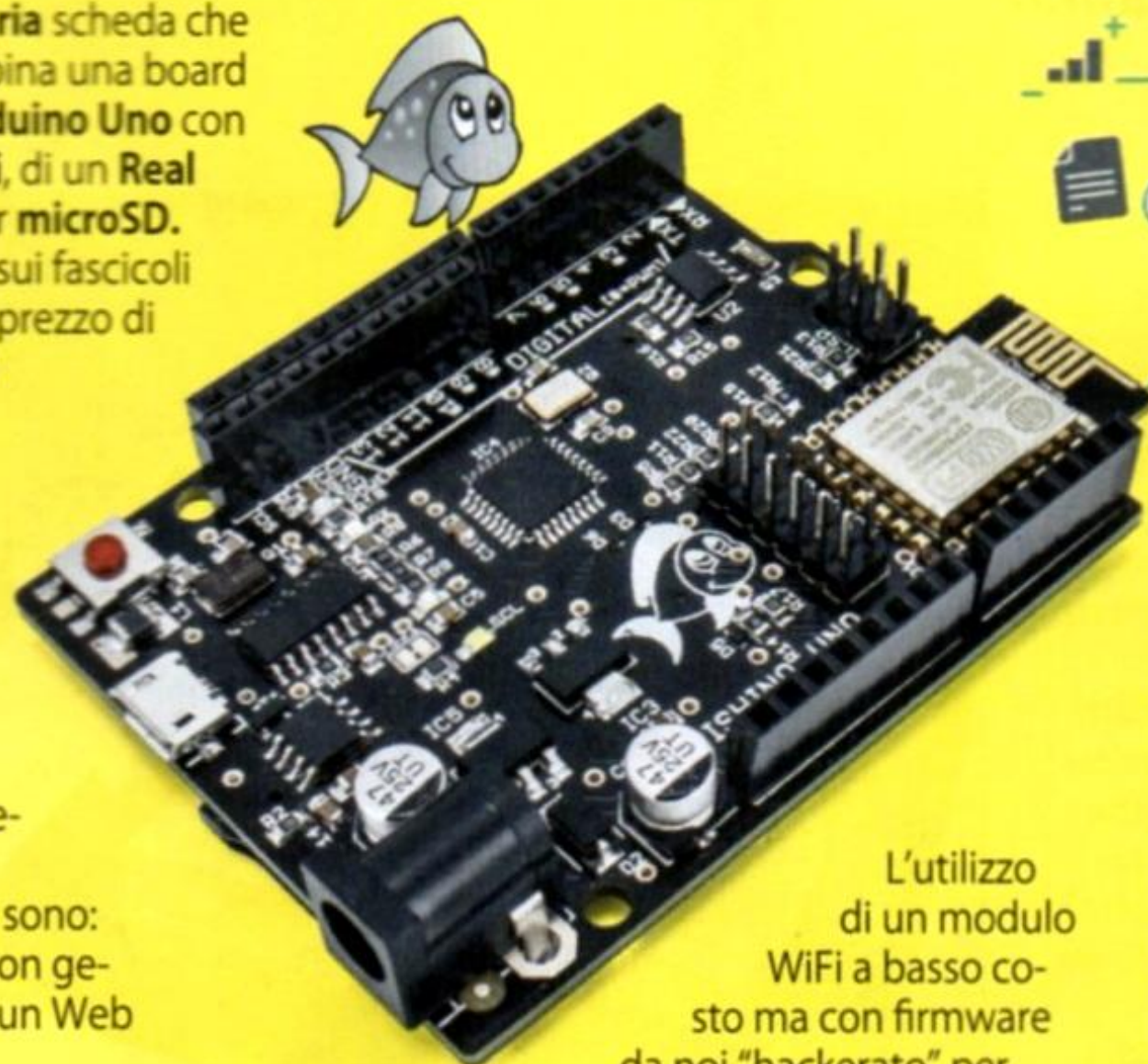
PROGETTA & VINCI CON FISHINO! TRE OSCILLOSCOPI IN PALIO

Fishino UNO è la rivoluzionaria scheda che tutti aspettavano e che combina una board compatibile al 100% con Arduino Uno con l'aggiunta di un modulo WiFi, di un Real Time Clock e di un lettore per microSD. La scheda è stata presentata sui fascicoli 198 e 199 ed è disponibile al prezzo di 36 Euro IVA compresa presso www.futurashop.it.

L'integrazione della funzionalità WiFi (oltre che dell'RTD e della microSD) permette di realizzare una serie di dispositivi controllabili via Internet o in grado di connettersi e trasmettere dati rilevati in precedenza.

Tra le possibili applicazioni ci sono:

- sistemi di home automation gestibili via Internet tramite un Web Browser;
- data logger portatili in grado di connettersi e scaricare i dati sulla rete quando si entra nel campo di copertura di una rete WiFi;
- robot controllabili via rete e in grado di trasmettere tramite essa i dati rilevati da sensori.

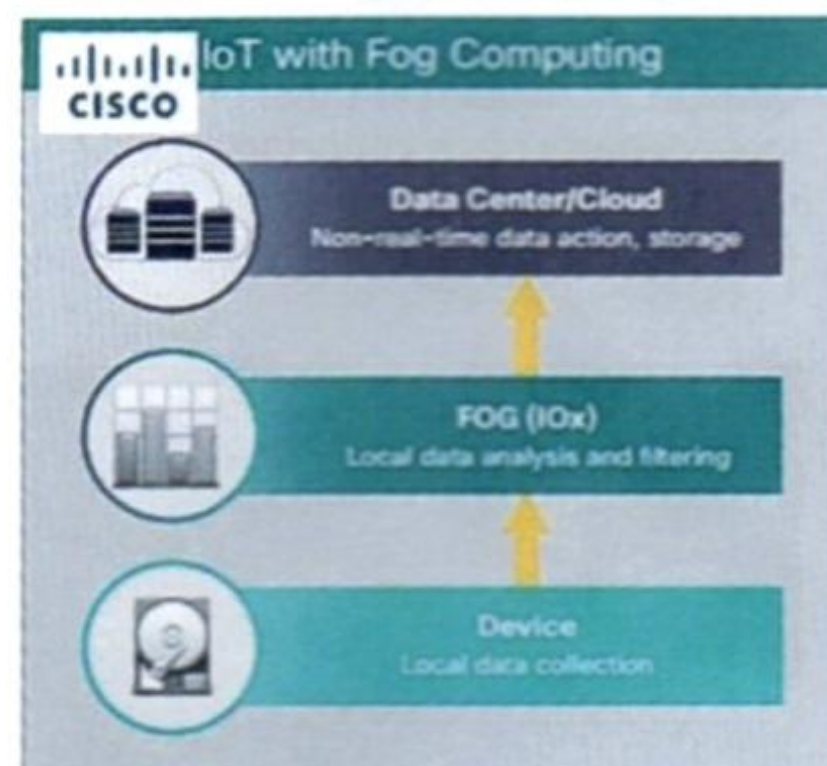


L'utilizzo di un modulo WiFi a basso costo ma con firmware da noi "hackerato" per ottenere prestazioni elevate e capace di funzionare anche come access point, ovvero senza la necessità di una struttura di rete WiFi esistente, permette il controllo via cellulare in qualsiasi momento, anche in assenza di copertura di rete, rendendo il dispositivo sempre interattivo.



La possibilità di eseguire l'aggiornamento degli sketch via Internet, già prevista a livello hardware, permetterà inoltre di avere dispositivi sempre aggiornati senza la necessità di doverli collegare fisicamente ad un computer. Per incentivare l'uso di questa board e proporre alla nostra community applicazioni originali, abbiamo dato vita a questa iniziativa che **PREMIA CON TRE OSCILLOSCOPI**, le migliori applicazioni basate sulla board Fishino. Tutti i progetti verranno presentati on-line mentre quelli che si classificheranno ai primi tre posti verranno pubblicati anche sulla rivista. Tutte le informazioni relative a questa iniziativa sono disponibili alla pagina: www.elettronica.in/fishinocontest. La scadenza di questo concorso è fissata al **31 marzo 2016**. Affrettatevi!

più avanzate, in primis nell'Industry 4.0. Tra i principali sostenitori del Fog Computing c'è il colosso del networking Cisco: secondo la società di San Jose non è pensabile che i miliardi di oggetti che tra qualche anno saranno connessi in rete dialoghino direttamente con i grandi centri di calcolo e storage del cloud: c'è bisogno di un livello intermedio, in grado di interpretare linguaggi differenti, di eseguire applicazioni in tempo reale, o



quasi, di ridurre la latenza al minimo; c'è bisogno di un livello molto più vicino ai sensori e agli attuatori, un livello in grado di gestire al meglio sistemi complessi quali, ad esempio, il collegamento tra automobili, semafori intelligenti, reti elettriche e molto altro ancora.

Un paladino del Fog Computing è anche l'ex vice presidente di Cisco, l'italiano Flavio Bonomi, che ha fondato Nebbiolo Technologies, un nome che tradisce sia le origini del suo fondatore che il core business della società, il Fog Computing, appunto.

Nebbiolo Technologies ha come obiettivo quello di integrare le migliori tecnologie di calcolo e di storage, da collocare nelle immediate vicinanze di sistemi IoT e IIoT (Industrial Internet of Things), in grado di garantire una bassa latenza, sempre più necessaria nelle piattaforme complesse. Ad esempio, un'autovettura autonoma e connessa che si muove nel traffico cittadino ha bisogno di conoscere in tempo reale la posizione degli altri veicoli, lo stato dei semafori, la presenza di un eventuale ostacolo sulla strada; allo stesso modo

un braccio robotico deve essere in grado di muoversi in perfetto sincronismo con tutti gli altri sistemi automatizzati e deve reagire in tempo reale ad eventi non programmati. Per tutte queste attività la gestione deve essere affidata a un sistema molto più vicino ai sensori e ai dispositivi coinvolti, riprogrammabile, in grado di eseguire differenti applicazioni e, naturalmente, essere connesso anche al cloud. Cosa delegare al primo livello (fog) o al livello superiore (cloud) in termini di elaborazione, archiviazione e interfaccia utente, dipende dai processi coinvolti e dalla tipologia dei dispositivi connessi, ma proprio la bassa latenza e la capacità di operare in tempo reale, hanno attirato l'attenzione su Nebbiolo Technologies di alcuni player globali nel campo dell'automazione industriale come KUKA e TTTech che hanno investito nella società di Flavio Bonomi.

Recentemente è stato anche fondato l'Open Fog Consortium (<http://www.openfogconsortium.org>) con l'intento di promuovere questa architettura sia in ambito industriale che in quello consumer.

OSCILLOSCOPIO 2 CANALI 200 MHz CON GENERATORE DI FUNZIONI

- ✓ Frequenza di campionamento fino a 1 GS/s
- ✓ Display LCD da 7" 64.000 colori 800x480 pixel
- ✓ 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

1° PREMIO



con Labs è l'ideale sotto tutti i punti di vista. Come vedi nello schema, tutti gli elementi di un completo ricevitore stereo FM sono integrati all'interno del piccolo chip di silicio, il quale, per funzionare, non ha bisogno di alcun componente esterno, tantomeno di bobine o trasformatori strani. L'unico componente esterno, a parte i condensatori di disaccoppiamento, è il quarzo da 32,768 MHz.

Naturalmente il circuito – completamente digitale – va opportunamente programmato utilizzando l'interfaccia di controllo. L'antenna va collegata al pin 2, le uscite audio sono disponibili sui pin 13 e 14 mentre il dato digitale dell'RDS è disponibile sul pin 15.

Il circuito funziona con una tensione di alimentazione compresa tra 2,7 e 5,5 V e consuma appena 10 mA. Particolarmente ridotte le dimensioni: appena 3x3 millimetri.



2° PREMIO

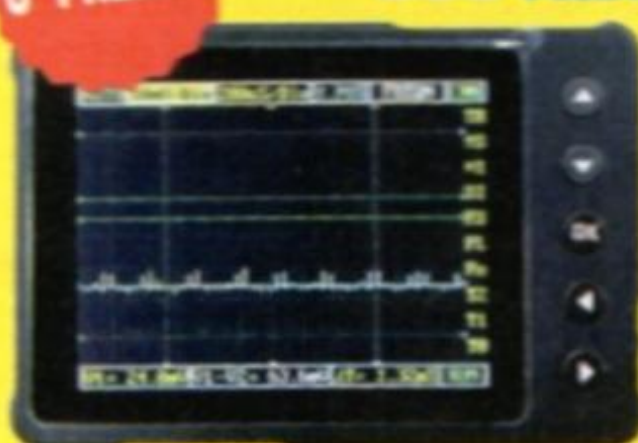
OSCILLOSCOPIO 2 CANALI 70 MHz

- ✓ Frequenza di campionamento fino a 1 GS/s
- ✓ Display LCD da 7" 64.000 colori 800x480 pixel
- ✓ Funzione FFT integrata
- ✓ Software per l'analisi in tempo reale con il PC
- ✓ Porta USB Host

OSCILLOSCOPIO TASCABILE MONOCANALE CON MEMORIA - 1MHz

- ✓ Frequenza di campionamento 1 MS/s
- ✓ Slot per micro SD Card con possibilità di salvare le forme d'onda
- ✓ Generatore di segnale da 10 Hz a 1 MHz
- ✓ Display LCD TFT 2,8" a colori
- ✓ Connessione USB e batteria ricaricabile al litio
- ✓ Completo di puntali ad uncino e a singolo terminale

3° PREMIO



DSO Nano

Progetti e DMX

Sto realizzando un impianto di illuminazione piuttosto complesso che vorrei controllare con lo standard DMX che, ho visto essere ancora oggi una dei più utilizzati in questo campo. Dove posso trovare maggiori informazioni su questa tecnologia, sui componenti e su circuiti d'esempio?

Renato Galimberti - Milano

In passato ci siamo occupati più volte della tecnologia DMX, sia dal punto di vista teorico che da quello pratico, proponendo numerosi progetti.

Come nostra abitudine nel caso di articoli e progetti che hanno come filo conduttore la medesima tecnologia, prepariamo una raccolta con articoli e progetti che rendiamo disponibile gratuitamente on-

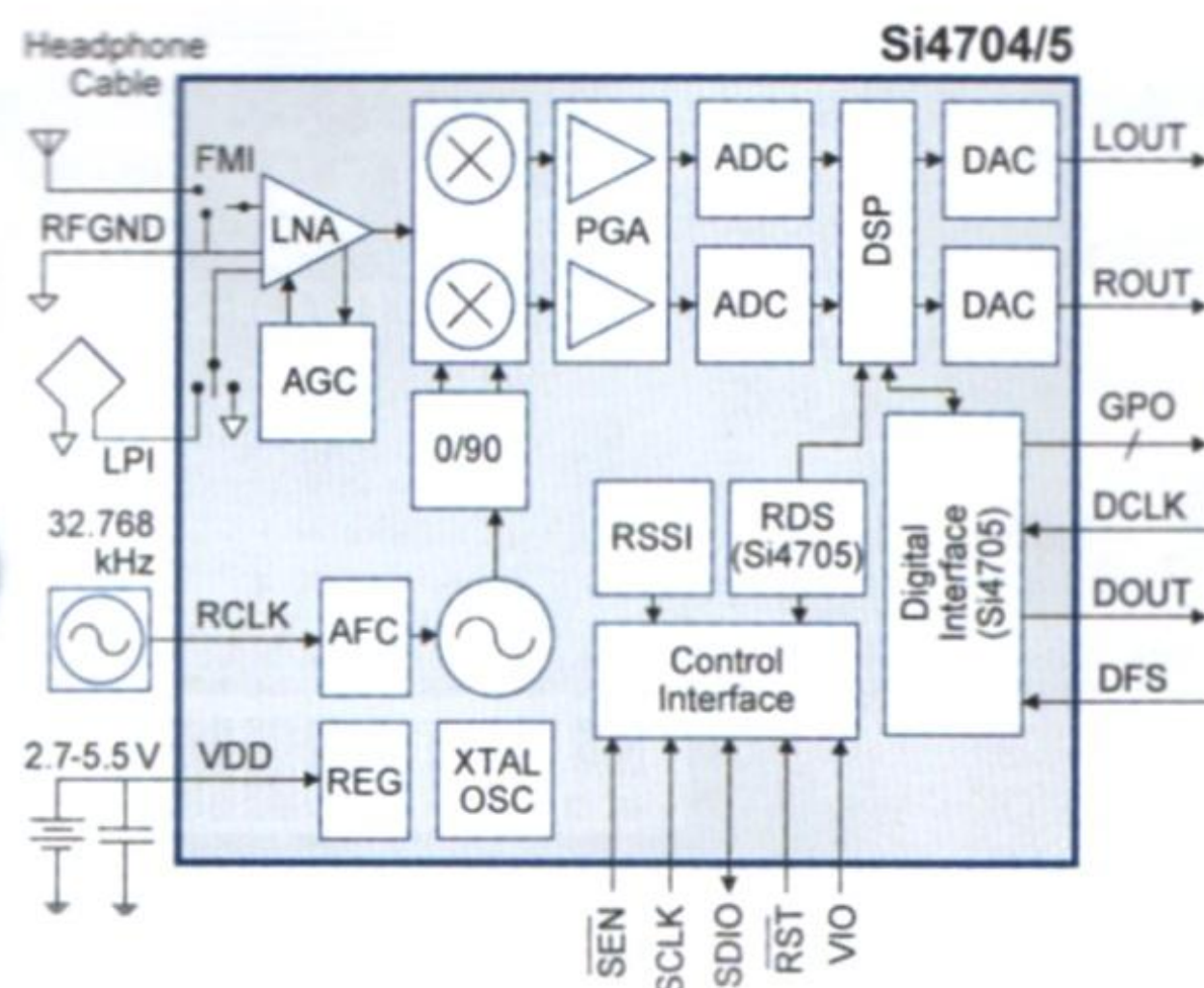
Chip FM con RDS

Vorrei realizzare un ricevitore per la banda FM che mi fornisca in uscita anche il dato dell'RDS, ovvero le informazioni sull'emittente,

sul tipo di brano, eccetera. Ad oggi non sono riuscito a trovare qualcosa di semplice, alla portata della mia scarsa conoscenza dell'elettronica. Mi potete dare qualche suggerimento?

Martino Gaggiolo - Torino

Sicuramente il dispositivo Si4705 di Sili-





line in formato digitale (basta registrarsi al sito per poter scaricare il contenuto che ti interessa). In questo caso trovi la monografia dedicata al DMX selezionando sulla side-bar di sinistra del nostro sito www.elettronica.in.it la voce "monografie" e, all'interno di questa, la raccolta dedicata al DMX. Nel file .zip da scaricare troverai, oltre a tutti gli articoli, anche le tracce dei circuiti stampati e il firmware utilizzato nei vari progetti. Anche tutte le altre monografie, così come i corsi, sono scaricabili gratuitamente in formato digitale.

La profondità di memoria degli oscilloscopi digitali

Ho trovato recentemente tra le caratteristiche di un oscilloscopio digitale la definizione "profondità di memoria". Di cosa si tratta esattamente?

Mario Romiti - Bologna

Tutti noi conosciamo caratteristiche come la banda passante, che rappresenta la gamma di frequenze che l'oscilloscopio è in grado di visualizzare, o la sensibilità verticale, che indica la più piccola variazione del segnale di ingresso che lo strumento è in grado di apprezzare. Con l'introduzione degli oscilloscopi digitali DSO (Digital Storage Oscilloscopes) e degli MSO (Mixed Signal Oscilloscopes), a

queste caratteristiche se ne sono aggiunte altre, dovute all'utilizzo di un convertitore digitale A/D d'ingresso mediante il quale il segnale d'ingresso, appunto, viene convertito in un dato digitale che viene successivamente elaborato e visualizzato sul display. Per inciso, la digitalizzazione del segnale analogico viene effettuata sia dagli oscilloscopi digitali veri e propri (DSO) che dagli oscilloscopi a segnali misti MSO i quali, in aggiunta ai normali canali di ingresso analogici con convertitore A/D, dispongono di un certo numero di ingressi digitali in grado di acquisire livelli logici in modo simile a quanto avviene in un analizzatore logico. Questi oscilloscopi consentono dunque di visualizzare contemporaneamente sul display l'andamento di segnali analogici e digitali. Un moderno oscilloscopio a segnali misti è dunque in grado di analizzare indifferentemente le forme d'onda analogiche, i segnali digitali e i protocolli di trasmissione.

Tornando al nostro convertitore A/D d'ingresso, come in tutti i circuiti di questo tipo, la misura dell'ampiezza del segnale d'ingresso viene effettuata *tot* volte in un secondo. Questo valore si definisce frequenza di campionamento e, di solito, è quattro volte superiore alla banda passante dell'oscilloscopio. Ciò significa che se la banda è di 100 MHz, la frequenza di campionamento sarà dell'ordine di 400 Msample al secondo, garantendo 4 campionamenti del segnale alla frequenza più elevata.

I dati digitali così ottenuti vengono memorizzati all'interno della memoria di acquisizione dell'oscilloscopio che deve essere in grado di memorizzare velocemente i dati campionati (con una velocità pari perlomeno alla massima velocità di campionamento) nonché avere dimensioni elevate in termini di numero di campioni in grado di memorizzare. Infatti, una grande memoria consente di visualizzare



una porzione di segnale più lunga, a parità di frequenza di campionamento. In alternativa, a parità di tempo di visualizzazione, una memoria più estesa consente di usare una maggiore frequenza di campionamento. A titolo di esempio, se un oscilloscopio digitale presenta una massima frequenza di campionamento pari a $f_c = 1 \text{ Gsample/s}$ e la memoria è in grado di contenere $N = 100.000$ punti, la massima velocità del convertitore può essere sfruttata solo per intervalli di osservazione non superiori a $N/f_c = 10^5 / 10^9 = 100 \mu\text{s}$, mentre per acquisire intervalli di durata maggiore si dovrà ridurre la frequenza di campionamento. La profondità di memoria rappresenta dunque la capacità di uno specifico oscilloscopio digitale di memorizzare (e poi visualizzare) il massimo intervallo di segnale, ed è strettamente correlata, come abbiamo visto, alla frequenza di campionamento.

Lampade UVC α LED

Possiedo un vecchio cancellatore di EEPROM che utilizzo per riprogrammare macchinari un po' obsoleti ma ancora efficienti. A seguito di una caduta, ho rotto la lampada montata all'interno che vorrei sostituire con una versione a LED. Esistono già lampade allo stato solido per questa applicazione?

Sergio Spina - Venezia

RM ELETTRONICA

Forniture per hobbisti, scuole e industria

Distributore autorizzato:

**FUTURA
ELETTRONICA**

**ELSE
Kit**

**NUOVA
ELETTRONICA**

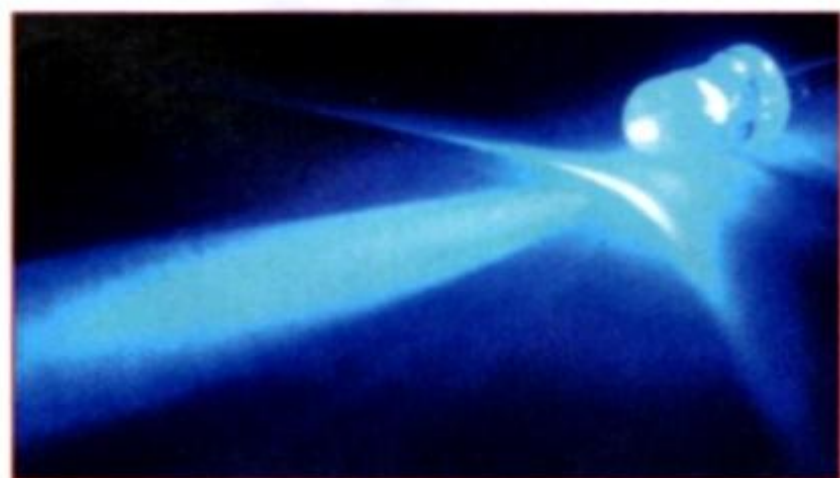
**Il tuo punto
di riferimento
per l'elettronica
a ROMA**

Via Val Sillaro 38 • 00141 Roma • Tel: 06-8104753

La lampada che hai rotto è una comunissima lampada fluorescente ultravioletta (UV-C) con lunghezza d'onda di 253,7 nm che costa 10-15 Euro. Queste lampade producono una radiazione UV-C a banda larga con potere germicida e per questo motivo sono impiegate anche per sterilizzare luoghi di lavoro e strumenti usati nei laboratori di biologia e apparecchiature mediche. Nel caso dei cancellatori di EEPROM la luce emessa ha un'energia sufficiente per permettere agli elettroni intrappolati nei floating gate delle giunzioni di passare attraverso il gate, rimuovendo così la carica elettrica immagazzinata che rappresenta gli zero e uno binari. L'emissione UV con questa lunghezza d'onda è molto pericolosa per l'uomo, specie nel caso di tempi lunghi di esposizione. Per questo motivo ti raccomandiamo di prestare estrema cautela nel sostituire la lampada.

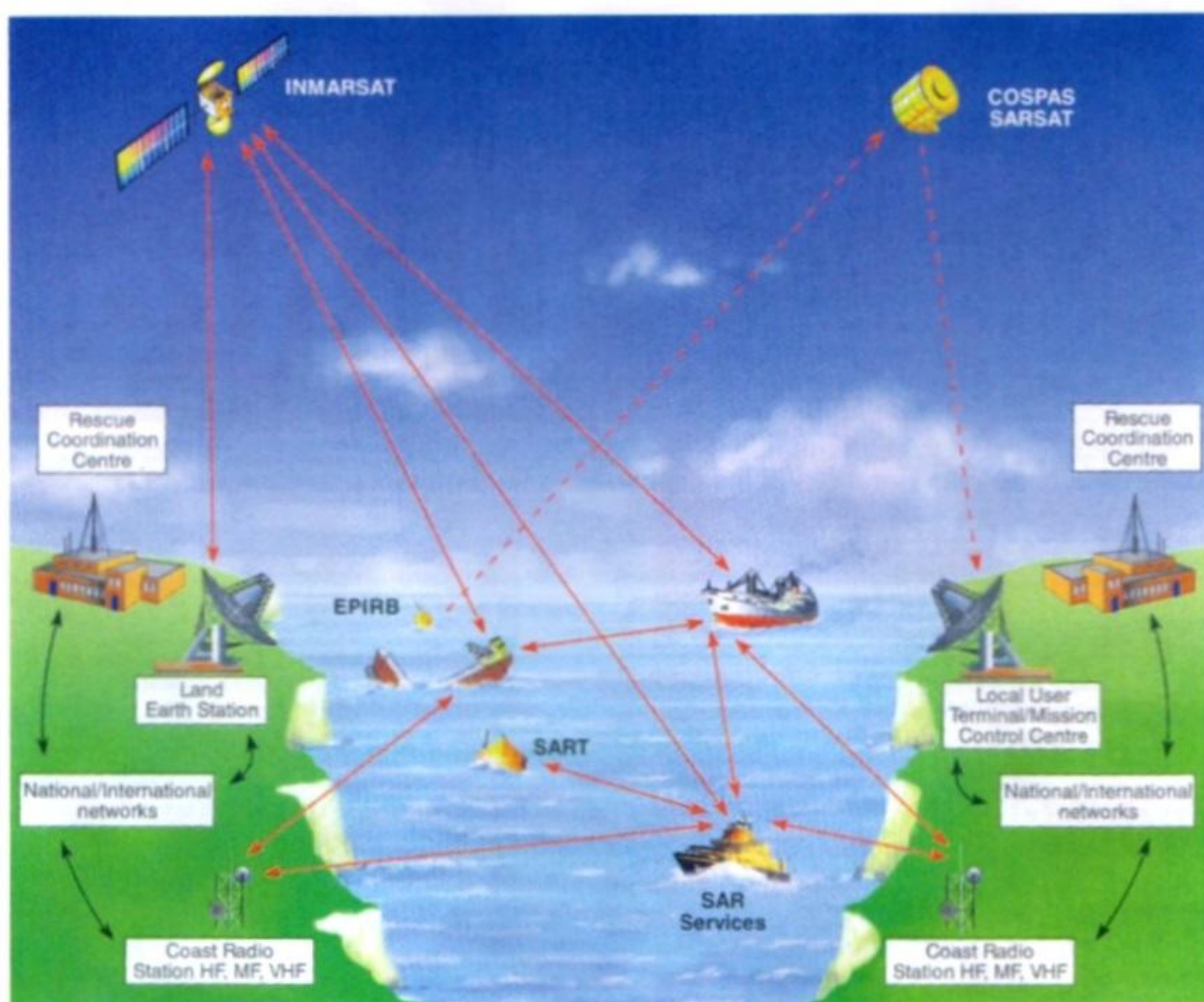
Per quanto riguarda la possibilità di utilizzare un LED UV con la stessa lunghezza d'onda, tecnicamente la cosa è fattibile ma dal punto di vista economico non ha molto senso. Un LED adatto a questo scopo presenta infatti un costo decisamente superiore, non giustificato dalla sua lunga durata (20-30 mila ore). Non pensiamo, infatti, che tu possa utilizzare il cancellatore per così lungo tempo.

Il mercato delle lampade UV a LED (specie quello UVA e UVB), sta crescendo rapidamente grazie all'incremento della potenza dei singoli dispositivi, alla durata ed all'elevata efficienza energetica rispetto alle lampade UV al mercurio. Interi settori industriali, specie quelli dove vengono utilizzati i processi di polimerizzazione stanno vivendo importanti cambiamenti. I sistemi a LED UV permettono di asciugare inchiostri, vernici, adesivi e altre sostanze foto sensibili tramite la polimerizzazione invece dell'evaporazione, con la prerogativa di accendersi istantaneamente senza tempo di riscaldamento.



EPIRB, come funziona?

Sto frequentando un corso per ottenere la patente nautica e, tra le cose che hanno attirato la mia attenzione di elettronico, c'è L'EPIRB,



l'apparecchiatura radio di sicurezza obbligatoria per chi naviga oltre le 50 miglia dalla costa. Mi piacerebbe sapere come funziona dal punto di vista tecnico...

Giovanni Strippoli - Genova

L'EPIRB (Emergency Position-Indicating Radio Beacon, emettitore radio di emergenza con indicazione della posizione) è un trasmettitore di soccorso usato in ambito marittimo per localizzare una imbarcazione in una situazione di reale emergenza.

Il dispositivo può essere dotato di attivazione manuale o automatica mediante un gancio idrostatico, che si attiva con la pressione dell'acqua, prima che raggiunga una profondità di 4 mt. L'attivazione del gancio idrostatico, fa scattare una serie di meccanismi che permettono l'espulsione dell'EPIRB dal suo contenitore, permettendogli di risalire in superficie. L'EPIRB trasmette un segnale d'allarme sulla frequenza di 406 MHz contenente il codice MMSI dell'imbarcazione l'ora e la posizione (nel caso integri un ricevitore GPS). Il segnale viene ricevuto dai satelliti meteorologici della rete COSPAS-SARSAT in orbita polare che svolgono anche questa funzione nell'ambito del servizio GMDSS (Global Maritime Distress Safety System), il sistema mondiale di soccorso e sicurezza in mare. Il massimo intervallo di black-out della rete COSPAS-SARSAT è di due ore; per gli EPIRB sprovvisti di GPS, la posizione viene calcolata dai satelliti stessi utilizzando l'effetto doppler. Il segnale di emergenza viene ritrasmesso dai satelliti al Centro COSPAS-SARSAT di competenza il quale dispone di tutti

i dati dell'imbarcazione che ha generato l'allarme e che tenta di mettersi in contatto con la stessa (per assicurarsi che non si tratti di un falso allarme) prima di rilanciare il segnale di allarme.

Nel caso questi tentativi non diano esito positivo, viene abilitata la ritrasmissione dell'allarme mediante la rete INMARSAT il cui segnale sulla banda L (circa 1.500 MHz) può essere ricevuto da altre imbarcazioni e aerei che si trovano in zona. Naturalmente viene attivata anche la rete SAR di competenza.

Prima di essere portato a bordo, l'EPIRB va programmato con il codice MMSI dell'imbarcazione e va registrato presso il centro COSPAS-SARSAT di competenza, nel caso dell'Italia quello di Bari.

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.elettronica.in.it/webforum

MC4 - Motion Control for

Il 15 marzo 2016 torna a Bologna MC4 – Motion Control for, la prima e unica mostra convegno italiana interamente dedicata alle tecnologie e ai prodotti per il controllo del movimento. Questa dodicesima edizione è la più importante occasione di aggiornamento professionale per tecnici e progettisti che operano in ambito industriale e nel settore energetico.

I partecipanti hanno infatti la possibilità di seguire decine di workshop tecnici e completare la propria esperienza di visita entrando in contatto diretto con le aziende leader del settore, le loro proposte tecnologiche, la loro esperienza nei più disparati settori applicativi: dal packaging al food & beverage, dalla meccanica all'elettronica, dai grandi impianti al mondo dell'energia. Passando per l'acquisizione dati, la comunicazione e il mondo dell'interfacciamento.

Motion Control for si rivolge a tecnici e progettisti operanti in ambito industriale e nel settore energetico (impiantistica produttiva, macchine automatiche, macchine utensili, manutenzione ecc.) che utilizzano:



- motori e motoriduttori
- servomotori
- azionamenti e regolatori di velocità
- controllo assi
- sistemi di posizionamento
- comandi e attuatori
- sensori e comunicazione

Le novità di quest'anno si inseriscono nel format consolidato della manifestazione che prevede anche un'ampia area espositiva, dove i visitatori potranno "toccare con mano" i prodotti e incontrare gli esperti delle principali aziende del settore.

Gli ingegneri potranno sfruttare questa occasione di aggiornamento professionale anche per arricchire il proprio bagaglio di crediti per la formazione permanente (CFP).

La partecipazione è gratuita, previa pre-registrazione.

<http://mc4.mostreconvegno.it/>

Analog Devices & Xilinx - Software Defined Radio Seminar

Silica, Analog Devices e Xilinx organizzano un seminario gratuito sulla Software Defined Radio (SDR) argomento che permetterà di entrare nel dettaglio teorico/tecnico delle comunicazioni wireless nel range 400MHz - 4GHz. Durante la giornata gli specialisti Analog Devices e Xilinx analizzeranno argomenti quali conversione High Speed, JESD204B, componenti strategici nella catena del segnale, board FMC, Driver, Flusso di progetto e tools di sviluppo.

L'Agenda prevede:

- 08:30 Registrazione
- 09:00 High Speed JESD204B and FMC Rapid Systems Development & Integration (presented by Ching Man, Analog Devices, Snr. Applications Engineer). Poiché la risoluzione e la velocità dei convertitori ADC e DAC è aumentata notevolmente, l'esigenza di disporre di un'interfaccia più efficiente

e performante è diventata imprescindibile. Questa sessione analizzerà nel dettaglio il nuovo standard/protocollo JESD204B e le board FMC utili per velocizzare le fasi di sviluppo e prototipazione. Verranno inoltre evidenziati alcuni "tips and tricks", drivers, PCB layout e requisiti HW per l'utilizzo di questa nuova interfaccia seriale ad alta velocità.

- 10:30 Coffee break
- 10:45 Integrated Software Defined Radio (presented by Ching Man, Analog Devices, Snr. Applications Engineer). Questa sessione analizza la catena del segnale analogico high-speed per le comunicazioni wireless: dalla RF alla banda base e l'elaborazione digitale attraverso FPGA. Gli argomenti trattati includono il front-end analogico, la conversione diretta dell'architettura radio, la conversione AD/DA, l'interfacciamento alla FPGA e l'elaborazione digitale.

Saranno inoltre analizzate le imperfezioni introdotte dal modulatore/demodulatore con particolare focus sugli effetti delle variazioni di temperatura e frequenza. La board di valutazione utilizzerà l'ultima generazione di convertitori high speed di Analog Devices e la più performante tecnologia System-On-Chip Xilinx (Zynq-7000, Processore ARM Cortex-A9 + FPGA sullo stesso silicio).

- 12:15 Software Defined Radio Demo (presented by Ching Man, Analog Devices, Snr. Applications Engineer). In questa sessione si mostrerà in azione l'ultima generazione di con-



NXP Smarter World Tour

EVENTI

All'inizio di quest'anno, NXP Semiconductors ha accelerato la sua strategia di connessioni e tecnologie sicure per un futuro più intelligente, annunciando la fusione con Freescale Semiconductor. Recentemente l'accordo si è positivamente concluso e oggi la nuova NXP è la quarta azienda di semiconduttori (escluse le memorie) al mondo.

Lo Smarter World Tour Truck, unico nel suo genere, consente di prendere visione del nuovo mondo dell'Internet of Things. Durante un viaggio di 6 mesi sulle strade

d'Europa, che toccherà Milano il 2 febbraio 2016, Torino il 4 febbraio e Bologna il 10 febbraio 2016, il Truck avrà modo di mostrarci le tecnologie, i prodotti, i sensori, i componenti e il software che piloteranno l'Internet of Things di oggi e i vostri dispositivi connessi di domani. Avrete modo di valutare l'ampio portafoglio prodotti della nuova NXP: dal più piccolo microcontrollore alla più complessa infrastruttura per il networking, passando per le soluzioni NFC più sicure. Saranno presenti oltre 130 demo con dimostrazioni live focalizzate sulle più importan-

ti applicazioni IoT. Toccherete con mano i sistemi di sviluppo e vedrete com'è facile usarli: sicurezza, efficienza energetica e un completo eco-sistema saranno disponibili per rendere concrete le vostre applicazioni IoT.

NXP è l'unica società in grado di fornirvi soluzioni IoT, dal più semplice nodo attraverso il network fino al cloud. Nessun'altra azienda può offrirvi tanto e lo Smarter World Tour Truck di NXP vi guiderà verso il futuro.

Oltre all'IoT truck experience, visitabile tutto il giorno, avrete modo di assistere durante la giornata a una serie di presentazioni che entreranno nel dettaglio delle tecnologie e delle famiglie di prodotto della nuova NXP. A Milano il 2 febbraio 2016

e a Torino il 4 febbraio 2016 (www.avnet.com); a Bologna il 10 febbraio 2016 (www.silica.com).



vertitori ad alta velocità Analog Devices, i dispositivi RF e clock sempre di Analog Devices, congiuntamente con il SoC Xilinx Zynq-7000. Inoltre verrà costruito un modello completo di tutto il ricetrasmittitore utilizzando gli strumenti di The MathWorks. Ciò consentirà di provare differenti scenari ed ottimizzare gli algoritmi in banda base. In particolare per l'implementazione degli algoritmi sarà illustrata la tecnica di generazione automatica del codice HDL.

- 13:00 Lunch, Learn and Discuss
- 14:00 Xilinx DSP design flows (presented by D. Bagni, Xilinx DSP Specialist for EMEA). Questa parte del seminario illustrerà il flusso di progettazione DSP di Xilinx per le applicazioni Software Defined Radio: la sinergia tra il model based design realizzato con System Generator for DSP e la sintesi ad alto livello (C-to-RTL) di Vivado HLS accorciano drasticamente i tempi di sviluppo e portano la produttività a livelli straordinari. Verrà in particolare mostrato come un algoritmo descritto in C/C++ viene compilato attraverso Vivado High Level Synthesis generando codice RTL sintetizzabile, aggiunto come un blocco di System Generator per la verifica, e poi implementato utilizzando Vivado Design Suite.

La mappatura e lo sharing automatico delle risorse effettuati da Vivado HLS è fondamentale per la conversione di un algoritmo complesso in una soluzione ottimizzata in termini di prestazioni e risorse impiegate. Saranno inoltre presentati i

ANALOG DEVICES

nuovi Tools ad alto livello recentemente introdotti da Xilinx: SDSoc per il partizionamento HW / SW su dispositivi della famiglia Zynq edSDAccel, ambiente di sviluppo basato su linguaggio OpenCL.

- 15:30 High Speed Converters: Front-end Design, Conditioning & Clocking (presented by Neha Baheti, Analog Devices, Applications Engineer). La sessione approfondisce gli aspetti relativi alle prestazioni dell'interfaccia JESD204B per i convertitori high speed con particolare focus sulle soluzioni per il front-end: pilotaggio dei convertitori ADC, topologie circuitali, parametri di sistema, implementazione di filtri differenziali per le catene di segnale high speed, condizionamento del segnale d'uscita dei convertitori DAC e riduzione del jitter con soluzioni di clock appropriate. Infine si parlerà di come simulare le prestazioni di un convertitore JESD204B utilizzando una board di sviluppo high speed Analog Devices in combinazione con una FPGA Xilinx.
- 17:00 End

Il 3 febbraio 2016 a Firenze, il 4 febbraio 2016 a Cinisello Balsamo.
www.silica.com



Altera MAX10 & IDT Components Seminar

Il seminario, la cui partecipazione è gratuita previa registrazione, sarà di carattere prettamente tecnico, con alcune demo pratiche, tenuto da personale specializzato di EBV Elektronik e IDT.

L'agenda prevede:

- 09:15 – 09:30 Welcome and Registration.
- 09:30 – 10:00 Max10 Family general introduction and overview.
- 10:00 – 10:30 Architecture, features and technical details.
- 10:30 – 11:00 Solving board design challenges with IDT products (part1).
- 11:00 – 11:15 Coffee Break.
- 11:15 – 13:00 Practical example with Nios2 system using standard and custom peripherals.
- 13:00 – 14:00 Lunch.
- 14:00 – 14:30 Solving board design challenges with IDT products (part2).
- 14:30 – 16:00 Practical example: Project creation with QSys, system generation and compilation with Quartus2, software download and project debug on the evaluation kit.

A tutti i partecipanti verrà applicato uno sconto del 50% sull'acquisto del kit di sviluppo Altera EK-10M08E144ES/P + Cavo di programmazione USB-BLASTER-CABLE TER. Inoltre i partecipanti riceveranno in omaggio uno USB stick contenente l'ultima versione del tool IDT Timing Commander.

Il giorno 11 febbraio a Modena.

www.ebv.com

ALTERA

Workshop Vivado SpeedWay Design

Avnet e Silica presentano il nuovo workshop "Xilinx SpeedWay Design" che utilizza la tecnologia low power Artix-7 FPGA Xilinx e la nuova board Arty kit di Avnet.

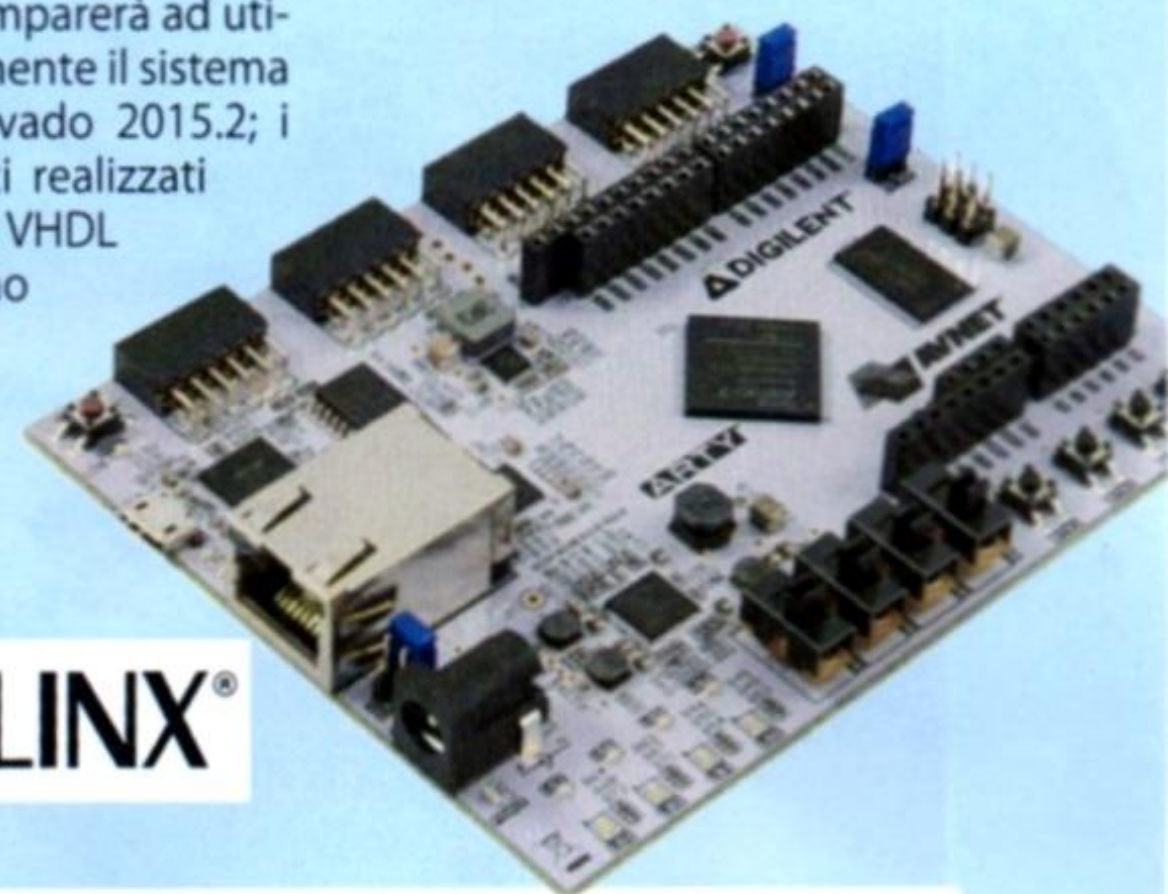
Vivado è la suite di Xilinx in grado di affrontare la progettazione degli FPGA più evoluti, utilizzati per realizzare SoC sempre più grandi e complessi.

Utilizzando la Metodologia UltraFast di Xilinx e la scheda Arty, l'utente imparerà ad utilizzare rapidamente il sistema di sviluppo Vivado 2015.2; i progetti pratici realizzati con Verilog o VHDL RTL saranno inseriti,

analizzati e debuggati utilizzando Vivado Design Edition la cui licenza può essere attivata utilizzando il voucher incluso nel kit di valutazione Artix-7 35T "Arty".

A Torino il 22 marzo, a Firenze il 23 marzo, a Roma il 24 marzo, a Modena il 29 marzo, a Padova il 30 marzo, a Milano il 31 marzo 2016.

www.silica.com



XILINX

M2M New Opportunity in the IoT Market



Arrow Electronics organizza un nuovo appuntamento con "M2M New Opportunity in the IoT market! Billions of Low-Power connected objects, the new business wave."

Ancora una volta Arrow Electronics è in prima linea sulle soluzioni IoT. Primo distributore italiano ad introdurre NETTROTTER, il broadcaster che permetterà di connettere milioni di dispositivi sfruttando la nuova tecnologia low power SigFox.

L'evento permetterà di incontrare tutti gli attori del mercato, dall'inventore della tecnologia (SigFox), al gestore dell'infrastruttura (Nettrotter)

passando dai produttori di soluzioni come Atmel, On Semiconductor, Silicon Labs, ST Microelectronics, Telit e Texas Instruments.

L'agenda prevede:

- 09:30 - 10:00 Welcome
- 10:00 - 10:30 Coffee
- 10:30 - 11:15 SigFox
- 11:15 - 11:30 Nettrotter
- 11:30 - 12:00 Axel
- 12:00 - 12:30 SmartEverything
- 12:30 - 13:00 Atmel
- 13:00 - 14:00 Silicon Labs
- 14:00 - 14:30 Telit
- 14:30 - 15:00 Lunch
- 15:00 - 15:30 ST Microelectronics
- 15:30 - 16:00 Texas Instruments
- 16:00 - 16:30 On Semiconductor

A Milano il 4 febbraio 2016 presso l'Hilton Hotel, Via Galvani 12.

www.arroweurope.com

STM32L4 Hands-on Workshop

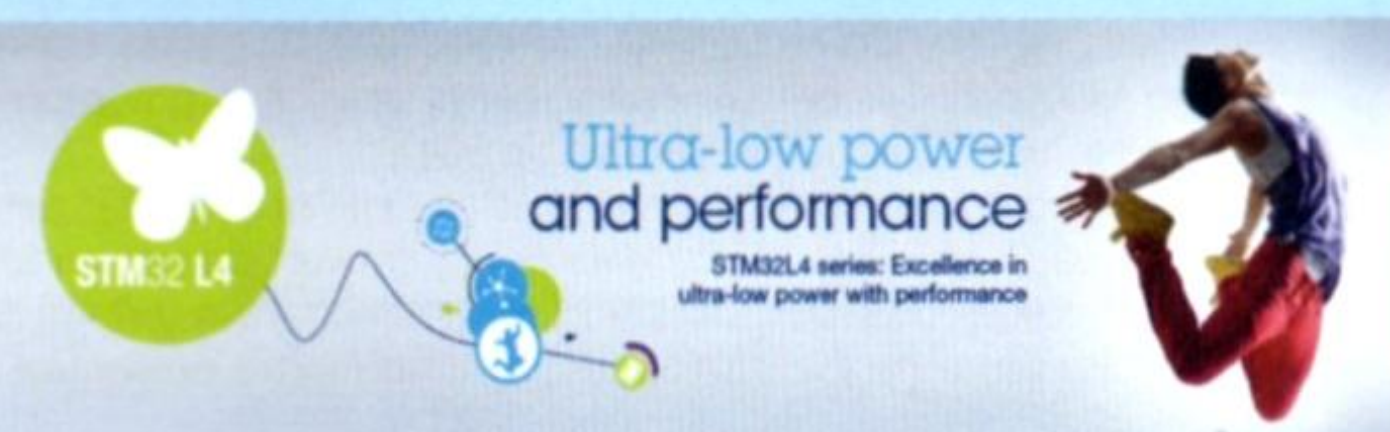
Un evento alla scoperta della famiglia STMicroelectronics STM32L4 con tecnologia ARM Cortex-M4 in grado di offrire una combinazione unica di alte prestazioni e bassissimo consumo. Lo staff tecnico unisce teoria ad esercizi pratici per fornire agli sviluppatori una approfondita comprensione dei microcontrollori STM32L4 che rafforzerà sia la conoscenza specifica che generale dei microcontrollori.

Ogni partecipante riceverà un kit omaggio STM32L4 Discovery e un set completo di manuali.

Gli esercizi pratici consentiranno di evidenziare la tecnologia alla base di questi dispositivi e le molteplici combinazioni possibili, con applicazioni sia in ambito industriale che nel campo dei dispositivi indossabili, con funzionalità integrate che consentono di ridurre al minimo non solo i consumi ma anche il BOM.

A Milano il 12/02/2016 e a Napoli il 17/3/2016.

www.ebv.com



■ 23-25 MARZO 2016

Embedded World

Norimberga • Germania



Miniaturizzazione spinta in abbinamento a livelli di elaborazione sempre più elevati, sicurezza ed efficienza nelle comunicazioni di rete, spesso anche per sistemi mobili: gli sviluppatori di sistemi embedded si trovano a dover soddisfare requisiti sempre più stringenti.

Così come a dover affrontare numerosi altri temi quali l'affidabilità dei sistemi elettronici, l'intelligenza distribuita, l'Internet delle Cose, l'efficienza energetica, e l'e-mobility, solo per citarne alcuni.

Sono tutti temi all'ordine del giorno nel settore industriale embedded che saranno al centro della quattordicesima edizione della manifestazione Embedded World 2016 di Norimberga.

Come ogni anno i visitatori potranno ottenere informazioni sui prodotti più innovativi direttamente dai principali produttori di semiconduttori e sistemi, tutti presenti alla manifestazione con propri stand e demo dei propri prodotti.

Anche quest'anno il tema dell'Internet of Things farà da sfondo al nutrito programma di conferenze e laboratori, tra le quali l'embedded world Conference, la più importante conferenza al mondo sulla tecnologia embedded; la conferenza sarà anche un momento di scambio di

■ 11-20 MARZO 2016

SXSW 2016

Austin [Texas] • USA

SXSW 2016 South by Southwest, o più semplicemente SXSW, è diventata negli anni una delle manifestazioni più importanti degli Stati Uniti, un evento culturale unico al mondo, che richiama tutti gli anni ad Austin, in Texas, appassionati di cinema, musica e tecnologia. Una manifestazione che poteva nascere e svilupparsi solamente in questa città, che non è solamente la capitale del Texas, ma che è anche la capitale mondiale della musica dal vivo con centinaia di locali dove si esibiscono band di ogni genere. Ma non solo. Austin è sede di una delle più grandi università americane, di tantissime aziende elettroniche e informatiche e, per completare il quadro, è ...l'unica città del Texas a

maggioranza democratica. Un evento i cui contenuti anticipano sicuramente i tempi, un po' come è successo per la Maker Faire californiana, e che proprio per questo val la pena di conoscere e seguire.

Il Festival (perché di festival si tratta) offre centinaia di eventi suddivisi in tre grandi aree tematiche: musica, film e interattive, quest'ultima dedicata all'innovazione e alle nuove tecnologie.

Al SXSW di Austin sono di casa personaggi come Elon Musk, Anne Wojcicki, Adam Savage, Bill Gates, Chelsea Clinton e tanti altri. Quest'anno il Festival si svolgerà dal 11 al 15 marzo (SXSW Interactive), dal 11 al 19 marzo (SXSW Film) e dal 15 al 20 marzo (SXSW Music).

www.sxsw.com



opinioni ad alto livello su questi temi e sui principali trend del momento. Figure di spicco nel campo delle tecnologie embedded illustreranno la loro visione sui futuri sviluppi di questa tecnologia e sulle sfide che attendono le aziende che operano in questo settore.

Un'altra importante conferenza di Embedded World 2016 sarà quella sui display, che quest'anno si focalizzerà sull'ottimizzazione di questi dispositivi, dai touch-screen agli schermi LED e OLED. Infine, come ogni anno, il terzo giorno della manifestazione sarà dedicato al premio edC Student Paper Award.

L'edizione 2015 ha battuto tutti i precedenti record con la presenza di 25.500 visitatori e 902 espositori; le previsioni per il 2016 parlano di un ulteriore incremento di visitatori ed espositori. Inoltre, un sondaggio indipendente che valuta la soddisfazione di visitatori e espositori ha evidenziato come Embedded World sia una delle manifestazioni più apprezzate tra gli eventi di questo tipo

www.embedded-world.de



■ 22-25 MARZO 2016

GSMA Mobile World Congress

Barcellona - Spagna

All'insegna del "Mobile is Everything", il GSMA Mobile World Congress si appresta ad accogliere dal 22 al 25 febbraio 2016 visitatori ed espositori presso la Fira Gran Via di Barcellona, con eventi che avranno luogo anche alla Fira Montjuïc. Quest'anno, alla più importante manifestazione mondiale di telefonia mobile e tecnologie connesse, sono attesi oltre 95.000 visitatori provenienti da più di 200 paesi.

Alla parte congressuale prenderanno parte, con specifiche keynote, i rappresentanti delle principali aziende del settore, da Simon Segars CEO di ARM, a Hans Vestberg, CEO di Ericsson. Oltre alle keynote, la conferenza includerà sessioni di approfondimento sui trend del momento, dai big data al commercio digitale, dai servizi finanziari in mobilità a sicurezza e privacy. Su tutti, ancora una volta, le sessioni dedicate ai vari aspetti dell'Internet of Things.

Come consuetudine, al Mobile World Congress saranno presenti oltre 2.000

aziende la cui attività spazia dai prodotti, alle infrastrutture, dai servizi tecnologici al software, passando attraverso gli immensi stand degli operatori globali.

Tra i nomi: Alcatel-Lucent, AT & T, Deutsche Telekom, Ericsson, Ford Motor Company, Google, Huawei, Intel, Lenovo, Microsoft, NEC, Nokia, Qualcomm, Samsung, SK Telecom, Sony Mobile Communications, Telefónica, VMware e Vodafone, oltre a tanti altri. Lo spazio espositivo comprenderà anche una serie di mostre incentrate su specifiche aree tecnologiche: Graphene Pavilion, Green Pavilion, IoT Pavilion, Mobile Money Pavilion, Wearables Pavilion.

Il GSMA Innotion City tornerà ad essere un punto culminante della manifestazione con partner del calibro di AT & T, Jasper, KT Corporation e Sierra Wireless. Su una superficie di 1.900 metri quadrati, questo spazio unico consentirà ai partecipanti di sperimentare la tecnologia mobile in contesti particolari e vedere come prodotti e servizi mobili collegati pos-

sono migliorare la vita quotidiana dei cittadini e delle imprese, in applicazioni industriali, in agricoltura, tra le mura di casa, in viaggio, in ambito sanitario e in tante altre applicazioni. Parte integrante del Mobile World Congress è il GSMA Ministerial Programme, che riunisce governi, regolatori e leader del settore per discutere di norme, politiche di sviluppo, accordi e più in generale per plasmare lo sviluppo della telefonia mobile in tutto il mondo che vede società sempre più collegate e senza barriere. Oltre all'esclusivo vertice mondiale, questo programma prevede sessioni focalizzate su argomenti chiave come inclusione digitale, identità mobile, l'Internet of Things, nonché sessioni regionali per esplorare i problemi specifici dei mercati di Asia Pacifico, America Latina, Medio Oriente, Nord Africa e l'Africa sub-sahariana. Il GSMA Ministerial Programme 2016, sponsorizzato da Intel, ospiterà più di 1.500 partecipanti nel corso dei quattro giorni di Mobile World Congress. Mobile World Congress continua ad ampliare la gamma dei programmi





dei partner, piattaforme per discutere le tendenze del mercato e su interessi specifici dei partecipanti. Quest'anno sono in programma anche una serie di sessioni verticali focalizzate su settori quali sanità, mobile advertising e marketing e media. Tra gli altri programmi dei Partner da segnalare:

App Planet - Collega la comunità di sviluppatori e l'ecosistema più ampio cellulare con esposizione, conferenze per gli sviluppatori app, prodotti e dimostrazioni di servizi e laboratori da alcuni dei più grandi e influenti aziende del settore di telefonia mobile. Generalitat de Catalunya è sulla Gazzetta App Planet Sponsor.

Power Hour - Il quartiere dei teatri ospiterà le sessioni di Power Hour, con presentazioni di un'ora di leader del settore su una serie di argomenti. Il primo ha confermato Power Hour Presenter è Laird Technologies.

Quattro anni da oggi - 4YFN è un programma internazionale organizzato da Mobile World Capital Barcelona e da GSMA, che riunisce leader, start-up focalizzate sul mobile, imprenditori, investitori, acceleratori, incubatori e rappresentanti di grandi aziende sia del mondo digitale che non. Il programma, che sarà guidata da esperti a livello mondiale in materia di imprenditorialità e innovazione, si compone di conferenze, workshop, discussioni interattive, incontri one-to-one con gli investitori e competizioni tra start-up.

Infine una notizia di servizio: da quest'anno è in funzione un nuovo tronco di metropolitana con una stazione che si trova direttamente sotto Fira Gran Via e che consente di raggiungere più facilmente la manifestazione da qualsiasi punto della città.

www.mobileworldcongress.com

■ 18 FEBBRAIO 2016

NIDays 2016

Milano



Torna a Milano la 23^a Edizione di NIDays, il Forum Tecnologico sulla Progettazione Grafica di Sistemi che riunisce oltre 600 ingegneri, tecnici e insegnanti e che si pone l'obiettivo ambizioso di illustrare concretamente le tecnologie necessarie al funzionamento dell'Internet of Things e di mostrare come la piattaforma NI ne supporterà la realizzazione. Non a caso l'evento di quest'anno è caratterizzato dall'hashtag #tuttoconnesso. James Truchard, CEO, Presidente e Co-fondatore di National Instruments, alternandosi con altri speaker di fama internazionale, sarà presente sul palco di NIDays per accompagna-

re il pubblico in sala in un viaggio attraverso le tappe di questo percorso. L'evento, della durata di un giorno, è l'occasione per fare il punto sulle tecnologie più recenti e sui sistemi definiti da software, per sviluppare applicazioni di test, misura e controllo in modo più rapido, intelligente e profittevole.

Il programma di NIDays 2016 seminari e conferenze tematiche, dimostrazioni pratiche dei prodotti NI e dell'ambiente di sviluppo LabVIEW, con particolare riguardo alle novità introdotte con la versione 2015.

www.nidays.it

MOSTRE MERCATO

FAENZA (RA)

EXPO ELETTRONICA

Quartiere Fieristico - Faenza (RA)

Organizzazione: Blu Nautilus srl

Tel: 054153295 - Fax: 0541439584

info@expoelettronica.it

www.expoelettronica.it

05-06 Marzo 2016

ERBA (CO)

FIERA DELL'ELETTRONICA

Lario Fiere - Erba (CO)

Organizzazione: Lariofiere

Tel: 031637414

erbaelettronica@lariofiere.com

www.lariofiere.com

05-06 Marzo 2016

OSNAGO (LC)

FIERAMERCATO DELL'ELETTRONICA

Fiera San Giuseppe - Osnago

Organizzazione: Printmix

Tel: 0362999186 - Fax: 0362978042

info@fieramercatodellelettronica.it

www.fieramercatodellelettronica.it

05-06 Marzo 2016

MONTICHIARI (BS)

RADIANTISTICA EXPO

Centro Fiere del Garda - Montichiari (BS)

Organizzazione: CENTRO FIERE DEL GARDA SPA

Tel: 030961148 - Fax: 0309961966

radiantistica@centrofiere.it

www.radiantistica.it

12-13 Marzo 2016

PIANA DELLE ORME (LT)

Mostra Scambio Radio d'Epoca, HI-FI

Museo Piana delle Orme,

Via Migliara 43 Borgo Faiti

Organizzazione: Franco Nervegna

Tel: 3386155146

www.franconervegna.it

12-13 Marzo 2016

BASTIA UMBRA (PG)

EXPO ELETTRONICA

Quartiere Fieristico - Bastia Umbra (PG)

Organizzazione: Blu Nautilus srl

Tel: 054153295 - Fax: 0541439584

info@blunautilus.it

www.expoelettronica.it

19-20 Marzo 2016

L'elenco aggiornato di tutte le Mostre Mercato del 2016 è disponibile sul sito www.elettro-nicain.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.elettro-nicain.it/myMain/Fiere/p_fiere.asp

LTM4677, regolatore a singola uscita da 36A o doppia da 18A con PMBus

Linear Technology presenta l'LTM4677, un regolatore μ Module DC/DC step-down con due uscite da 18A o una sola uscita da 36A, con interfaccia digitale seriale PMBus per il power management digitale autonomo di sistema. L'interfaccia digitale consente ai progettisti di sistemi e agli operatori remoti di controllare e supervisionare le condizioni e i consumi di potenza di un sistema tramite un bus I2C con i comandi READ e WRITE.

L'LTM4677 viene utilizzato nei sistemi di trasporto ottici dei dati, switch e router per comunicazioni dati e telecomunicazioni, apparecchiature di test industriali, robotica, sistemi RAID



e aziendali in cui i costi di energia, raffreddamento e manutenzione sono aspetti critici e devono essere misurati in modo costante e preciso.

L'LTM4677 include EEPROM, interfaccia I2C, doppio controller DC/DC, MOSFET di potenza, induttori e compo-

nenti di supporto. Le due uscite da 18A possono condividere la corrente per fornire un'uscita singola fino a 36A. È possibile sfasare tra loro fino a quattro dispositivi LTM4677 per condividere la corrente con un'uscita fino a 144A. Un altro vantaggio dell'LTM4677 è la possibilità di condivisione della corrente con regolatori μ Module senza interfaccia PMBus, come l'LTM4620 da 26A e l'LTM4630 da 36A.

Le versioni dell'LTM4677 con potenza inferiore, l'LTM4675 (9A, 9A) e l'LTM4676A (13A, 13A), sono compatibili a livello di pin con tale dispositivo e sono alternative perfettamente funzionanti.

www.linear.com

F2923, deviatore SPDT realizzato con tecnologia a impedenza costante K_Z



Integrated Device Technology (IDT) ha presentato il primo switch RF sul mercato in configurazione di deviatore (SPDT, single-pole, double-throw) realizzato con la sua tecnologia a impedenza costante K_Z in attesa di brevetto. Il dispositivo IDT F2923 è uno switch a bassa perdita per segnali a radiofrequenza (RF) progettato per affrontare una moltitudine di ap-

plicazioni RF, tra cui stazioni radio base (2G, 3G, 4G, 5G), reti di raccolta del traffico wireless (backhaul), TV via cavo e dispositivi portatili.

Con una copertura in frequenza che spazia da 300 kHz a 8000 MHz, l'innovativa tecnologia K_Z di IDT garantisce che i dispositivi mantengano un'impedenza praticamente costante (con VSWR < 1,4:1 anziché di 9:1 come nei normali switch) durante la commutazione da una porta RF ad un'altra senza compromettere isolamento, linearità o perdita di inserzione. Il dispositivo F2923 viene alimentato con una tensione singola di 3,3 V.

www.IDT.com

TFT da 5.0 pollici "IPS-like" di KOE

KOE ha annunciato l'introduzione di un nuovo display TFT UWVA da 5.0 pollici. Il TX13D06VM5BAA presenta una risoluzione WVGA (800 x 480 pixel) e un formato a 16:9. I moduli LCD di KOE ad angolo di visione ultra wide (UWVA) sono caratterizzati dalla tradizionale tecnologia TFT TN (Twisted Nematic) con migliori caratteristiche ottiche dell'LCD. Molte applicazioni embedded mediche e industriali richiedono omogeneità di colore per poter ottenere immagini nitide e uniformi e dai colori brillanti. I display UWVA di KOE hanno una performance ottica "IPS-like" che permette una stabile uniformità di colore in tutte le di-

rezioni (su / giù, sinistra / destra) anche con un ampio angolo di visione di 80 gradi. L'eccellente performance ottica si ottiene con una luminosità specifica di 800cd/m², un rapporto di contrasto di 400:1 e dal trattamento anti-glare. La retroilluminazione a LED ha una luminosità specifica di 50K ore fino a metà della luminosità stessa.

www.koe-europe.com



AD777x, SoC per il monitoraggio delle Smart Grid

Analog Devices, ha presentato un SoC (Sistem-on-Chip) per l'acquisizione dati a 24 bit progettato per migliorare le prestazioni delle apparecchiature di protezione, di monitoraggio e di misura della potenza, utilizzate per gestire la trasmissione e distribuzione di energia elettrica nei sistemi Smart Grid.

L'AD7770 garantisce elevate prestazioni e consente l'impiego di relè di protezione di dimensioni più ridotte mentre

l'AD7771 supporta la realizzazione di sistemi di controllo della rete in grado di fornire una diagnosi precoce in caso di guasti o anomalie di fun-



zionamento. Il terzo SoC della serie, l'AD7779, garantisce un rapido ripristino nei circuiti di interruzione.

La serie AD777x incorpora un amplificatore a guadagno programmabile (PGA) che facilita l'interfacciamento con le uscite dei sensori, consentendo ai progettisti di sistemi di utilizzare una piattaforma comune per applicazioni con diversi requisiti di precisione e sensori di vario tipo, tra cui trasformatori di corrente, trasformatori

di tensione e stringhe resistive. Inoltre, la serie AD777x riduce il costo totale del sistema, eliminando la necessità di componenti esterni quali amplificatori di segnale e circuiti PLL. L'intera serie AD777x incorpora convertitori della frequenza di campionamento che consentono la regolazione fine della frequenza di campionamento mantenendo una variazione entro i requisiti previsti dalla norma IEC61000-4-30 classe A.

www.analog.com



MAXREFDES67, AFE a 24 bit in formato Micro PLC da Maxim Integrated

I progettisti di sistemi d'automazione industriale possono oggi ottenere misure di alta precisione con il progetto di riferimento MAXREFDES67 di Maxim Integrated, un modulo universale d'ingresso (front end analogico, AFE) in formato Micro PLC. Con l'aumento del numero di bit nella conversione dei

segnali, i sistemi ad alta risoluzione sono sempre più sensibili al rumore, il che rende difficile la loro realizzazione ed il mantenimento del loro livello di precisione. MAXREFDES67 è un esclusivo progetto di riferimento di un AFE a 24 bit che consente di vincere questa sfida. In grado di accettare quattro

diversi tipi di segnali, il modulo universale d'ingresso analogico MAXREFDES67 non richiede ponticelli ed è configurabile al 100% via software.

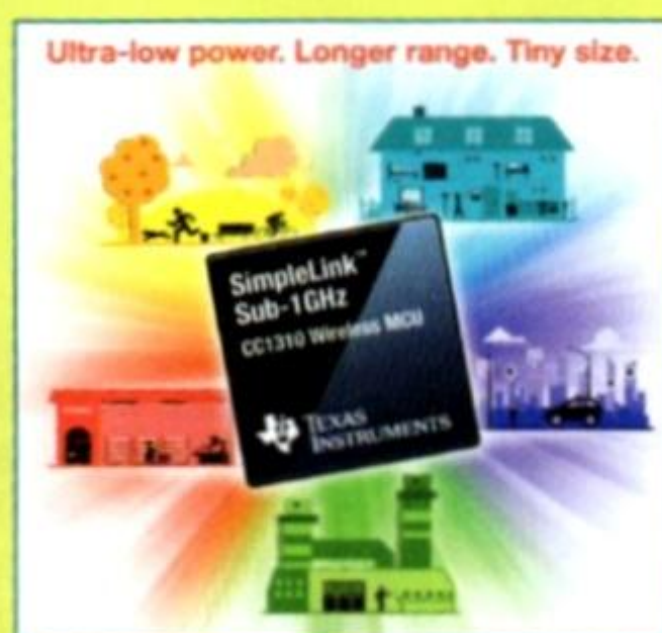
I file di progetto hardware, il firmware ed i dati di test sono forniti gratuitamente e disponibili online.

www.maximintegrated.com

CC1310, il chip RF per long-range a basso consumo targato TI

Anche Texas Instruments entra nel mercato dei dispositivi RF long-range low-power Sub-1 GHz con questa versione della MCU wireless SimpleLink che secondo TI è in grado di garantire una portata di 20 km e che può essere alimentata con una batteria a bottone.

La piattaforma a potenza ultrabassa SimpleLink è stata studiata per implementare facilmente la connettività a lungo raggio con consumi bassissimi in progetti Internet of Things (IoT). I nuovi microcontrollori (MCU) wireless SimpleLink Sub-1 GHz CC1310 garantiscono un'autonomia della batteria fino a 20 anni per applicazioni di automazione di fabbrica e domotica, allarmi e sicurezza, rete elettrica intelligente e reti di sensori wireless. TI ha semplificato lo sviluppo con le MCU wireless SimpleLink CC1310 mediante opzioni software che rispondono alle esigenze degli sviluppatori. Il software attualmente disponibile comprende esempi di comunicazione punto-punto con EasyLink e uno stack di protocolli wM-Bus basata



su TI RTOS, oltre allo stack per rete mesh Contiki 6LoWPAN. Gli sviluppatori possono accedere anche a strumenti di sviluppo, progetti di riferimento, formazione online e supporto della E2E community per agevolare il processo di progettazione. Inoltre, TI offre memorie scalabili in base alle dimensioni del codice dello sviluppatore, con memorie Flash da 32kB, 64kB e 128kB. Le MCU wireless SimpleLink Sub-1 GHz CC1310 saranno disponibili per le bande ISM 315 MHz, 433 MHz, 470 MHz, 500 MHz, 868 MHz, 915 MHz e 920 MHz in package QFN da 4x4, 5x5 e 7x7mm.

www.ti.com

MLX92292, switch/latch con sensibilità ai flussi magnetici ortogonali e laterali

Melexis ha compiuto un importante progresso nella tecnologia di misura magnetoresistiva, che avrà vaste ripercussioni sull'attuale progettazione di autovetture. Il suo ultimo prodotto – il componente MLX92292 – rappresenta di fatto un modo completamente nuovo di effettuare misure. Questo dispositivo fornisce funzioni di commutazione e di arresto, ma diversamente dai prodotti già esistenti sul mercato, esso può determinare la presenza di campi magnetici con flusso laterale, e non solo ortogonale. L'unicità di questa offerta è ulteriormente accresciuta dal fatto che il latch/switch MLX92292 è l'unico a supportare ad oggi il livello di sicurezza ASIL B (in conformità allo standard ISO 26262), con una varietà di meccanismi diagnostici integrati disponibili. La flessibilità è una caratteristica chiave dell'MLX92292. I clienti OEM posso-

no scegliere unità dirette pre-programmate, o in alternativa possono trarre vantaggio dalla capacità di programmazione a fine linea (EoL). Attraverso ciò, ciascun dispositivo può essere configurato (tramite il suo pin di uscita) durante il processo di produzione in fabbrica, di modo da realizzare completamente l'ottimizzazione a livello di sistema. La funzione di programmazione consente inoltre l'impostazione di entrambi i punti operativi magnetici in piccoli incrementi lungo un intervallo che va da -90mT a +90mT.

www.melexis.com



Soluzione Modem Powerline OFDM di terza generazione

Renesas OFDM
Powerline Modem Solution



Renesas Electronics ha annunciato la disponibilità di un nuovo modem powerline OFDM single-chip di terza generazione per la sua piattaforma powerline communication (PLC), leader di mercato. Anche questo dispositivo è in grado di supportare qualsiasi protocollo standard PLC, come ad esempio PRI-

ME1.3.6/1.4 e G3, per tutte le bande di frequenza globali, incluse CENELEC A, FCC e ARIB.

La terza generazione di soluzioni powerline a singolo chip di Renesas consente ai produttori che operano in ambito metering di gestire più facilmente le sfide del mercato in quanto contribuisce significa-

tivamente a ridurre il time-to-market, contenere al minimo i rischi e massimizzare l'efficienza dei costi del sistema. Con la sua memoria ancora più ampia e le migliorate prestazioni di elaborazione, possono essere supportati anche i protocolli PLC di nuova generazione come PRIME 1.4.

www.renesas.eu

VIPerOP, nuovo chip da ST per consumo zero in standby

Il tempo dei vampiri che succhiano corrente elettrica quando i nostri dispositivi elettronici sono in standby sta per finire (o quasi) grazie ad un nuovo chip di STMicroelectronics. L'innovativo dispositivo è conforme alle più stringenti normative in materia zero-power standby, ed è il primo al mondo a fornire un modo intelligente di gestire la funzione di "risveglio" negli elettrodomestici, negli impianti di illuminazione e nei sistemi industriali. Grazie alla sua architettura, VIPerOP è in grado di fornire alimentazione di standby per il microcontrollore host del sistema, pur essendo in modalità idle. Questo permette al sistema di essere riattivato tramite l'interfaccia utente principale dell'apparecchio, come il touchscreen o il telecomando. Non vi è alcuna necessità di un interruttore



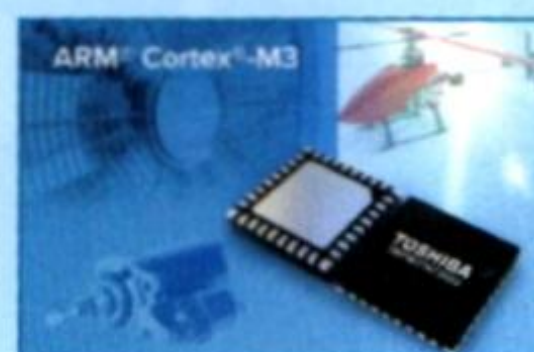
meccanico ad alta tensione di standby e VIPerOP consuma meno di 5 mW in modalità idle (a 230Vac). Questo consumo, secondo la clausola 4.5 della norma IEC 62301 per uffici e abitazioni, può essere considerato nullo, e il dispositivo può essere definito come zero-power standby.

Se viene utilizzato uno switch per controllare lo standby, il consumo in modalità idle scende sotto i 4 mW. La commutazione può essere realizzata con componenti SELV (Safety Extra-Low Voltage), che evitano l'impiego di ingombranti componenti ad alta tensione. VIPerOP è un integrato convertitore di potenza off-line di tipo switching che può essere configurato in modalità flyback, buck o buck-boost switched-mode power supply (SMPS).

www.st.com

TMPM37AFSQG, MCU miniatura per azionamento di piccoli motori

Toshiba Electronics Europe ha esteso la serie di microcontrollori TX03 basati su core ARM Cortex-M3 con il nuovo dispositivo TMPM37AFSQG. Il nuovo chip è il più piccolo microcontrollore con funzioni di controllo vettoriale per motori elettrici che comprende il coprocessore Toshiba VE+ (Vector Engine Plus) e un circuito di pre-pilotaggio (pre-driver) la cui combinazione facilita il controllo di motori brushless (senza spazzole) in corrente continua. Il circuito di pre-pilotaggio integrato nel dispositivo TMPM37AFSQG può comandare direttamente i MOSFET generando dei segnali in uscita di tipo PWM



trifase complementari con un intervallo minimo di 25 ns. Si tratta di funzionalità che rendono questo microcontrollore particolarmente adatto per pilotare piccoli motori, ventole per server, piccole ventole di raffreddamento, ventole di aerazione, ventole elettriche, ventole a soffitto, pompe e giocattoli. Il core del microcontrollore è alimentato dalla tensione di pre-pilotaggio, così da poter utilizzare un alimentatore unico per l'intero dispositivo.

www.toshiba.com

Nuovi prodotti ON Semiconductor per l'Internet of Things

ON Semiconductor ha ampliato la sua offerta di prodotti per le comunicazioni wireless con nuovi dispositivi ad alte prestazioni e basso consumo destinati alle bande sub-gigahertz, che sono il frutto dell'acquisizione della società AXSEM. I dispositivi permettono di realizzare connessioni wireless supportando funzionalità avanzate destinate alle applicazioni nel settore Internet of Things (IoT), telelettura di contatori intelligenti, domotica, reti di sensori e comunicazioni satellitari.

Il ricetrasmittitore RF a bassissima potenza AX5043 offre prestazioni molto avanzate grazie a una combinazione unica di consumi estremamente ridotti in trasmissione e ricezione, estrema sensibilità e grande selettività. La sua architettura radio programmabile permette di adattarsi a molti protocolli e standard di comunicazione, oltre che a protocolli di comunicazio-



ne personalizzati dal cliente. La famiglia di microcontrollori RF AX8052Fxx è stata specificamente progettata per facilitare la realizzazione di radio complete con un solo chip per tutti i livelli di complessità. Il modello AX8052F143 offre una soluzione a singolo chip caratterizzata da bassissimi consumi e con tutta la flessibilità necessaria per supportare le applicazioni tipiche del mercato IoT, oltre a offrire la compatibilità con le reti di messaggistica standard WM-BUS, POCSAG e FLEX, mentre il modello AX8052F151 è una soluzione per realizzare sistemi di comunicazione secondo lo standard ZigBee e 6LoWPAN.

www.onsemi.com

"GB6", nuova famiglia PIC24F di Microchip

Microchip Technology ha annunciato l'espansione del proprio portafoglio di microcontrollori (MCU) PIC a basso consumo con l'introduzione della nuova famiglia PIC24F "GB6" con memoria flash – completa di Error Correction Code (ECC) – fino a 1 MB e 32 KB di RAM, la prima MCU a 16-bit nel portafoglio Microchip in grado di offrire una capacità di memoria così grande: memoria flash che può essere suddivisa in due blocchi con la funzionalità Live Update, che consente di eseguire due applicazioni differenti, permettendo la programmazione simultanea di una partizione durante l'esecuzione del codice dell'altra. Questa eccezionale combinazione di prestazioni rende la famiglia PIC24F "GB6" ideale per applicazioni industriali, di elaborazione, in ambito medicale/

fitness, e in tutte quelle applicazioni portatili che richiedono una lunga durata della batteria, nonché nella memorizzazione di dati senza l'impiego di una memoria esterna come, ad esempio, nei contatori elettrici, nei sistemi di climatizzazione, negli scanner per impronte digitali e nei giochi elettronici. Le MCU della famiglia PIC24F "GB6" presentano un consumo durante il funzionamento di 190 μ A/MHz, mentre in modalità sleep assorbono solamente 3,2 μ A.

www.microchip.com





Utilizziamo un PIC10F322 in una scheda demo che permette di testarne le funzionalità realizzando, nello specifico, un controllo di temperatura.

TERMOSTATO A MICROCONTROLLORE

dell'Ing. VINCENZO MENDOLA

Quando diciamo microcontrollore, oggi pensiamo a componenti molto complessi e ad alte prestazioni, dimenticando che esiste tutta una serie di famiglie di microcontrollori ad 8 bit molto semplici, per applicazioni senza particolari pretese, offerti a cifre molto contenute e quindi alla portata anche degli sperimentatori. Questi dispositivi, per quanto essenziali ed economici, possono risolvere in maniera eccelsa applicazioni relativamente complesse; non a caso oggi il settore Automotive impiega decine, se non centinaia di microcontrollori e che tra essi vi sono, per la gestione di sottosistemi di bordo, piccoli microcontrollori a 8 bit. In questo articolo vogliamo dimostrarvi proprio come non sia sempre necessario scomodare schede di prototipazione come quelle della famiglia Arduino, Raspberry Pi, mbed e così via, per realizzare un'applicazione elettronica che richieda un controllo più o meno avanzato ma che, al contrario, circuiti e sistemi non particolarmente complessi possano essere affidati a piccoli microcontrollori, magari equipaggiati con periferiche specifiche.

Le principali aziende produttrici di microcontrollori hanno solitamente a catalogo almeno una famiglia a 8 bit; noi vogliamo presentarvi un prodotto Microchip, il PIC10F322, le cui caratteristiche ci hanno permesso di realizzare un semplice termostato veramente versatile, ma anche altre applicazioni. Il progetto che presentiamo è in realtà una demoboard che permette di testare alcune delle interessantissime caratteristiche del micro. Tra le tante qualità del PIC10F322 c'è il fatto che può essere programmato in C senza troppa difficoltà, utilizzando il compilatore XC8 (che può essere utilizzato in modalità gratuita facendo a meno delle ottimizzazioni introdotte dalle versioni a pagamento) e facendo uso dell'IDE MPLAB X (anch'esso gratuito) cui abbiamo dedicato un corso nei fascicoli dal 190 al 198.

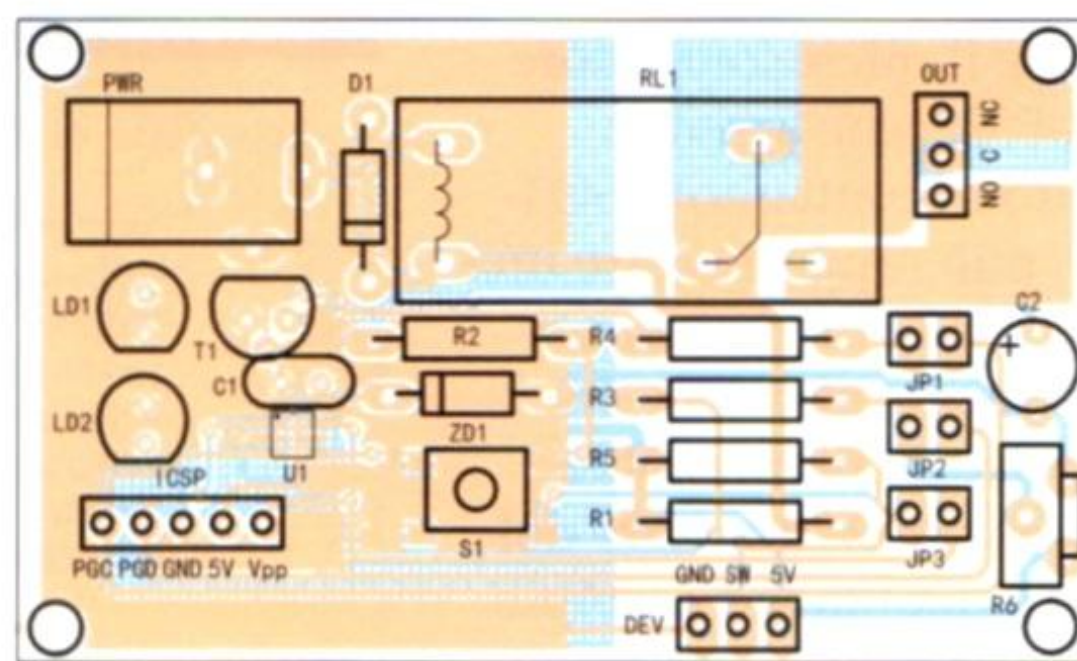
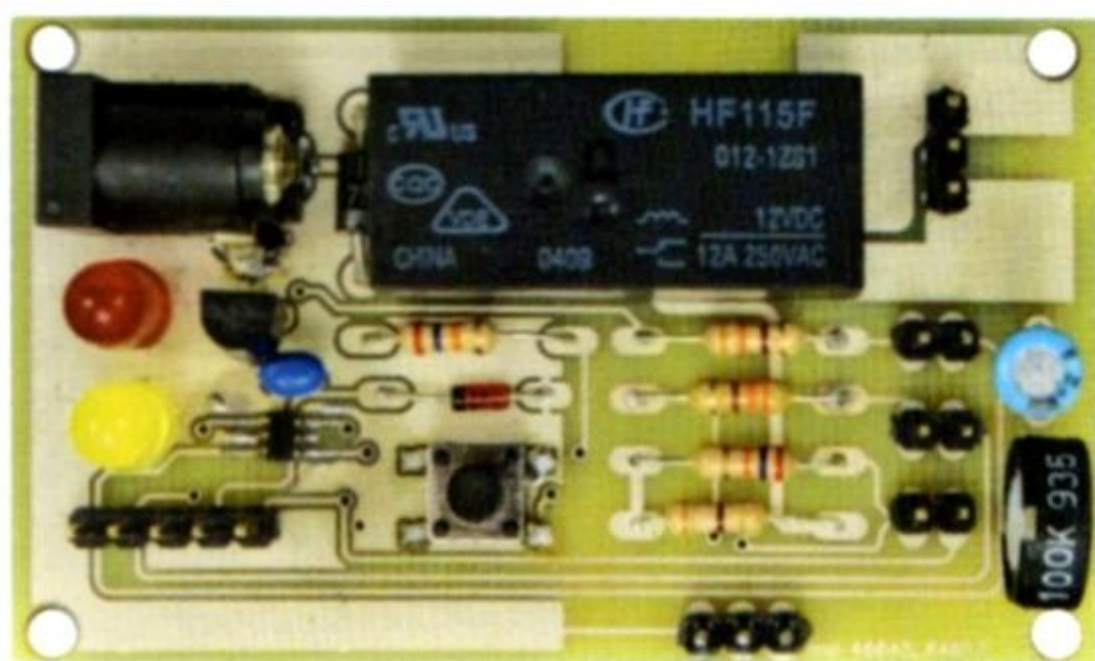
IL NOSTRO MICRO

Iniziamo a conoscere il PIC10F322, facendo una panoramica delle sue caratteristiche: il componente è disponibile in una versione a 6 pin in package



SCHEMA ELETTRICO

Sempre nell'ottica di realizzare la funzione termostato, abbiamo previsto il trimmer R6, inseribile sempre tramite il jumper JP2, il cui cursore viene letto dal



Elenco Componenti:

R1: 390 ohm	C1: 100 nF ceramico
R2: 27 kohm	C2: 10 μ F 63 VL elettrolitico
R3, R4: 330 ohm	LD1: LED rosso 5 mm
R5: 27 kohm	LD2: LED giallo 5 mm
R6: Trimmer 10 kohm MV	ZD1: Zener 5,1V 400mW
	D1: 1N4007
	T1: BC547

RL1: Relé 12V uno scambio
DEV: Deviatore volante
S1: Microswitch SMD
U1: PIC10F322T-I/OT (MF1236)

Varie:
- Plug alimentazione

- Strip maschio 2 vie (3 pz.)
- Strip maschio 3 vie (2 pz.)
- Strip maschio 5 vie
- Strip femmina 3 vie
- Circuito stampato S1237

convertitore A/D del microcontrollore; il trimmer è alimentato dagli stessi 5,1 volt che fanno funzionare il microcontrollore e, per come è fatto il circuito, se vogliamo utilizzarlo dobbiamo sconnettere il LED LD2, il quale, insieme alla sua resistenza limitatrice, si trova in parallelo al circuito di uscita del trimmer stesso.

R6 può essere utilizzato per impostare la soglia d'intervento alla quale attivare, ad esempio una ventola, tramite il relé. Nel circuito troviamo anche il pulsante S1, con il proprio resistore di pull-up R5, inseribile tramite il jumper J3 (da aprire durante la programmazione, dato che la linea cui è collegato il pulsante è anche l'MCLR dell'ICSP): premendolo, la linea RA3/MCLR del microcontrollore assume lo zero logico, mentre rilasciandolo va a livello alto. Oltre al pulsante S1, è stato previsto un deviatore DEV, da usare in maniera esclusiva (o l'uno o l'altro); a differenza del pulsante, il deviatore permette ad esempio l'impostazione di una posizione manuale e di una automatica. Ovviamente

la possibilità di utilizzare una delle due tipologie di selezione fornisce la massima versatilità durante la programmazione del codice, in base al comportamento desiderato. Come accennato, per provvedere alla programmazione in-circuit del microcontrollore, si deve utilizzare l'apposito programmatore. Ricordiamo che i tre jumper JP1, JP2 e JP3 servono per disconnettere -durante la programmazione- i pin ICSPDAT, ICSPCLK, MCLR/VPP del PIC dal resto del circuito utilizzato durante il normale funzionamento; tali jumper andranno pertanto rimossi durante la programmazione del PIC effettuata attraverso il connettore ICSP e dovranno essere inseriti una volta conclusa.

La demoboard può essere alimentata semplicemente applicando una tensione continua di 12V al DC PLUG PWR; la tensione fornita viene limitata e stabilizzata da uno Zener da 5,1V (D2) il quale, grazie alla resistenza R1 (che ne limita l'assorbimento) fornisce la tensione di alimentazione del microcontrollore U1 e del resto del circuito. I 5,1 volt vengono

filtrati dal condensatore C1, che sopprime eventuali disturbi impulsivi introdotti dalla linea di alimentazione o passati dall'alimentatore.

Gli stessi 12 volt d'ingresso alimenteranno invece lo stadio d'uscita, vale a dire la bobina del relé e il circuito pilota basato sul transistor.

Notate che il microcontrollore, qui alimentato a 5,1 volt, potrebbe funzionare anche a tensioni minori, utilizzando microcontrollori della serie LF, fermo restando che i livelli logici con cui lo interfacciamo devono essere compatibili con la tensione di alimentazione.

REALIZZAZIONE PRATICA

Descritto lo schema, passiamo alla parte pratica: a chi vorrà cimentarsi nel montaggio di questo semplice circuito, consigliamo come sempre di utilizzare un saldatore di buona qualità, preferibilmente una stazione saldante separata dalla tensione di rete tramite un trasformatore; comunque serve uno stilo a punta sottile, da 20÷25 watt di potenza. I componenti utilizzati sono tutti a montaggio tradizio-

nale, tranne il microcontrollore del PIC10F322, che essendo incapsulato in contenitore SOT-23 a 6 pin può rappresentare una criticità per chi non ha manualità e pratica con gli SMD. Per montarlo ricordiamo le solite raccomandazioni: utilizzare del fluxante di qualità (ad esempio il Kester 951) da applicare in piccola quantità sulle piazzole prima di saldare l'integrato, utilizzare un saldatore a punta conica finissima e infine, non indugiare troppo sui terminali durante la saldatura, onde evitare di surriscaldare il piccolo microcontrollore. Terminata la saldatura date un'occhiata approfondita con una lente d'ingrandimento, per verificare che non vi siano baffi di stagno a unire piazzole vicine. Per il corretto orientamento dei componenti polarizzati, quindi LED, diodi, transistor e condensatore elettrolitico (ma anche del microcontrollore) seguite il piano di montaggio visibile a pagina precedente. Ricordate di montare sullo stampato dei pin-strip a passo 2,54 mm che vi serviranno per realizzare i ponticelli (per il relé potete lasciare le piazzole libere o prevedere una morsettiera a passo idoneo).



per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono di facile reperibilità. Il master del circuito stampato può essere scaricato dal sito della rivista così come il firmware utilizzato per programmare il microcontrollore PIC10F322.

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

Listato 1

```
void main (void){

    OSCCONbits.IRCF=0b111; //16 MHz INTOSC
    WPUA=0; // PULL UP DISABLED
    TRISA= 0b1001; //RA1, RA2 OUT, RA0, RA3 IN
    LATA=0;
    ANSELA=0b0001; // ADC AN0 ENABLE
    FVRCON=0b10110011; //FVR ON 4.096V TEMP. SENS. ENABLED 4VT
    ADCON= 0b00111001;
    WDTCONbits.WDTPS=0b01101; // WDT 8s

    //TMRO CONFIGURATION
    OPTION_REGbits.T0CS=0; //TIMER 0 WITH FC/4
    OPTION_REGbits.PSA=0; //PRESCALER ON
    OPTION_REGbits.PS=0b111; //PRESCALER 256

    //INTERRUPT TMRO
    INTCONbits.TMR0IF=0;
    INTCONbits.TMR0IE=1;
    INTCONbits.GIE=1;
    __delay_us(500);

    //VARIABLE INITIALIZATION
    BASE=0;
    LED1=0;
    SWITCH=1;

    while(1){
        TEMP(); // READ TEMP INTERNAL SENSOR
        __delay_ms(10);
        TCOMP();
        __delay_ms(10);

        if (SWITCH==0){BASE=1;LED1=1;} //MANUAL ACTIVATION

        //AUTO ACTIVATION
        if ((SWITCH)&&(VADC>VTRIMMER)){__delay_ms(5);BASE=1;LED1=1;} ;}}
        if ((SWITCH)&&(VADC<VTRIMMER)) {if (DELAY==1){BASE=0;LED1=0;TMRO
=0;DELAY=0;counter_OFF=0;}}
        CLRWD();
    }
}
```

IL FIRMWARE

Per l'applicazione termostato trovate nel nostro sito *www.elettronica.in.it*, insieme agli altri file del progetto un esempio di firmware che utilizza un solo LED (LED1), un deviatore (il comune al contatto SW mentre il normalmente aperto e il normalmente chiuso rispettivamente a 5V e GND o viceversa, a seconda della propria applicazione)) e un trimmer per impostare la soglia di temperatura d'intervento del relé e di attivazione del carico da esso commutato. Essendo il firmware un po' lungo, nel **Listato 1** riportiamo soltanto il main

program, tralasciando il resto delle configurazioni.

La nostra demoboard può essere utilizzata per realizzare, oltre che un termostato, un semplice temporizzatore; allo scopo, tra i file del progetto trovate anche il firmware da caricare nel micro per questa applicazione.

Nel caso del temporizzatore, il trimmer permette di impostare la durata e il pulsante avvia il conteggio del tempo; il deviatore permette di stabilire la condizione del relé durante la temporizzazione impostata. Il relé gestisce la corrente che attraversa il carico.

STIMA: MISURAZIONE DI PARAMETRI METEO E AMBIENTALI



Tre nuovi moduli per l'acquisizione e trasmissione al server RMAP dei parametri rilevati, utilizzabili anche per applicazioni domotiche come chiudere una tenda da sole in caso di pioggia.

di PAOLO PATRUNO
e DANIELE CASTELLARI

Dopo avervi spiegato che cos'è il sistema modulare STIMA e come esso si integra nel progetto RMAP (Rete di Monitoraggio Ambientale Partecipativo), in queste pagine presentiamo ulteriori strumenti per acquisire parametri dall'ambiente che ci circonda; abbiamo sviluppato tre moduli con caratteristiche differenti che si adattano a situazioni di installazione e di utilizzo eterogenee per tipo di alimentazione e tipologia di comunicazione dei dati. I moduli presentati in questa puntata sono tre e vanno ad aggiungersi allo STIMA-Bluetooth presentato nella scorsa puntata:



L'OCCORRENTE PER L'APPLICAZIONE

- Microduino core+ 1284p 5V
- Microduino core+ 644p 5V
- Microduino-ENC (RJ45 POE Included)
- Microduino ft232R
- Microduino-GPRS/GSM
- Microduino SDcard
- Microduino-nRF24
- board Stima Bluetooth
- board Stima I2C
- SET CAVI POE
- Display LCD 20x4 con interfaccia I²C
- MODULO 4 RELÈ 5VDC 10A
- MODULO 2 RELÈ 5 VDC 10A
- Modulo 1 relè 5 Vdc 10A

Stima-Master da collegare all'ethernet, Stima-Satellite collegabile via radio al modulo Master, Stima-GSM/GPRS che invia i dati su GSM e salva i dati su SD-Card. Il software che qui presentiamo per la pubblicazione e visualizzazione dei dati è utilizzabile su differenti sistemi operativi: la geolocalizzazione, autenticazione e pubblicazione dei dati rilevati manualmente o a vista è stata testata su Linux, Windows, OS X e Android. In parte, l'utilizzo è stato spiegato nella scorsa puntata; in queste pagine utilizzeremo l'applicazione grafica per la configurazione dei moduli STIMA, compresa la geolocalizzazione, che è una parte fondamentale. Le configurazioni hardware sono molteplici e utilizzano differenti board; sono compatibili con i moduli hardware maggiormente diffusi e conosciuti dai maker e generalmente a basso costo. Qui utilizzeremo la configurazione "impilabile" delle board Microduino. L'unico firmware sviluppato si adatta all'hardware tramite il preprocessore C; molte opzioni quindi sono definite a tempo di compilazione, mentre altre durante l'esecuzione e salvate sulla EEPROM per poi essere rilette all'avvio.

MQTT

I moduli STIMA che qui esaminiamo utilizzano MQTT per pubblicare sul broker rmap.cc i propri dati e metadati. MQTT (acronimo di Message Queue Telemetry Transport) è un protocollo publish/subscribe particolarmente leggero, adatto per la comunicazione M2M tra dispositivi con poca memoria o potenza di calcolo e server (o message broker). HTTP è il protocollo più diffuso, ma dal punto di vista dei device che devono trasmettere dei dati, può invece non essere la scelta migliore: nell'ambito dell'Internet of Things si stanno affermando altri protocolli, soprattutto "binari", che non solo hanno un pattern diverso noto come publish/subscribe, ma soprattutto garantiscono un trasferimento minimo di byte "inutili" rispetto al contenuto informativo. Tra i più noti c'è, appunto, MQTT, nato proprio per la "telemetria", molto leggero ed affidabile (garantisce tre livelli differenti di QoS) soprattutto su reti non perfette in termini di stabilità della connessione. Nello schema publish/subscribe, mittenti e destinatari di messaggi dialogano attraverso un intermediario detto broker: il mittente di un messaggio si limita a "pubblicare" il proprio messaggio sul broker; i destinatari si rivolgono a loro volta al broker "abbonandosi" alla ricezione di messaggi. Quindi il broker inoltra ogni messaggio inviato da un publisher a tutti i subscriber interessati a quel messaggio. Il meccanismo di sottoscrizione consente ai subscriber di precisare a quali messaggi sono interessati tramite un pattern (topic). Pub/Sub disaccoppia un client che sta inviando un particolare messaggio da un altro client (o molti clients), che sta ricevendo il messaggio. Questo significa che

publisher e subscriber non conoscono l'uno l'esistenza dell'altro. Client e broker si scambiano messaggi di polling per monitorare lo stato delle comunicazioni; se il broker non riceve una PINGREQ o qualsiasi altro pacchetto da uno specifico client, chiuderà la connessione e invierà l'ultimo messaggio "will and testament" (se il client ne ha specificato uno). Il meccanismo di keep alive può essere disattivato.

IL FIRMWARE STIMA

Presentiamo ora le caratteristiche generali del firmware utilizzato nei moduli STIMA.

Stazioni fisse o mobili

È possibile installare stazioni sia fisse (la cui posizione non cambia nel tempo) che mobili, sia terrestri che marine; le stazioni sono quelle che acquisiscono i parametri ambientali. Per aggiornare le posizioni delle stazioni mobili viene utilizzato un ricevitore GPS, che può essere o a bordo del modulo Stima o quello integrato in uno smartphone o in un tablet basato su Android.

Differenti tipologie di rete

La configurazione della rete può essere differente a seconda delle esigenze; oltre alla classica configurazione a stella (moduli master e base) con un broker al centro, è disponibile la configurazione ad albero sia via cavo (modulo master + base) che via radio: con la possibilità di utilizzare moduli radio di maggiore potenza (portata di circa 1 km in aria libera) è possibile prevedere coperture di un territorio di ampia superficie.

Salvataggio locale dei dati

I dati possono essere pubblicati in real time e/o salvati localmente. È previsto un meccanismo di salvataggio dei dati su SD-Card

formattata FAT; i file vengono frammentati con una dimensione prefissata per farne circa uno al giorno e numerati da 000 a 999; i dati salvati prevedono un flag che indica se sono stati già pubblicati correttamente su MQTT. I file che devono essere controllati per possibili reinvii hanno estensione **.que**, mentre l'estensione di quelli che hanno tutti i dati già inviati è **.don**. Con tali premesse si ottengono queste funzionalità:

- salvataggio dati su SD almeno per due anni con campionamenti ogni 5 s (un parametro);
- reinvio automatico al server dei dati salvati ma non pubblicati correttamente sul server;
- ottimizzazione dei tempi, in quanto solo i file che contengono dati da inviare vengono letti per selezionare i dati da reinviare ;
- i dati possono essere riletti su un normale PC estraendo la SD-Card.

Messaggistica di diagnostica

C'è la possibilità di ottenere un'ampia messaggistica di diagnostica per la soluzione dei problemi; attiva per impostazione predefinita, può essere disabilitata.

Configurazione

Le versioni delle configurazioni vengono verificate e quando il firmware non è retrocompatibile il modulo resta in attesa di una nuova configurazione. Si può forzare la configurazione tramite un apposito ponticello sulla board Stima-I2C. Le configurazioni vengono subito verificate: non è possibile configurare un modulo con dei sensori non corretti o non funzionanti.

Operazioni periodiche di manutenzione

Il software effettua periodicamente tutte le funzioni di manutenzione necessarie al corretto

funzionamento, quali quelle relative al DHCP o alla sincronizzazione dell'orologio interno con una sorgente esterna, la gestione dei pacchetti per la manutenzione dei protocolli su TCP/IP o via radio. Tutti i firmware hanno attivo un watchdog hardware che evita blocchi permanenti dovuti a malfunzionamenti su eventi improbabili; ogni 8 secondi, quindi, il watchdog deve essere reinizializzato per evitare un reset del microcontrollore.

Orologio di riferimento

Quando è necessario salvare i dati localmente (su SD-Card) e la connessione utilizzata per pubblicare i dati sul server (broker) non è considerata stabile, bisogna disporre di una base dei tempi precisa. Se invece la connessione (trasporto) viene considerata stabile (o non sia necessario recuperare i dati in caso di guasto) un preciso orologio di riferimento non è necessario e il tempo di riferimento verrà aggiunto automaticamente dal server in tempo reale alla pubblicazione del dato. Ci sono diversi sistemi per avere un orologio di riferimento preciso sui moduli Stima.

Crittografia

Qualora il trasporto non sia considerato sicuro (via radio) viene utilizzata la crittografia per garantire riservatezza e autenticità.

Risparmio energetico

Nello sviluppo del firmware, particolare attenzione è stata posta alla limitazione dei consumi: quando possibile, i microcontrollori e i sensori vengono messi in sleep e sono alcuni interrupt a risvegliare il sistema. Questo agevola l'utilizzo con batterie dei sistemi a basso consumo quali il modulo Stima-satellite che funziona con un modulo radio.

Integrazione con la domotica

Per quello che è stato possibile, nello sviluppo di STIMA si è cercato di integrarsi con gli standard della domotica (MQTT). Tutti i moduli possono essere utilizzati anche da attuatori on/off (fino a quattro relé) ma è molto semplice aggiungere altre funzionalità tramite remote procedure in formato json su tutti i trasporti o tramite MQTT.

LIBRERIE

Analizzate le caratteristiche del firmware, vediamo adesso le librerie che esso utilizza e che in parte abbiamo visto nella puntata precedente.

aJson

Questo è un tentativo di portare una completa implementazione di JSON nell'ambiente Arduino; la libreria è basata su cJSON, ridotta di dimensione. Di JSON e JSON-RPC abbiamo parlato nella scorsa puntata in quanto tutte le comunicazioni implementate dal nostro firmware si basano su tale standard. La libreria è stata adattata per ridurre l'uso della memoria e implementare alcuni tipi di dato non supportati.

JsonRPC

JsonRPC è la libreria che implementa un sottoinsieme del protocollo JSON-RPC.

PubSubClient

Fornisce un client per semplici publish/subscribe, scambiando messaggi con un server che supporta MQTT (broker); è adattata per funzionare, oltre che con l'ethernet, anche con GSM/GPRS.

Arduino_uip

Implementa le stesse API della Arduino Ethernet library, ma utilizzando l'ENC28J60 Microchip come chip per la comunicazione

ethernet. Oltre a creare un'implementazione completamente open, infatti, questa libreria permette di utilizzare l'ENC28J60, che è molto più economico dei chip con stack IP incluso. Offre supporto completo per le connessioni TCP e UDP persistenti (streaming) (Client e Server), ARP, ICMP, DHCP e DNS. La libreria è stata sviluppata da Norbert Truchsess derivandola dallo stack uIP di Adam Dunkels.

RF24/RF24Network

OSI Network Layer ottimizzato per moduli radio nRF24L01(+) 2,4GHz ISM. Abbiamo scelto l'ultima versione elaborata da TMRh20, che dalle nostre prove è risultata la più stabile e capace di offrire più opzioni, oltre che di funzionare con Arduino e Raspberry Pi. Supporta una topologia di rete a stella o ad albero; con il modulo da noi utilizzato si possono coprire distanze di circa 50 metri in aria libera. Queste le caratteristiche principali di funzionamento: *Host Addressing* ogni nodo ha un indirizzo logico nella rete local; *Message Forwarding* i messaggi possono essere mandati da un nodo a qualsiasi altro nodo senza limite al numero di "salti" che il messaggio deve fare; *Ad-hoc Joining* un nodo può entrare a far parte della rete senza nessun cambiamento alla configurazione dei nodi già esistenti. Per quanto riguarda *RF24Network indirizzamento e topology*, a ogni nodo deve essere assegnato dall'amministratore un indirizzo a 15-bit. Questo indirizzo descrive esattamente la posizione del nodo all'interno della rete ad albero. L'indirizzo è un numero in formato ottale dove ogni cifra rappresenta una posizione all'interno dell'albero, partendo dalla base:

- Nodo 00 è il nodo base;
- Nodi 01-05 sono nodi il cui genitore è la base;

- Nodo 021 è il secondo figlio del nodo 01;
- Nodo 0321 è il terzo figlio del nodo 021, e così via.

L'indirizzo di nodo più grande è 05555, così in teoria sono possibili fino a 3,125 nodi in singolo canale.

sim800

Questa libreria è stata sviluppata ex novo in quanto le funzionalità di cui c'era bisogno non sono disponibili in nessun'altra libreria di gestione dei moduli GSM SIM800/SIM900 della SIMCom.

La libreria implementa:

- protocollo TCP/IP transparent mode con le API Etherlib;
- http in modalità nativa sim800;
- utilizzo dell'RTC interno alla sim800.

Relativamente a, TCP/IP, una volta che la connessione è stabilita nella modalità trasparente, tutti i dati ricevuti dalla porta seriale saranno trattati per formare pacchetti da essere poi trasmessi, similmente tutti i dati ricevuti dal lato remoto saranno inviati direttamente alla porta seriale.

SensorDriver

Libreria di "driver" per la gestione dei sensori; di essa esistono attualmente due versioni: una in C++ e una in python. La libreria porta la gestione della sensoristica ad un livello di astrazione più alto. In essa, aggiungere un nuovo tipo di sensore consiste nell'estendere una classe con quattro metodi per effettuare la lettura di quello specifico sensore; il comando:

```
int setup (int address);
```

effettua eventuali settaggi necessari al funzionamento del sensore. Ad esempio, per la temperatura la sintassi è: numero di bit di

risoluzione, operazione di misura one-shot. Il comando:

```
int prepare (unsigned long* waittime);
```

impartisce al sensore il comando per effettuare una singola misurazione e ritorna il tempo in millisecondi di attesa necessario.

Invece il comando:

```
int get (int* value);
```

ritorna i valori della misurazione. Il comando:

```
JsonObject* getJson();
```

ritorna i valori misurati in formato compatibile json.

Time/TimeAlarms

Time fornisce un orologio software che può essere sincronizzato con sorgenti esterne per mantenerlo sempre preciso. Nel nostro caso utilizziamo NTP se è disponibile la rete ethernet, l'RTC della scheda sim800 o l'RTC del DS1307 tramite I²C. *TimeAlarms* unitamente a *Time* esegue funzioni a istanti di tempo specificati, una-tantum o periodicamente.

SdFat

SdFat è una libreria per Arduino che supporta FAT16 and FAT32 file systems su SD-Card standard o ad alta capacità (SD/SDHC Flash card). SdFat supporta la creazione di file, cancellazione, read, write, oltre al troncamento. SdFat supporta anche l'accesso a subdirectories, nonché la creazione e cancellazione di subdirectories. Supporta Long File Names e usa la libreria Arduino SPI.

AESLib

Questa libreria implementa la crittografia tramite AES per Arduino; è un metodo per rendere un messaggio "offuscato" in

modo da non essere comprensibile/intelligibile a persone non autorizzate a leggerlo. Questo è particolarmente importante quando si utilizza un trasporto non sicuro quale quello via radio. La crittografia, nel nostro firmware è utilizzata per le comunicazioni con nrf24, ma il livello di sicurezza non è ancora elevato e l'implementazione è incompleta.

YwrobotLiquidCrystal_I2C

La libreria per la gestione del display LCD tramite I²C-Bus possiede tutte le funzioni presenti nella libreria di sistema "LiquidCrystal". Inoltre supporta gran parte delle funzioni listate nella specifica ufficiale per i display LCD.

IL FILE DI CONFIGURAZIONE A TEMPO DI COMPILAZIONE

Il firmware STIMA fa un uso intensivo delle direttive del preprocessore C; definendo delle variabili è possibile ottenere da un unico codice diversi firmware per i vari moduli. Il file *rmap_config.h* situato in *sketchbook/rmap/rmap* comanda quasi tutte le opzioni e le configurazioni hardware; contiene i template da utilizzare per ottenere i firmware per i moduli qui presentati. Non riuscendo, in queste pagine, ad analizzare in modo completo tutto il firmware, ci limitiamo a evidenziare le funzioni principali contenute in *sketchbook/rmap/rmap/rmap.ino*. Nel main loop troviamo:

- *mgrserialjsonrpc*; provvede alla gestione della porta seriale per jsonrpc;
- *mgrrf24jsonrpc*; provvede alla gestione della comunicazione radio e jsonrpc;
- *mgrmqtt*; esegue la gestione del protocollo MQTT e jsonrpc;
- *mgrethserver*; gestisce il server tcp/ip per jsonrpc.

La funzione *mgrjsonrpc* è la funzione utilizzata per gestire il protocollo json-RPC. Se il modulo è attivo periodicamente viene eseguita la funzione *Repeats*, che svolge le operazioni di interrogazione dei sensori e pubblicazione dei dati.

L'HARDWARE DEL PROGETTO

È giunto il momento di descrivere le tre configurazioni proposte in modo da spiegare che tipo di misura possono eseguire e quale connettività possono supportare. Ogni modulo consta di una scheda Microduino e di alcuni shield per essa.

MODULO STIMA MASTER

Questo modulo è ideale per gli appassionati di meteorologia che sognano una stazione meteo fissa, con sensori locali e remoti collegati sia con cavo che via radio. Il Modulo Stima Master può essere dotato di display LCD per il controllo dello stato del sistema e trasmette i dati tramite ethernet al server RMAP. Vediamo da quali board è composto.

Board microduino core+ 1284

La board core+ 1284 è molto simile al modello core+ 644 che abbiamo visto nello scorso numero, solo che è più prestante; l'abbiamo scelta perché il Modulo STIMA Master ha bisogno di un microcontrollore dotato di molta memoria Flash e SRAM per le diverse funzioni che svolge. Sulla board, che rappresenta il modello più evoluto fra le Microduino core, il componente principale è l'Atmega1284P funzionante a 16 MHz, dotato di 128 K di memoria Flash e 16 K di SRAM. Il firmware che vi carichiamo è molto articolato

perché gestisce comunicazioni via LAN e via radio, nonché tutti i protocolli R-map.

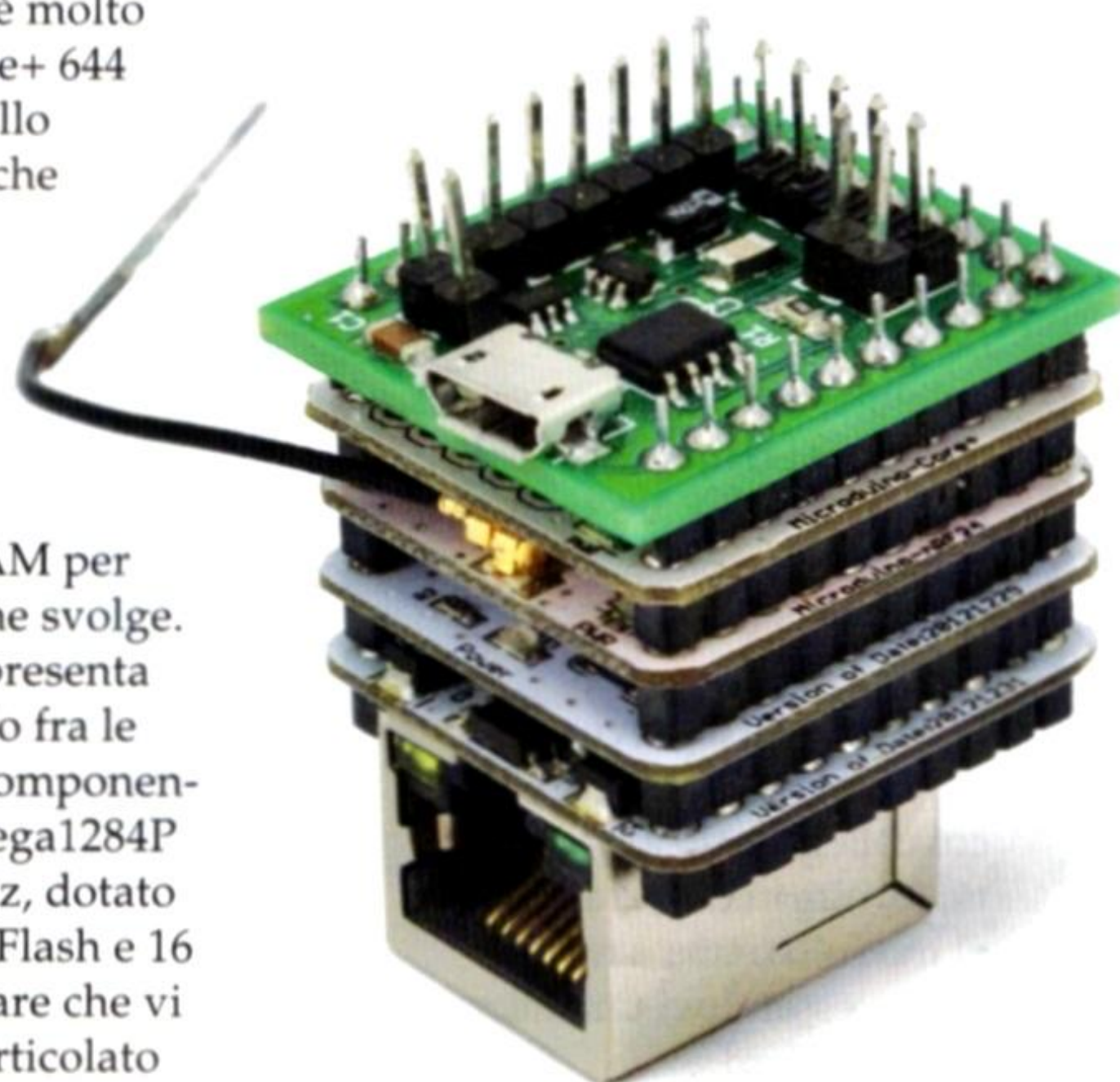
Board microduino ENC

Permette di collegare alla rete cablata il modulo Stima Master; è ancora uno shield Microduino, o meglio, una coppia: una scheda che alloggia il connettore RJ45 e lo stabilizzatore switching per il PoE, ed una seconda sulla quale si trova l'integrato ENC28J60, compatibile con lo standard IEEE 802.3. Questo integrato prodotto da Microchip integra entrambe le funzioni MAC e 10BASE-T PHY e permette una gestione semplificata per il microcontrollore di tutto lo stack di comunicazione Ethernet; anche questa board dialoga tramite la porta SPI.

Microduino ENC ha anche una caratteristica molto interessante, permette l'alimentazione della board e delle altre schede in modalità PoE.

Utilizzando un apposito adattatore (ad esempio quello

STIMA Master al completo.



commercializzato dalla Futura Elettronica con il codice 7300-SETCAVIPOE) ed un alimentatore da 12 a 24 volt (a seconda della distanza del collegamento) possiamo alimentare il modulo tramite un solo cavo, usato sia per la comunicazione Ethernet che per l'alimentazione del modulo. Ciò facilita notevolmente i cablaggi.

Board STIMA-I2C

Questa board l'abbiamo vista nella prima puntata ed è utilizzata da tutti i moduli; ha diverse funzioni: genera 3,3 volt per l'alimentazione, protegge i componenti da scariche elettrostatiche ed espone il bus I2C per il collegamento dei sensori con un comodo connettore.

Board microduino nRF24 (opzionale)

Questa board, che è opzionale, è prodotta da Microduino e progettata per comunicare via radio in banda ISM alla frequenza di 2,4 GHz. Componente principale della scheda è il diffusissimo integrato nrf2401+ prodotto da Nordic Semiconductor, che permette comunicazioni radio fino a qualche decina di metri di distanza in campo aperto. La board permette la ricezione dei dati da moduli STIMA Satellite, oltre che da sensori remoti che possiamo collocare sul terrazzo o in punti non collegabili con cavo.

Sulla scheda è previsto un connettore SMA cui collegare un'antenna fornita in dotazione al modulo o una a nostra scelta, essendovi un attacco di tipo standard.

MODULO STIMA-SATELLITE

Si tratta di un'unità, composta al solito da più schede, che è di tipo "passivo" in quanto rimane in attesa di richieste via radio da un modulo STIMA Master o da uno STIMA-GSM/GPRS; all'arrivo di una richiesta, il modulo interroga i sensori collegati via I2C-Bus e ne trasmette le misurazioni al modulo Master. Il modulo STIMA Satellite si può utilizzare nel sistema solo se sul modulo STIMA Master o GSM/GPRS avete montato la board Microduino nRF24.

Lo STIMA Satellite è composto, oltre che da Board STIMA-I2C (per la quale vale quanto già detto nel modulo Master) e da una board microduino nRF24, da una Microduino core+ 644 che ne costituisce il cervello.

La board nRF24 gestisce i sensori e le comunicazioni radio. È possibile utilizzare più moduli Stima-satellite: fino a cinque che comunicano direttamente con lo STIMA Master, più eventuali

altri disposti in una configurazione ad albero. È necessaria un'opportuna configurazione, ma in questo caso la possibilità di far fare ai dati "salti multipli" rende possibile l'uso di questi sensori in campo aperto su medie distanze, alimentando il modulo con batterie.

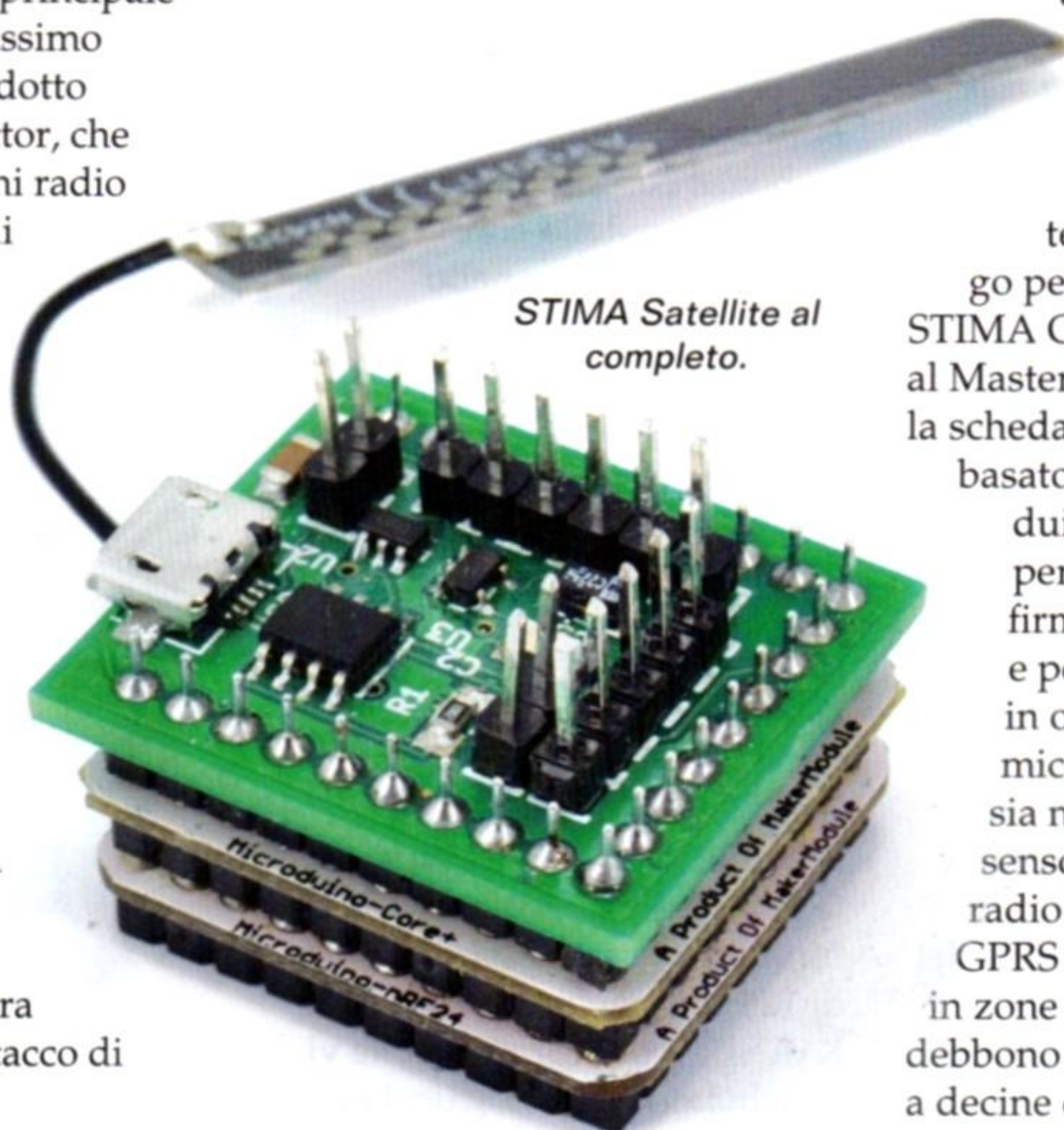
Board microduino core+ 644

Trattandosi di un modulo con funzionalità meno articolate del modulo Master, la board core+ 644 è sufficiente per eseguire le funzioni richieste, perché ha un firmware di dimensioni contenute se non si utilizza la crittografia o altre estensioni software.

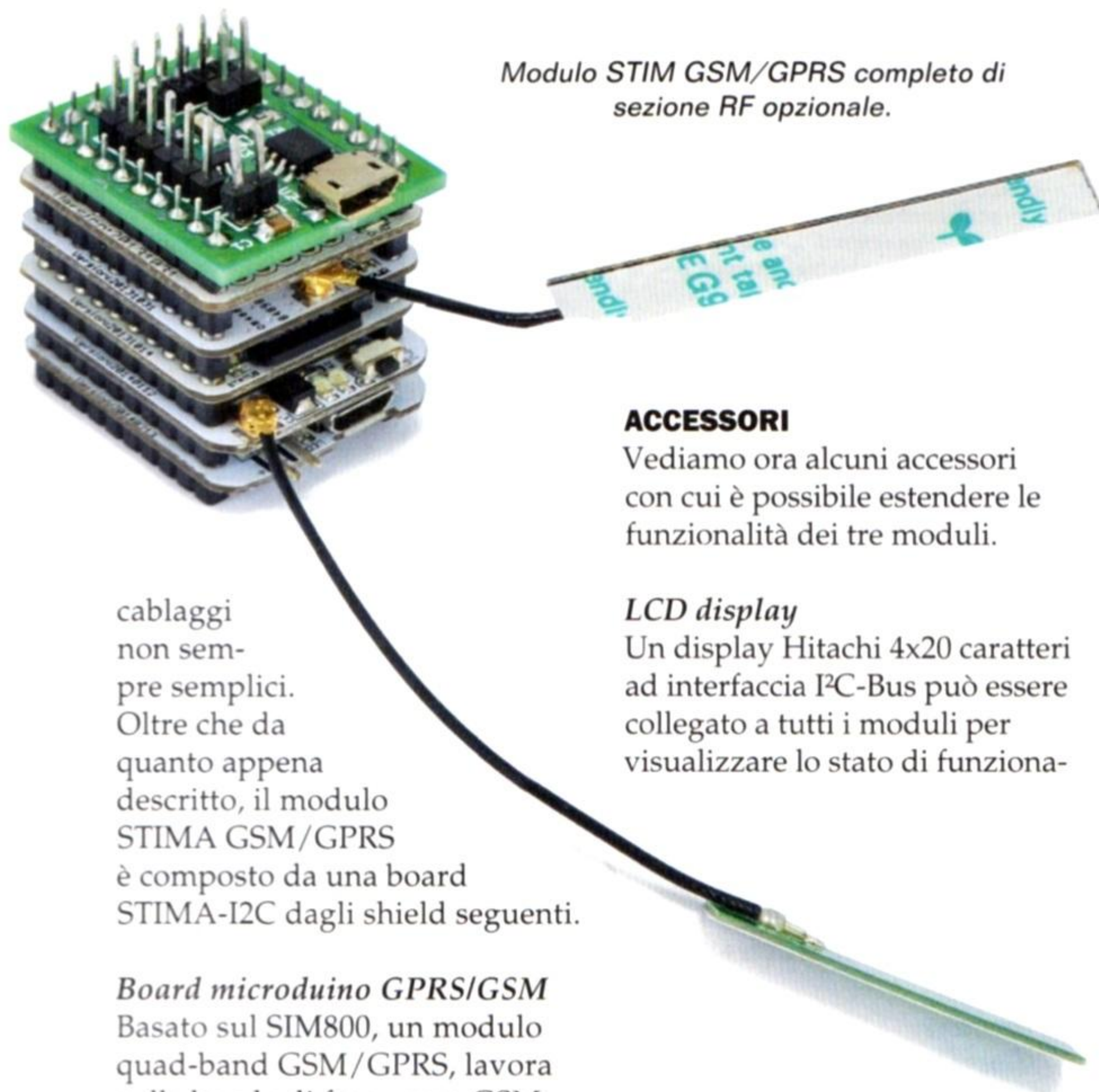
MODULO STIMA GSM/GPRS

Abbiamo analizzato il modulo Stima-Master, che comunica con il server RMAP attraverso rete cablata, ma per utilizzare il sistema STIMA in zone senza connessione ethernet è possibile utilizzare questo modulo con la rete di telefonia mobile.

Vista la minore stabilità di questa rete di comunicazione, il modulo è in grado di salvare permanentemente i dati per un lungo periodo su una SD-Card. STIMA GSM/GPRS è simile al Master per quanto riguarda la scheda Arduino (è infatti basato su una board Microduino core+1284, scelto per eseguire ed ospitare il firmware, che è complesso e pesante) e può disporre, in opzione, di una board microduino nRF24 laddove sia necessario aggiungere sensori remoti collegati via radio. Infatti lo STIMA GSM/GPRS nasce per essere usato in zone isolate, dove i sensori debbono essere montati su pali o a decine di metri di distanza con



Modulo STIM GSM/GPRS completo di sezione RF opzionale.



cablaggi non sempre semplici. Oltre che da quanto appena descritto, il modulo STIMA GSM/GPRS è composto da una board STIMA-I2C dagli shield seguenti.

Board microduino GPRS/GSM
Basato sul SIM800, un modulo quad-band GSM/GPRS, lavora sulle bande di frequenza GSM 850 MHz, EGSM 900 MHz, DCS 1.800 MHz e PCS 1.900 MHz. Il SIM800 è un GPRS multi-slot class 12/class 10 e supporta gli schemi di codifica GPRS CS-1, CS-2, CS-3 e CS-4. Per gestirlo si utilizzano comandi AT impartiti via seriale con un data-rate che può variare tra 1.200 e 115.200 bps. Il transfer-rate della comunicazione su GPRS è di 85,6 kbps sia in uplink che in downlink.

Board microduino SD
Per ridurre i costi di connessione o evitare perdite di dati in caso di mancata connessione al server, il modulo memorizza i dati su SD-Card, che nello specifico è nel formato microSD. Quando la connessione al server viene ristabilita, i dati vengono inviati alla destinazione prevista. La memorizzazione dati su SD consente di salvare dati campionati ogni 5 secondi per un periodo massimo di due anni.

ACCESSORI

Vediamo ora alcuni accessori con cui è possibile estendere le funzionalità dei tre moduli.

LCD display

Un display Hitachi 4x20 caratteri ad interfaccia I²C-Bus può essere collegato a tutti i moduli per visualizzare lo stato di funziona-

mento, la connessione al server, l'ora e i dati rilevati, permettendo una diagnostica immediata.

Relé

Siccome il sistema STIMA è predisposto per la domotica, sulla board Stima-I2C si trova un connettore pronto per collegare moduli da uno a quattro relé pilotabili con JsonRPC. Dato che la scheda equipaggia tutti e tre i moduli, ogni modulo può pilotare degli attuatori.

Sensori

L'ampia possibilità di collegamento dei sensori sul bus I²C verrà esaminata nella prossima puntata; per ora utilizziamo per i test il sensore di temperatura incluso sulla board STIMA-I2C.

COMPILAZIONE E INSTALLAZIONE DEL FIRMWARE

Passiamo adesso al firmware, installando per prima cosa l'IDE

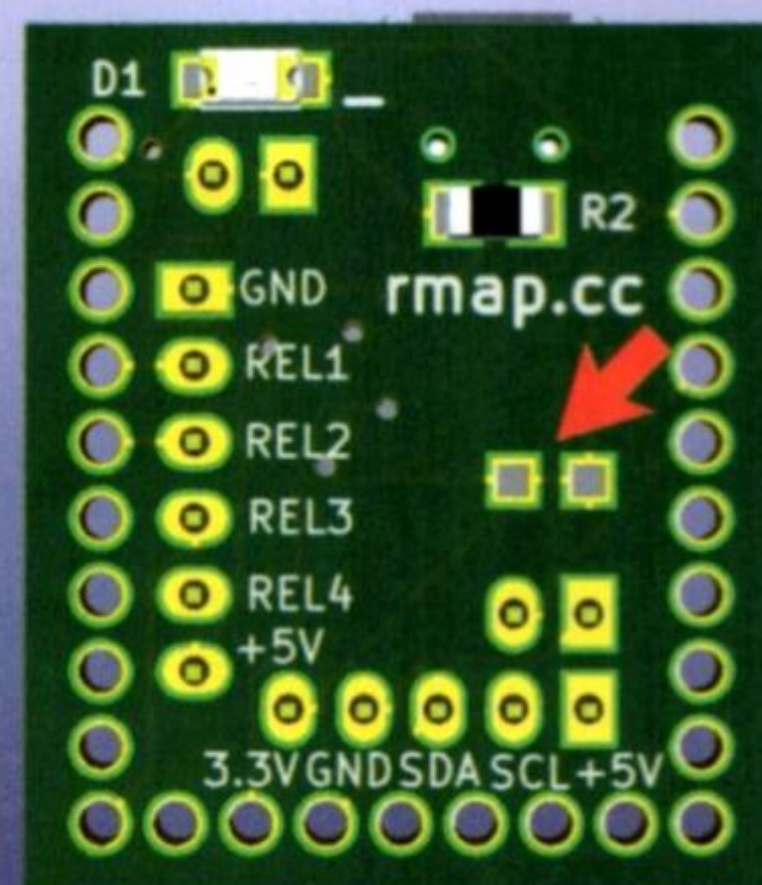


Fig. 1 - Le piazzole SET sono dal lato saldature della board STIMA-I2C.

Arduino 1.6.5 da <https://www.arduino.cc/en/Main/Software>.

Scaricate l'ultima versione del software stima (file stima-sketchbook) da <https://github.com/r-map/rmap/releases> e scompattate il file zip stando attenti a ignorare i file "Source code". Aprite l'IDE Arduino e in File->Impostazioni disabilitate "Controlla aggiornamenti all'avvio" poi in File->Impostazioni->Percorso della cartella degli sketch selezionate la cartella sketchbook appena scompattata dal file scaricato e chiudete e riaprite l'IDE. In questo modo, quando compilerete il firmware utilizzerete alcune opzioni specializzate contenute nel file `sketchbook/hardware/Microduino/avr/platform.local.txt`, necessarie per ottenere un firmware funzionante.

Per quanto riguarda il modulo Stima-master, se in esso avete montato anche la board microduino nRF24 e quindi volete utilizzare anche il modulo Stima-Satellite, con un editor modificate il file `sketchbook/libraries/SensorDriver/SensorDriver_config.h` decommentando l'opzione "#define RADIORF24"; decommentando anche l'opzione "#define AES" abilitate anche la crittografia AES, tuttavia consigliamo l'utilizzo di quest'ultima opzione solo ai più esperti. In `sketchbook/rmap/rmap` dovete copiare il file `stima_master.h` in `rmap_config.h`.

Listato 1

```
rmapctrl --syncdb
rmap-configure --station_slug=ETH_fixed --board_slug=master_eth_fi-
xed --height=<altezza stazione in metri> --stationname="<descrizione
della stazione>" --username=<il vostro utente> --password=<la vostra
password> --server=rmap.cc --samplerate=5 --lat=<latitudine stazione>
--lon=<longitudine stazione> --addsensors_by_template=<sensortemplate>
--wizard --config_station --upload_to_server
```

Poi andate nel menu *Strumenti* e nella voce *Scheda* selezionate *Microduino Core+ (1284pa)*; nella voce *Processore* scegliete *ATmega1284pa@16M5V* e in *Porta* impostate la COM disponibile. Adesso, impartite il comando *Strumenti->Cartella degli sketch->rmap->rmap*; infine, impartite il comando *Sketch->Carica*.

Passiamo adesso alla configurazione per il modulo STIMA Satellite: in *sketchbook/rmap/rmap* copiate il file *stima_satellite.h* in *rmap_config.h*. A questo punto dal menu *Strumenti* scegliete:

- *Scheda: Microduino Core+ (644pa)*;
- *Processore: ATmega644pa@16M5V*;
- *Porta: quella disponibile*.

Impartite quindi il comando *Strumenti->Cartella degli sketch->rmap->rmap* e poi *Sketch->Carica*. Concludiamo con la configurazione per il modulo STIMA GSM/GPRS: in *sketchbook/rmap/rmap* copiate il file *stima_gsm.h* in *rmap_config.h*; il file è predisposto per l'utilizzo di una SIM-Card della TIM. Laddove ne utilizzate una di diverso operatore, inserite in fondo al file *rmap_config.h*:

```
#define GSMAPN ""
#define GSMUSER ""
#define GSMPASSWORD ""
```

scrivendo in GSMAPN l'APN fornito dal vostro operatore, in GSMUSER lo username con cui accedete a Internet e in GSMPASSWORD la password per l'accesso. Salvate quindi il file. Ora aprite il file accessibile da

sketchbook/libraries/PubSubClient/PubSubClient.h e modificate:

```
//if use sim800 client
#include "sim800Client.h"
#define TCPCLIENT sim800Client
```

con questa porzione di codice:

```
//if use arduino_uip or etherclient
#include "Client.h"
#include "Stream.h"
#define TCPCLIENT Client
```

Nel menu *Strumenti* scegliete:

- *Scheda: Microduino Core+ (1284pa)*;
- *Processore: ATmega1284pa@16M5V*;
- *Porta: quella disponibile*.

Sempre in *Strumenti*, impartite il comando *Cartella degli sketch->rmap->rmap*. Infine, impartite il comando *Sketch->Carica*.

CONFIGURAZIONE

Per configurare le board dovete utilizzare l'applicazione grafica o l'utilità a linea di comando. Come avere il software installato sul proprio computer lo abbiamo spiegato nella scorsa puntata, ma le istruzioni si possono anche recuperare sul web da http://www.raspibo.org/wiki/index.php/Gruppo_Meteo/HowTo.

Poi per pubblicare i dati sul server RMAP.cc bisogna registrarsi al sito; il pulsante "Registrazione" dell'app dovrebbe aprire un browser alla url della registrazione (<http://rmap.cc/registrazione/register/>). Una volta fatta la registrazione sarete in possesso di uno user e di una password.

A questo punto dovreste trasferire

la vostra configurazione sulla EEPROM del microcontrollore e allo scopo:

- ponticellate sulla board Stima-I2C le piazzole "SET" (Fig. 1).
- collegate il modulo con la board Microduino FT232RL alla USB del vostro PC.

Configurazione tramite l'applicazione grafica

Aprirete l'applicazione grafica e dal menu *Impostazioni* che si aprirà automaticamente al primo avvio, nella sezione "Rmap" dovrete inserire "RMAP user" e "RMAP password" ottenuti durante la registrazione a rmap.cc.

Adesso, dal menu *Impostazioni* selezionate la sezione "Sensors" e impostate:

- per il modulo Stima-Bluetooth:
 - Name: HC-05;
 - Station: BT_fixed;
 - Board:
 - su android: BT_fixed;
 - su linux: BT_fixed_LINUX;
 - su windows: BT_fixed_WINDOWS;
 - su OSX: BT_fixed_OSX.
 - Template: test_indirect;



Fig. 2

- Remote Board: stima_bt;
- Remote Template: test ;
- per il modulo Stima-Master o Stima-GSM/GPRS:
 - Station: ETH_fixed;
 - Board:
 - su linux: rmapgui_LINUX;
 - su windows: rmapgui_WINDOWS;
 - su OSX: rmapgui_OSX.
 - Template: test_indirect;
 - Remote Board: master_eth_fixed;
 - Remote Template: test (test_master se avete la board nRF24).

Per il modulo Stima-Satellite fate come per lo Stima-Master, ma in Remote Board scegliete satellite_eth_fixed. Nella sezione "Location" potete inserire manualmente le vostre coordinate e selezionare "Close" attivando così la stazione. Se non conoscete le vostre coordinate, dalla pagina "Posizione" selezionate accuratamente la vostra posizione con il mouse e salvatela con il pulsante "Salva posizione" (Fig. 2). La prima pagina dell'App "Avvia" presenta un manuale che potrà aiutarvi. Dalla pagina "Dati automatici" fate clic sul pulsante "configura" e verificate che tutto vada a buon fine.

Configurazione a linea di comando

È possibile eseguire tutte le configurazioni con due comandi da terminale: da Windows utilizzate la finestra che rimane aperta dopo aver eseguito `rmap-configure.bat`; su Linux o OSX attivate il Virtualenv di python come per eseguire l'App grafica. Il primo comando inizializza il DB. Il secondo, con l'opzione `--wizard` inserisce tutti i metadati nel DB, `--config_station` trasferisce la configurazione sulla EEPROM del microcontrollore

e `--upload_to_server` comunica i metadati al server (Listato 1). Ovviamente sostituite il contenuto tra <> con i vostri dati; cercate di avere latitudine e longitudine definite fino alla quinta cifra decimale; <sensor-template> dovrà essere "test_master" se il vostro modulo Stima-master o Stima-GSM/GPRS comprende una board microduino nRF24, "test" in tutti gli altri casi.

Operazioni finali

Scollegate il modulo dalla USB, rimuovete la board Microduino FT232RL e attivate l'alimentazione a 3,3V chiudendo il ponticello JP2 della board Stima-I2C.

MESSA IN OPERA

Sul modulo Stima-GSM/GPRS inserite una micro SD formattata FAT; dovete inserire anche una SIM-Card. Alimentate i moduli tramite il connettore micro-USB sulla board Stima-I2C; il modulo Stima Master potete alimentarlo con l'apposito Injector e un alimentatore da 12 a 24V (connettore con positivo al centro). Sul modulo Stima-GSM/GPRS il SIM800 va acceso manualmente tenendo premuto l'apposito switch. Sulla rete ethernet dovrete avere un server DHCP in quanto STIMA Master lo richiede. Se tutto funziona regolarmente, ogni 5 secondi i dati della temperatura di test del modulo verranno inviati a `rmap.cc`. A questo punto se sul server `rmap.cc` fate login con il vostro nome utente, accederete alla vostra pagina personale con l'elenco delle stazioni di misura e la possibilità di visualizzare i grafici (Fig. 3) dei vostri dati. Per ora potete inviare solo i dati di test (temperatura del modulo); nella prossima puntata vedrete come aggiungere sensori per la visualizzazione dei dati.

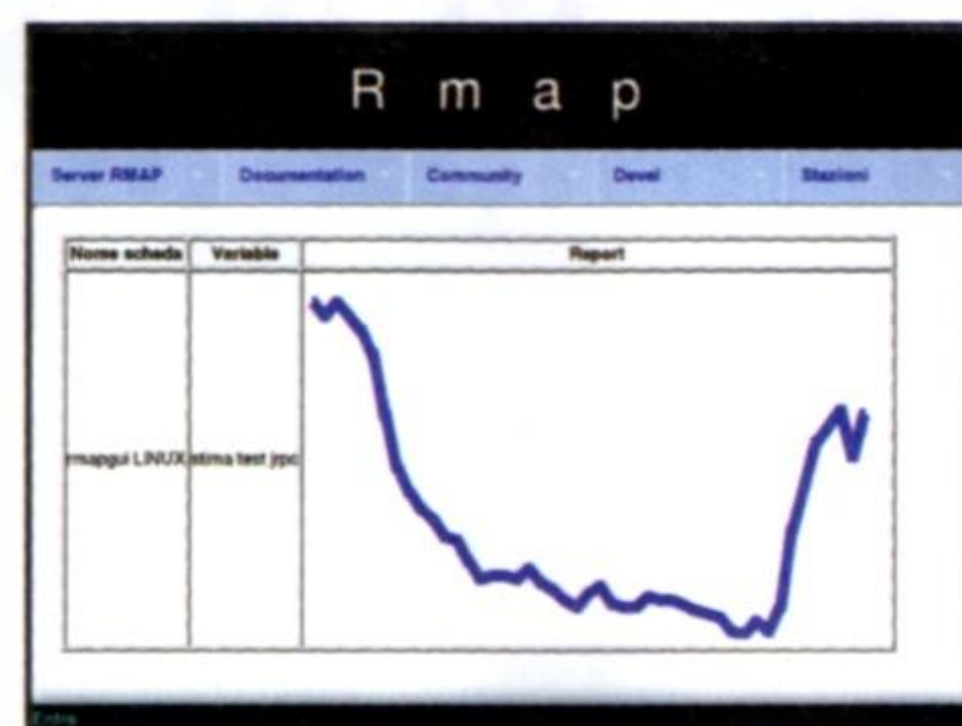


Fig. 3



per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono di facile reperibilità. I master dei circuiti stampati possono essere scaricati dal sito della rivista. I componenti elencati di seguito si possono acquistare presso Futura Elettronica:

- Microduino Core ATMEGA1284 (cod. MDUINO1284) a Euro 22,00.
 - Microduino Core+ ATMEGA644PA (cod. MDUINOCORE+) a Euro 15,50.
 - Microduino Shield USB/TTL (cod. MDUINOUSBTTL) a Euro 19,00.
 - Microduino Shield SDcard (cod. MDUINOSD) a Euro 12,50.
 - Microduino Shield NRF24 (cod. MDUINONRF24) a Euro 16,00.
 - SET CAVI POE (cod. SETCAVIPOE) a Euro 6,00.
 - Display LCD 20x4 con interfaccia I²C (cod. LCD20X4AI2C) a Euro 26,00.
 - MODULO 4 RELÈ 5VDC 10A (cod. RELAY4CH) a Euro 13,00.
 - MODULO 2 RELÈ 5 VDC 10A (cod. RELAY2CH) a Euro 8,00.
 - Modulo 1 relè 5 Vdc 10A (cod. RELAY1CH) a Euro 5,00.
- Tutti i prezzi si intendono IVA compresa.

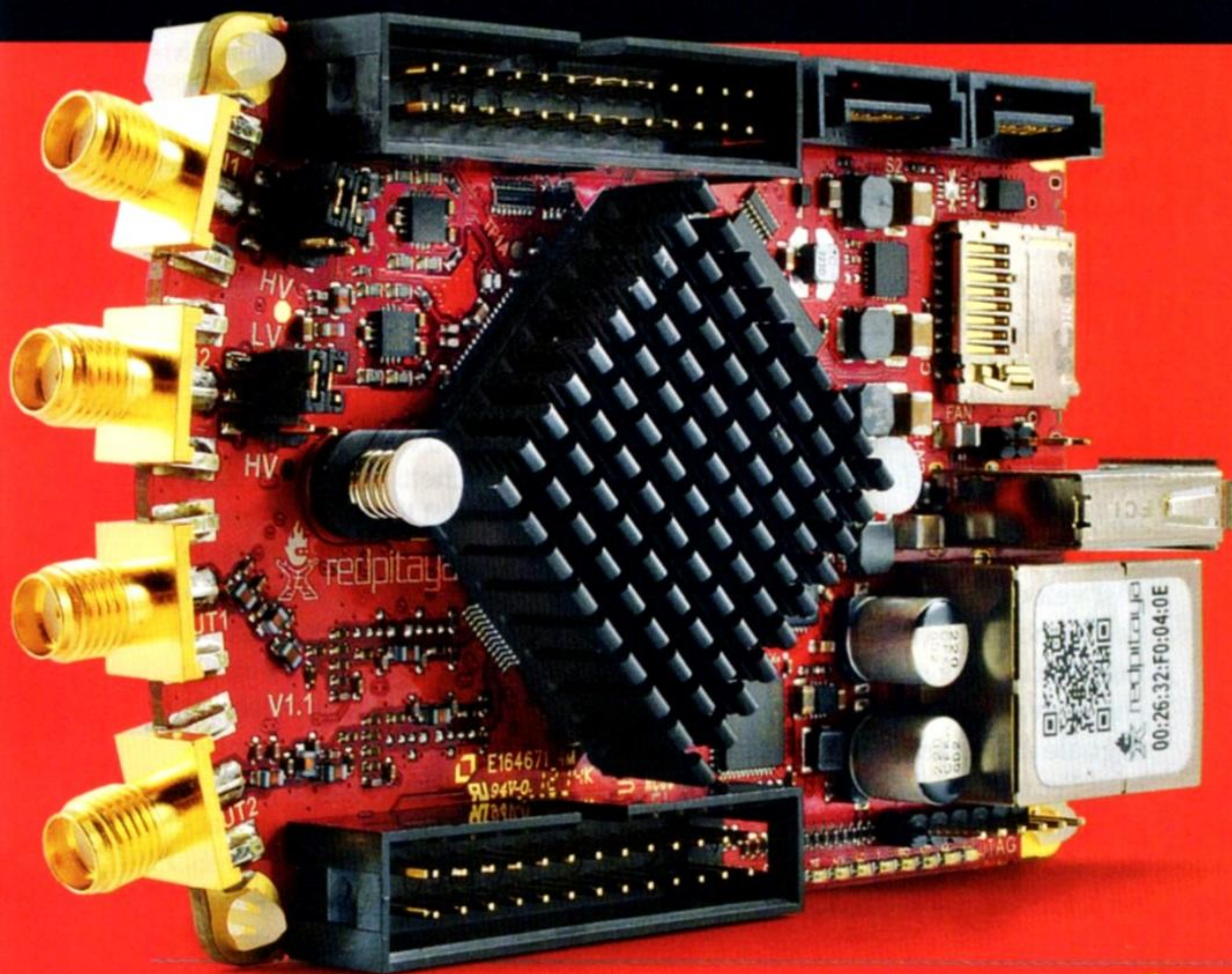
Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)

Tel: 0331-799775 - Fax: 0331-792287
<http://www.futurashop.it>

Un'insolita applicazione della scheda nata come strumento di misura riconfigurabile, che ci mostra come grazie alla FPGA si può ottenere una Software Defined Radio.

RED RADIO PITAYA

di RENZO DAVOLI, RD235, IZ4DJE



Tanto tempo fa, un oscilloscopio e una radio ricetrasmittente erano due oggetti diversi e avevano circuiti elettronici differenti. Oggi non più. Questo articolo vi condurrà attraverso vari percorsi di lettura, sperimentazione e apprendimento. Fin da quando è apparsa Red Pitaya, l'autore di questo articolo ha desiderato renderla una radio e avendo letto le caratteristiche tecniche dello strumento era certo che fosse possibile; grazie a Pavel Demin, un ricercatore Fisico dell'Università Cattolica di Leuven (Belgio) è passato

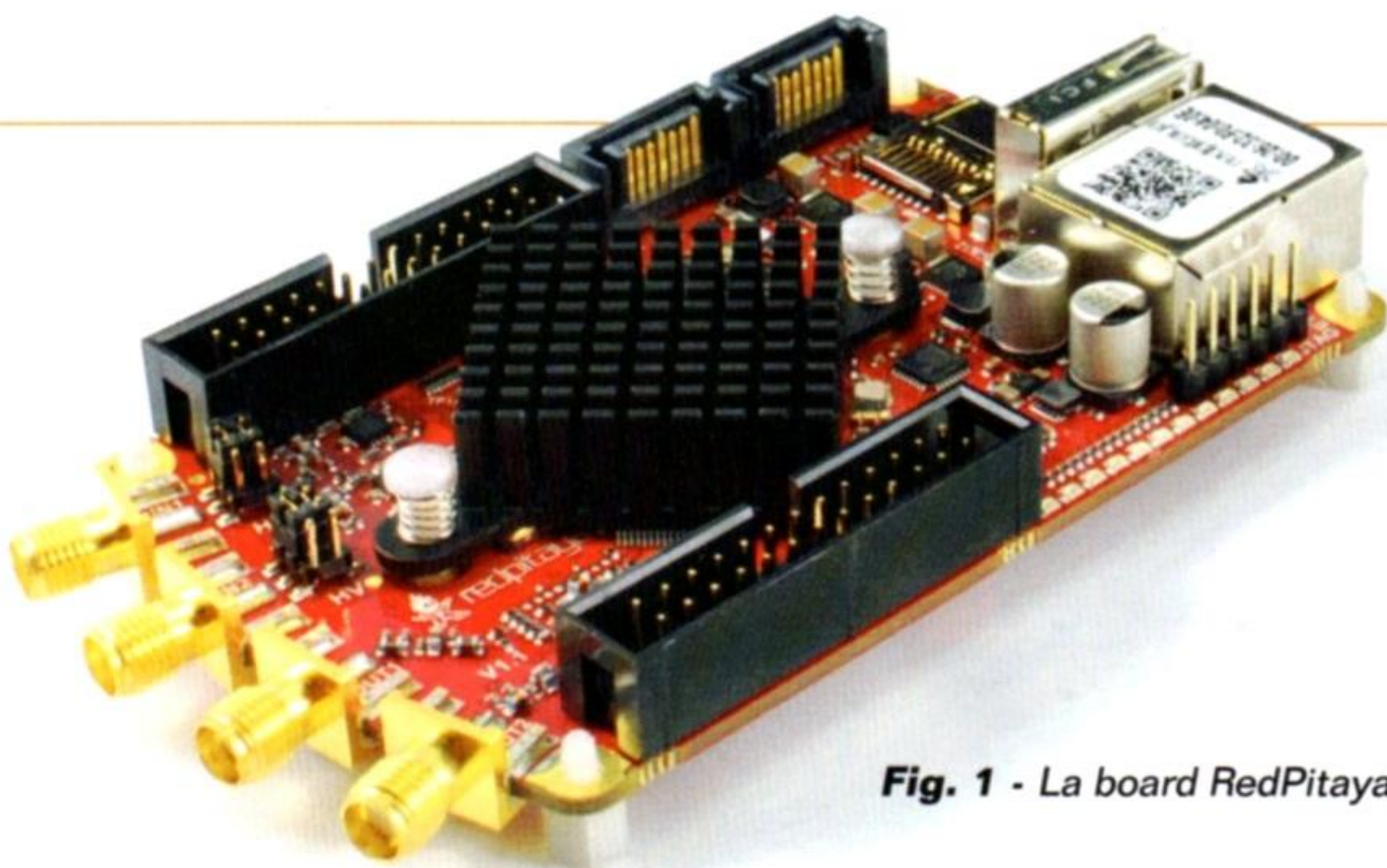


Fig. 1 - La board RedPitaya.

dall'idea all'azione. La scrittura, il debugging e la sperimentazione del software, sono stati suggestivi e divertenti, come l'uso della nuova, insolita radio che ne è nata. Red Pitaya è una scheda che viene commercializzata come un laboratorio elettronico open-source (come dice il titolo dell'articolo pubblicato nel fascicolo di maggio 2014 di *Elettronica In*). Questa scheda monta due convertitori Analogico-Digitale (ADC) e due convertitori Digitale-Analogico (DAC) in grado di lavorare alla velocità di 125 milioni di campioni al secondo e con una risoluzione di 14 bit.

Flussi così imponenti di dati, per essere elaborati in modo tradizionale richiederebbero processori potentissimi: i quattro convertitori a pieno regime generano un traffico complessivo di 875 milioni di byte al secondo. Per questo il SoC della scheda (Xilinx Zynq 7010) ha a bordo un Field Programmable Gate Array (detto, più familiarmente, FPGA).

Questo componente ha all'interno un alto numero di porte logiche (28 mila nel nostro caso) che possono venir collegate fra loro via software: è come se avessimo 28 mila porte logiche fisiche e creassimo circuiti collegando con ponticelli l'output di ogni porta all'input di un'altra. In un FPGA, l'intera mappa dei ponticelli può essere modificata con un comando e senza l'uso del saldatore. La radio che andremo a realizzare

con Red Pitaya è una Software Defined Radio (SDR), cioè una radio fatta quasi interamente da programmi. In particolare, la radio sarà a conversione diretta: sarà sufficiente collegare l'antenna ad uno degli ingressi per poter ricevere le trasmissioni operanti a frequenze minori di 60 MHz.

Gli "ingredienti" per realizzare un ricevitore radio sono: un'antenna, la scheda Red Pitaya e un Personal Computer (e i cavi per collegare Red Pitaya all'antenna e al PC). Negli esempi dell'articolo, il PC usa il sistema operativo GNU-Linux, distribuzione Debian 8 (jessie). Questi ingredienti consentiranno anche di poter trasmettere con Red Pitaya, ma questo lo analizzeremo in un secondo tempo. Possiamo paragonare questo tipo di SDR alla scheda audio. Tutti noi usiamo i programmi di registrazione o di riproduzione audio (Audacity, per esempio).

La scheda audio *campiona* il segnale audio: decine di migliaia di volte al secondo legge i *volt* in ingresso e genera così una sequenza di numeri che forniscono una rappresentazione approssimata del segnale audio. Red Pitaya è così veloce, sensibile e precisa che possiamo darle in input direttamente il debole segnale elettrico che arriva dall'antenna. Come nel segnale audio arrivano mescolati tutti i rumori ambientali, le voci o gli strumenti di un'orchestra, così nel segnale dell'antenna vengono a trovarsi i segnali di tutte le

emittenti che stanno inducendo una debole corrente nell'antenna, la quale altro non è che un filo elettrico sospeso di lunghezza opportuna per risuonare con le correnti indotte da trasmettitori lontani. In un ricevitore radio tradizionale ci sono molteplici circuiti per la conversione di frequenza, il filtraggio, la discriminazione/selezione del singolo segnale, il controllo automatico di guadagno e la demodulazione. Le stesse funzioni sono presenti in una radio SDR, ma vengono svolte da algoritmi, da programmi che svolgono i calcoli di opportune funzioni matematiche.

La radio che andremo a realizzare è, per così dire, *distribuita*. Una prima elaborazione verrà effettuata dall'FPGA e dal processore di Red Pitaya; i dati verranno quindi trasferiti al PC dove avverrà la parte finale del processo di elaborazione. Esistono più programmi *client* con caratteristiche diverse, che possono essere usati per questa fase finale. Nell'articolo si farà riferimento a GNU radio e a gqrx.

Gqrx è una radio classica: la sua interfaccia consente di impostare la frequenza e la modulazione, anche individuando il segnale desiderato su una porzione dello *spettro*.

GNU radio è invece un toolset, una cassetta degli attrezzi, che consente di costruire radio anche tramite il gnuradio-companion, un editor grafico dove i componenti logici vengono collegati come se fossero blocchetti delle costruzioni o istruzioni del linguaggio Scratch. Il modo più semplice (e flessibile) per trasformare Red Pitaya in una radio è di procurarvi una seconda scheda microSD: in questo modo potrete decidere se usare Red Pitaya come radio o come oscilloscopio/generatore di segnali semplicemente decidendo all'accensione quale delle due microSD usare (ovviamente per cambiare

fra radio e oscilloscopio e viceversa occorre spegnere Red Pitaya, staccare l'alimentazione, mettere l'altra microSD e riaccendere). Il file da caricare nella microSD li potete scaricare dal nostro sito www.elettronica.in insieme agli altri file del progetto; si tratta di un file zip, che è sufficiente decompattare nella scheda di memoria. I comandi shell per l'operazione (una volta inserita la scheda microSD in un lettore collegato al PC) sono:

```
$ cd /tmp
$ wget http://www.raspibo.org/redpitaya/sdr-transceiver.zip
$ sudo mount /dev/xxxx /mnt
$ cd /mnt
$ sudo unzip /tmp/sdr-transceiver.zip
$ cd /tmp
$ sudo umount /mnt
```

Al posto di xxxx occorre mettere il device relativo alla scheda microSD, che può essere /dev/mmcblk0p1, oppure /dev/sdc1, /dev/sdd1. Occorre essere certi del nome del dispositivo, perché se indicate il device errato potete cancellare i dati del vostro disco rigido. A questo punto potete togliere la scheda microSD dal lettore del computer e metterla nella Red-Pitaya. Se ora fate il boot, alla scheda automaticamente viene impostato l'FPGA con il *circuito virtuale* per la radio e vengono attivati i server per la comunicazione con i programmi client da attivare sul PC. Il primo problema è trovare l'indirizzo di Red Pitaya: potete applicare tutte le tecniche descritte nel manuale di Red Pitaya consultabile sul web all'indirizzo http://wiki.redpitaya.com/index.php?title=User_Manual e descritte nell'articolo del maggio 2014 di Elettronica In. Se la rete locale alla quale è collegata Red-Pitaya al momento del boot non è connessa a

Internet, avete due possibilità: accendere Red Pitaya con il cavo di rete scollegato e collegarlo in un secondo tempo (l'indirizzo predefinito è 192.168.1.100); negli esempi seguenti ipotizzeremo che Red-Pitaya sia collegata a questo indirizzo. La seconda opzione è accedere all'indirizzo di Red Pitaya con un browser: scoprirete che sono *sparse* tutte le applicazioni e compare una pagina vuota. Questo perché le applicazioni standard quali oscilloscopio e generatore di segnali sono state tolte dal sistema, sarebbero incompatibili con la configurazione caricata nell'FPGA con le funzionalità necessarie a costruire la radio.

Se volete tenere Red-Pitaya acceso per lungo tempo come radio, consigliamo di montare una ventola sull'aletta di raffreddamento del processore: noi ne abbiamo usata una a 5 volt fissata sul dissipatore con viti da 2,5 mm avvitate tra le alette. A riguardo notate che c'è una presa apposita per alimentare una ventola fra la presa USB di tipo A (quella verticale) e lo slot per la MicroSD (**Fig. 1**).

Ora possiamo fare il primo espe-

rimento: la prima radio sarà una semplice ricevente AM costruita con GNU Radio. Installate nel vostro PC il supporto per GNU radio:

```
apt-get install gnuradio
```

a questo punto scaricate i moduli GNU-Radio per Red Pitaya e il nostro primo ricevitore:

```
$ wget http://www.raspibo.org/redpitaya/rp_gnuradio.tgz
$ tar zxvf rp_gnuradio.tgz
$ cd rp_gnuradio
$ GRC_BLOCKS_PATH=. ./rx_am.py
```

Se funziona tutto correttamente vi comparirà una semplice interfaccia grafica che mostra uno slider e un campo numerico per sintonizzare la frequenza da ricevere. Nella parte bassa della finestra c'è un analizzatore di spettro che mostra il *panorama* dei segnali attorno alla frequenza indicata (**Fig. 2**). Se non avete collegato alcun'antenna, a questo punto l'unica cosa che potete sentire è un brusio di fondo. Collegando un'antenna potete iniziare a sentire segnali in onde medie e corte.

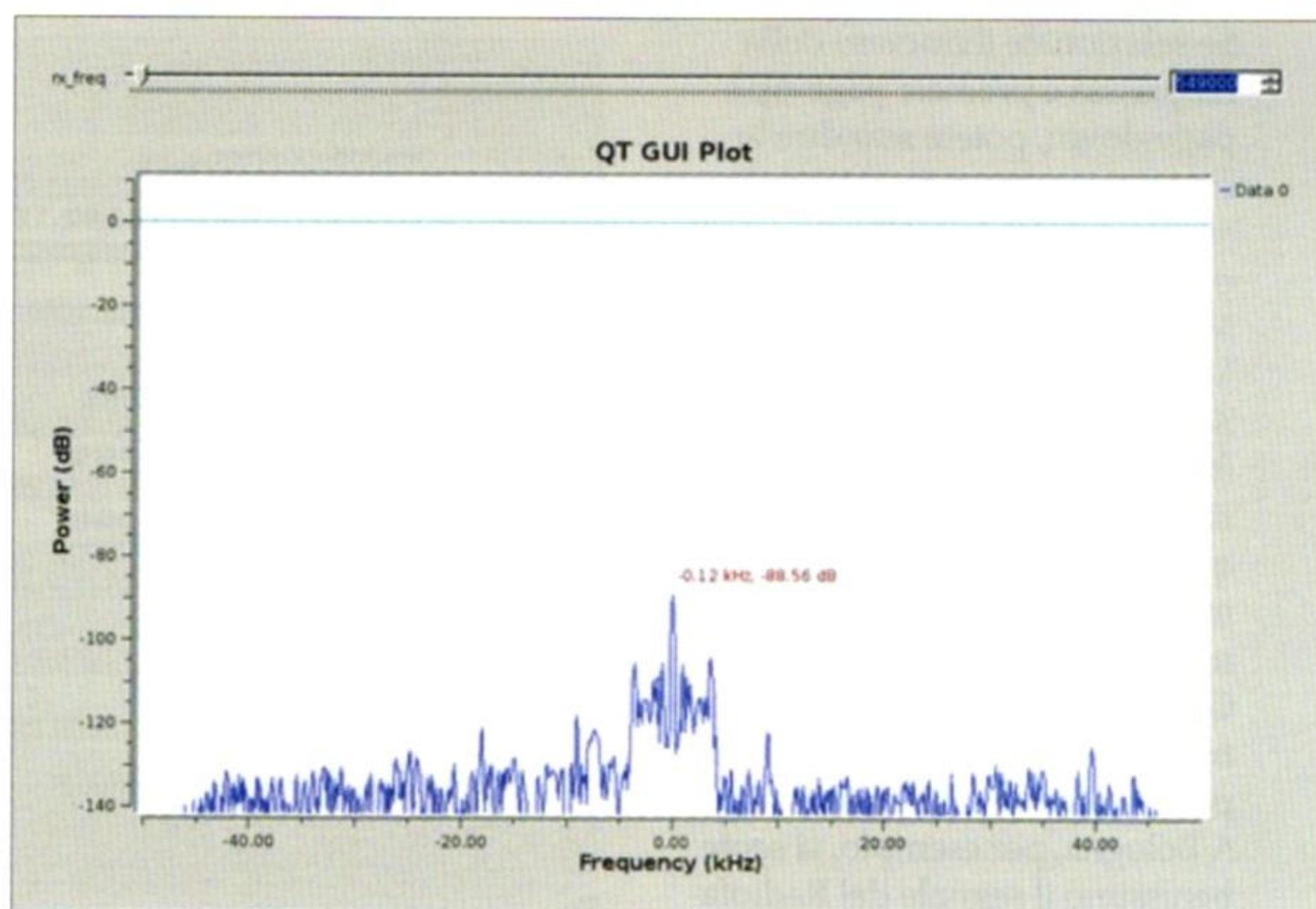


Fig. 2 - Grafico della frequenza.

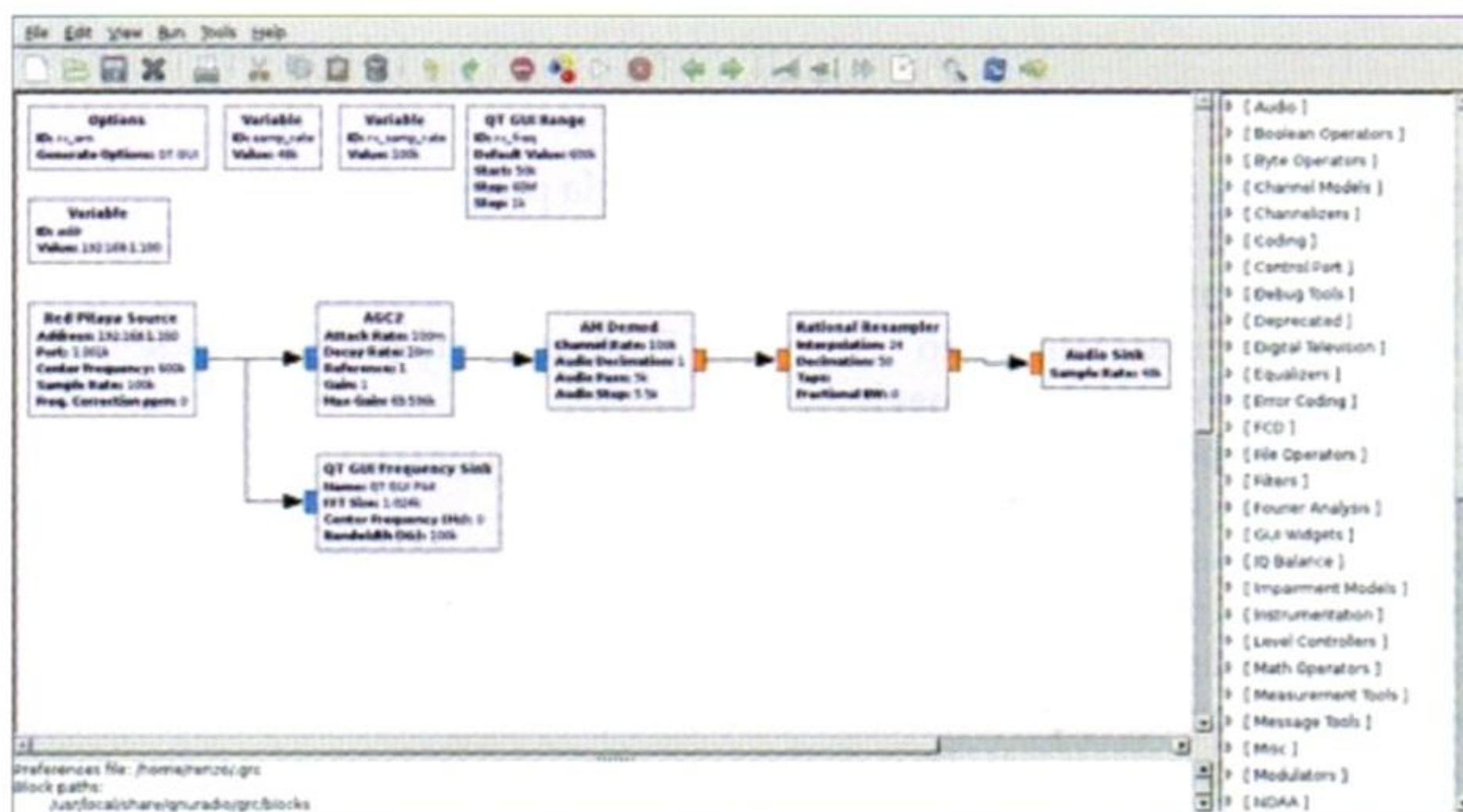


Fig. 3 - Schema a blocchi.

Come antenna potete inizialmente usare un lungo filo collegato al pin centrale della presa IN1 e collegare a terra la massa (ad esempio collegandola ad un termosifone di casa). Se vorrete ricevere veramente tanti segnali vi consigliamo di installare un'antenna per onde corte e collegarla alla presa IN1. Se siete radioamatori, l'antenna ce l'avrete sicuramente già sul tetto (o ne avrete cento autocostruite che vagano per il vostro laboratorio). L'interfaccia è semplicissima: potete muovere avanti e indietro lo slider, o se conoscete frequenze di trasmettitori, potete inserirle nella finestrella numerica.

Se selezionate il numero della frequenza e premete page-up e page-down, potete scandire le frequenze a passi di 10 kHz. Potete ascoltare le tante emittenti a onde corte, ad esempio nel range fra 9,4 MHz e 9,9 MHz, oppure fra 11,6 MHz e 12,2 MHz.

Se sintonizzate emittenti con forti segnali, inizialmente possono risultare distorte, ma attendendo qualche secondo il controllo automatico di guadagno provvederà a migliorare la ricezione.

Gironzolandolo per le frequenze potrete trovare segnali strani, scampanellanti o segnali morse.

A Bologna, per esempio, si sente benissimo il segnale del Radiofaro dell'aeroporto a 413 kHz, che

continua a emettere il messaggio BOA in morse. Se vivete vicino ad un aeroporto potete vedere la frequenza e il messaggio da ricevere cercando "Non-directional beacon" sulla Wikipedia italiana.

Ma ciò che è più divertente di questo esperimento è vedere *dentro* questa radio fatta di software, grazie a questi comandi:

```
$ export GRC_BLOCKS_PATH=.
$ gnuradio-companion rx_am.grc
```

Si vede lo schema a blocchi di questa semplice radio (Fig. 3): il segnale che arriva da Red Pitaya

viene mandato in input sia al controllo di guadagno (AGC) sia al demodulatore AM; poi c'è un ricampionatore per adattare il segnale alla frequenza di lavoro della scheda audio.

Il programma python che abbiamo lanciato prima come *radio*, *rx_am.py*, non è stato scritto da un programmatore ma è stato generato automaticamente a partire da questo schema. È un primo piccolo esempio di uso di GNU Radio. È possibile inserire e configurare i blocchi prendendoli dall'elenco posto sul lato destro della finestra di dialogo e collegarli fra loro come elementi di un meccanismo logico. Lo schema viene compilato in un programma python che può essere eseguito.

Nella directory è presente anche il programma *am_trx.grc* e il relativo *am_trx.py*, che è un ricetrasmittitore AM. Il trasmettitore non ha un finale di potenza, quindi la massima potenza emessa è di 80 mW circa (ricordate che per trasmettere in onde corte occorre avere la licenza di Radioamatore). Senza amplificazione, il segnale è ricevibile a decine o centinaia di

Fig. 4

metri: dipende dall'antenna collegata alla presa BNC OUT1. Come secondo esperimento vogliamo provare a usare Red Pitaya come ricevitore radioamatoriale vero, invece che come esperimento didattico.

Per questi abbiamo lavorato per rendere Red Pitaya compatibile con molti ricetrasmittitori SDR liberi. In particolare, c'è una versione sperimentale di qsdrr che supporta Red Pitaya. È stato scritto anche un modulo della libreria **gr-osmosdr** per consentire a tutti i programmi che usano questa libreria di collegarsi a Red Pitaya. Siamo in contatto con gli sviluppatori per inserire la compatibilità con Red Pitaya come caratteristica standard dei loro programmi. Il supporto di Red Pitaya è già parte integrante della libreria **gr-osmosdr** nella versione sperimentale disponibile nel loro repository git. La versione attualmente presente nei pacchetti Debian la 0.1.4, non ha ancora questa caratteristica. Per poter usare Red Pitaya occorre quindi compilare la versione presa da git. Useremo il ricevitore gqrx e la

libreria **gr-osmosdr** modificata perché è il percorso più semplice: molti componenti sono già disponibili nella distribuzione Debian. Installiamo gqrx (**gr-osmosdr** viene installata come dipendenza):

```
$ sudo apt-get install gqrx-sdr
```

Ora occorre scaricare e compilare il modulo per Red Pitaya:

```
$ git clone https://github.com/osmo-com/gr-osmosdr
$ cd gr-osmosdr
$ mkdir build
$ cd build
$ cmake ..
$ make
$ cd lib
$ ln -s libgnuradio-osmosdr.so.0.1.5git libgnuradio-osmosdr.so.0.1.4
$ LD_LIBRARY_PATH=.:$LD_LIBRARY_PATH
gqrx
```

Il supporto di Red Pitaya è già parte integrante della libreria **gr-osmosdr** nella versione sperimentale disponibile nel loro repository git. La versione attualmente presente nei pacchetti Debian la 0.1.4, non ha ancora questa caratteristica. Per poter usare Red Pitaya occorre quindi compilare la versione presa da git.", per-

ché gqrx nella distribuzione è stato erroneamente linkato dinamicamente alla revisione della libreria 0.1.4 invece che alla versione 0.1. Se non avete mai lanciato qsdrr dovrebbe comparire la finestra di configurazione. Se non comparisse, si apre selezionando "I/O Devices" nel menu "File".

Per collegarvi al vostro RedPitaya occorre scegliere: Device=Other..., Device string "redpitaya=192.168.1.100:1001" e come Sample rate occorre indicare 500000 (Fig. 4).

Ed ecco, in Fig. 5, gqrx in azione mentre mostra l'intera banda radioamatoriale dei 20 metri (14-14.350 MHz) con la cascata (waterfall) che mostra tutti i segnali ricevibili. Gli esperti possono riconoscere modulazioni SSB banda superiore (USB), CW (morse) ecc. È possibile digitare direttamente la frequenza centrale, selezionare la singola emittente con un clic sullo spettro, fare lo zoom di una parte dello spettro, ecc.

E questo è solo l'inizio di un'avventura. SDR è il punto di contatto di tre passioni: l'informatica, l'elettronica e la radio. Gli esperimenti

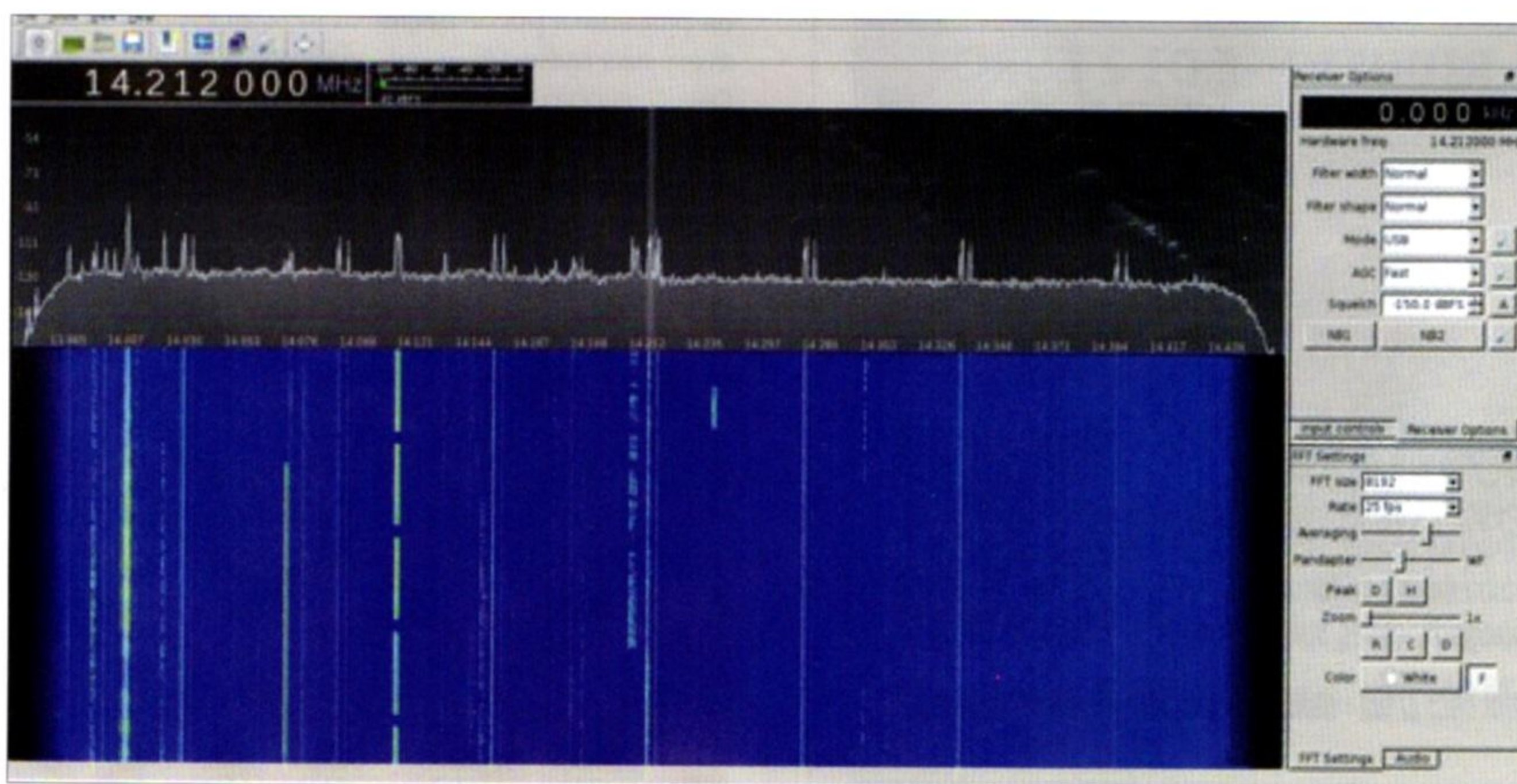


Fig. 5



Rovigo

Cen.Ser. - Viale Porta Adige 45

20 e 21 febbraio 2016

apertura dalle ore 9,00 alle ore 19,00

RIDUZIONE DI 2 EURO
presenta questo coupon alle casse
per ottenere il biglietto a prezzo ridotto

VINTAGE VILLAGE

mostra mercato dell'hobbista,
scambi tra appassionati di
radiantismo e cine-foto-ottica

**COSPLAY &
COMICS SHOW**

mostra mercato dei fumetti
videogiochi, games, Star Wars Tribute



info: 0425 27401 - www.arearebus.com - fiera@arearebus.com

di questo articolo vogliono solo mostrare le grandi possibilità di questi strumenti sia hardware (RedPitaya) sia software. Tanti temi possono essere affrontati e penso siano di vostro interesse: la struttura del programma in esecuzione nell'FPGA e l'architettura del sistema in esecuzione su Red Pitaya, la costruzione di radio tramite i blocchi di GNU radio e la scrittura di nuovi elementi per l'ambiente GNU radio stesso. Tutto questo, poi, diventa anche una scusa per capire come funziona una SDR, come strumenti matematici possano sostituire interi circuiti per la sintonizzazione, modulazione e demodulazione. E ancora si apre una finestra sul mondo della radio. Ci sono tantissimi segnali da ricevere. Nascosti nei picchi dello spettro ci sono testi, immagini, parole. Come interfacciare strumenti digitali per demodulare e decodificare queste informazioni è appassionante. Molte di queste cose sono ancora da sperimentare, e questo è sicuramente un punto di attrazione per ogni buon informatico, elettronico o radioamatore (o un pò tutte e tre le cose insieme). Attualmente nei nostri sistemi sperimentali Red Pitaya funziona come ricetrasmittitore capace di trasmettere e ricevere segnali fino ad una larghezza di 500 kHz, su quattro frequenze centrali indipendenti (due per la ricezione due per la trasmissione). I progetti aperti sono tanti. Per esempio l'interfacciamento di un finale di potenza filtrato e di un relé ad alta frequenza per rendere Red Pitaya una completa stazione radioamatoriale. Un altro progetto consiste nel realizzare un front-end per la radio che possa funzionare direttamente sul processore ARM del Red Pitaya stesso. Data la limitata potenza di calcolo occorrerà limitare la banda passante a poche decine di kHz, ma si potrebbe avere una radio accessibile via interfaccia web come l'oscilloscopio tipico del Red Pitaya. Se volete "giocare con la SDR" troverete interessante il lavoro svolto nel gruppo di creativi digitali chiamato RaspiBO a Bologna (www.raspibo.org).



per il MATERIALE

La board Red Pitaya (cod. 8007403RS) è in vendita al prezzo di Euro 249,00 ed è disponibile presso Futura Elettronica. Il prezzo si intende IVA compresa.

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

L'ACQUARIO CONNESSO

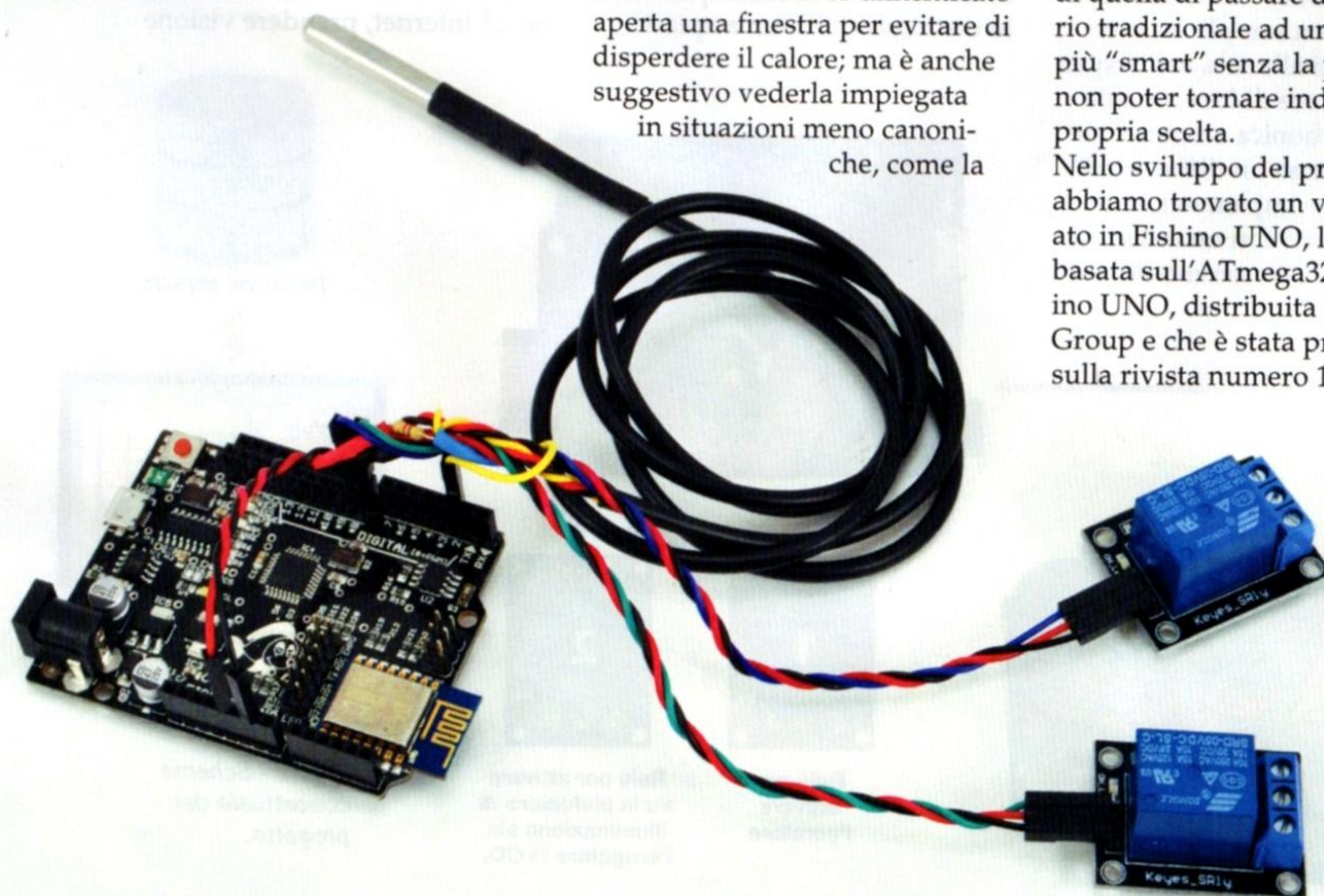
di SIMONE DALL'ASTA

Rendiamo smart un acquario esistente e inseriamolo nel mondo dell'IoT grazie alla scheda Fishino.

La tecnologia si è fatta strada ormai anche nella nostra vita quotidiana e la domotica diviene ogni giorno più familiare: è comodo pensare, ad esempio, ad un sistema di riscaldamento che ci avvisa di aver dimenticato aperta una finestra per evitare di disperdere il calore; ma è anche suggestivo vederla impiegata in situazioni meno canoniche, come la

gestione di un acquario proposta in queste pagine. Il progetto che vi presentiamo consente infatti di gestire un acquario senza però modificarne la struttura o l'aspetto elettrico. L'idea è quindi quella di passare da un acquario tradizionale ad una versione più "smart" senza la paura di non poter tornare indietro nella propria scelta.

Nello sviluppo del progetto abbiamo trovato un valido alleato in Fishino UNO, la scheda basata sull'ATmega328 di Arduino UNO, distribuita da Futura Group e che è stata presentata sulla rivista numero 198; la scel-



CARATTERISTICHE TECNICHE

- Alimentazione: 5 Vcc
- Basato su Fishino UNO (WiFi, RTC e lettore di microSD a bordo)
- Connessione con un database MySQL
- Acquisizione della temperatura dell'acqua tramite la sonda DS18B20
- Possibilità di monitorare la temperatura dell'acquario in un grafico ovunque ci si trovi tramite qualsiasi dispositivo con connessione a Internet
- Invio automatico di e-mail ad un indirizzo impostato in caso di temperature critiche
- Personalizzazione tramite microSD
- Attivazione o disattivazione automatica di aeratore, diffusore di anidride carbonica, plafoniere di illuminazione (se già presenti nell'acquario) tramite due relé

ta è stata dettata da specifiche funzionalità di cui la scheda dispone: una connessione WiFi per permettere di monitorare a distanza la temperatura dell'acqua utilizzando qualsiasi dispositivo dotato di connessione Internet e un orologio interno (RTC, Real Time Clock) con backup a batteria per temporizzare le fasi di funzionamento al posto dei soliti timer analogici (che in caso di blackout dovrebbero essere rimessi in passo). Infine, tramite una scheda microSD, è possibile definire con semplicità vari aspetti legati alla vita nell'acquario, come le ore di luce, quanta anidride carbonica erogare e se attivare o meno un diffusore di ossigeno. Per implementare le stesse funzioni con una classica Arduino UNO avremmo dovuto

dotarla di shield RTC, SD e WiFi, con evidente aumento dei costi e degli ingombri.

La scelta di Fishino UNO è poi naturale, in quanto il nome ...si addice al compito assegnatole!

IL SISTEMA

Vediamo dunque di cosa si tratta riferendoci alla Fig. 1; in essa vedete che il cuore pulsante del sistema è Fishino UNO, grazie al quale è possibile dotare l'acquario delle caratteristiche che lo rendono "smart" e che quindi lo distinguono da un acquario

tradizionale. Il funzionamento del dispositivo è semplice: Fishino UNO "apprende" gli orari di accensione e spegnimento delle plafoniere di illuminazione, dell'erogatore di anidride carbonica e dell'aeratore tramite i file contenuti nella scheda microSD. Dopo questa operazione sarà in grado di attivare o disattivare i dispositivi a seconda dell'orario. Il tempo è scandito dal modulo RTC (Real Time Clock) innestato in Fishino UNO e, anche in caso di alimentazione assente, la batteria di backup non fa perdere l'orario programmato. Grazie alla sonda impermeabile contenente un DS18B20 della Maxim è possibile rilevare la temperatura dell'acqua e, ogni mezz'ora, questo dato viene inviato in automatico tramite WiFi per essere poi memorizzato su un database MySQL (vedremo come). Successivamente sarà possibile, tramite un qualsiasi dispositivo dotato di connessione ad Internet, prendere visione

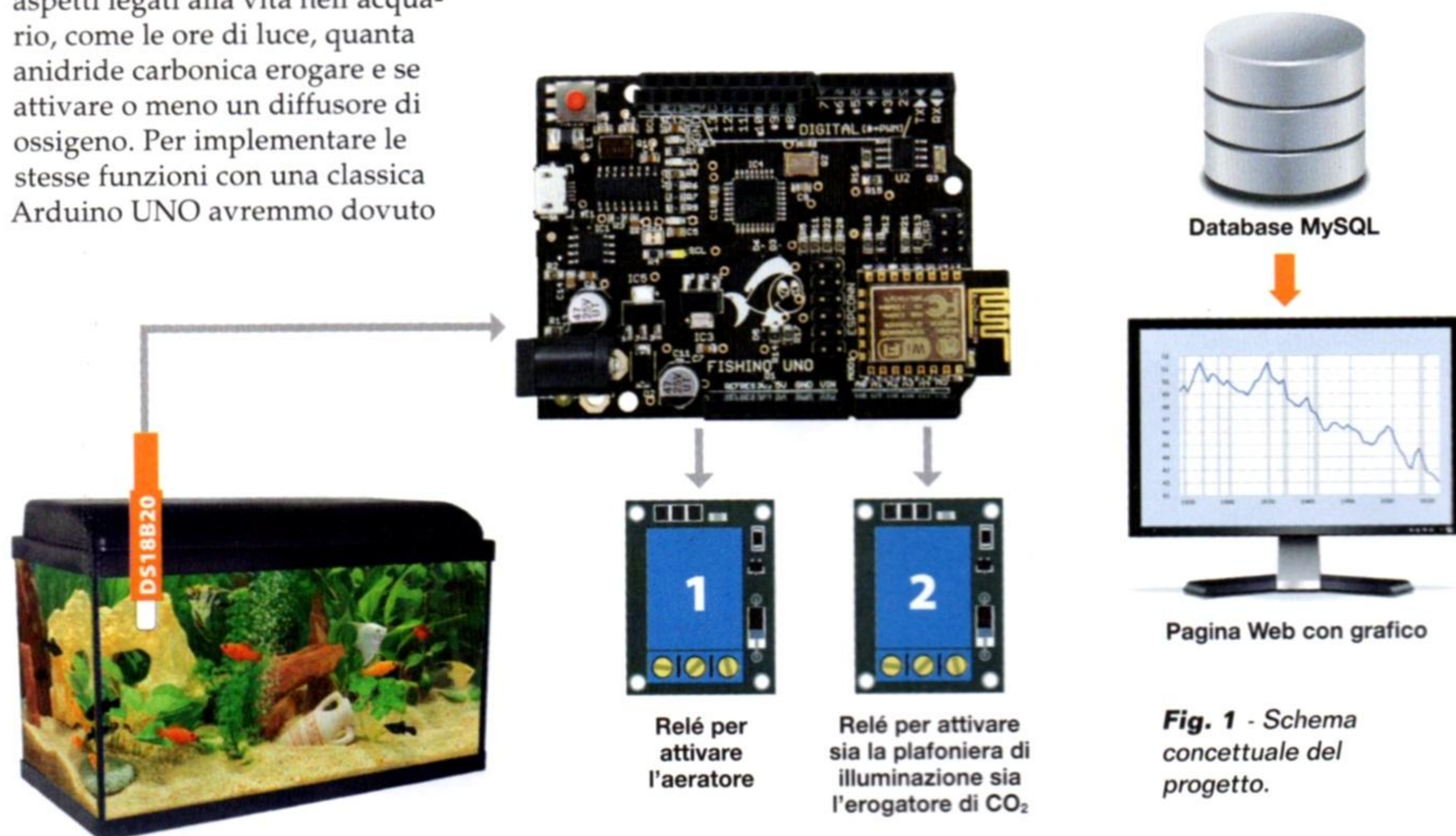


Fig. 1 - Schema concettuale del progetto.

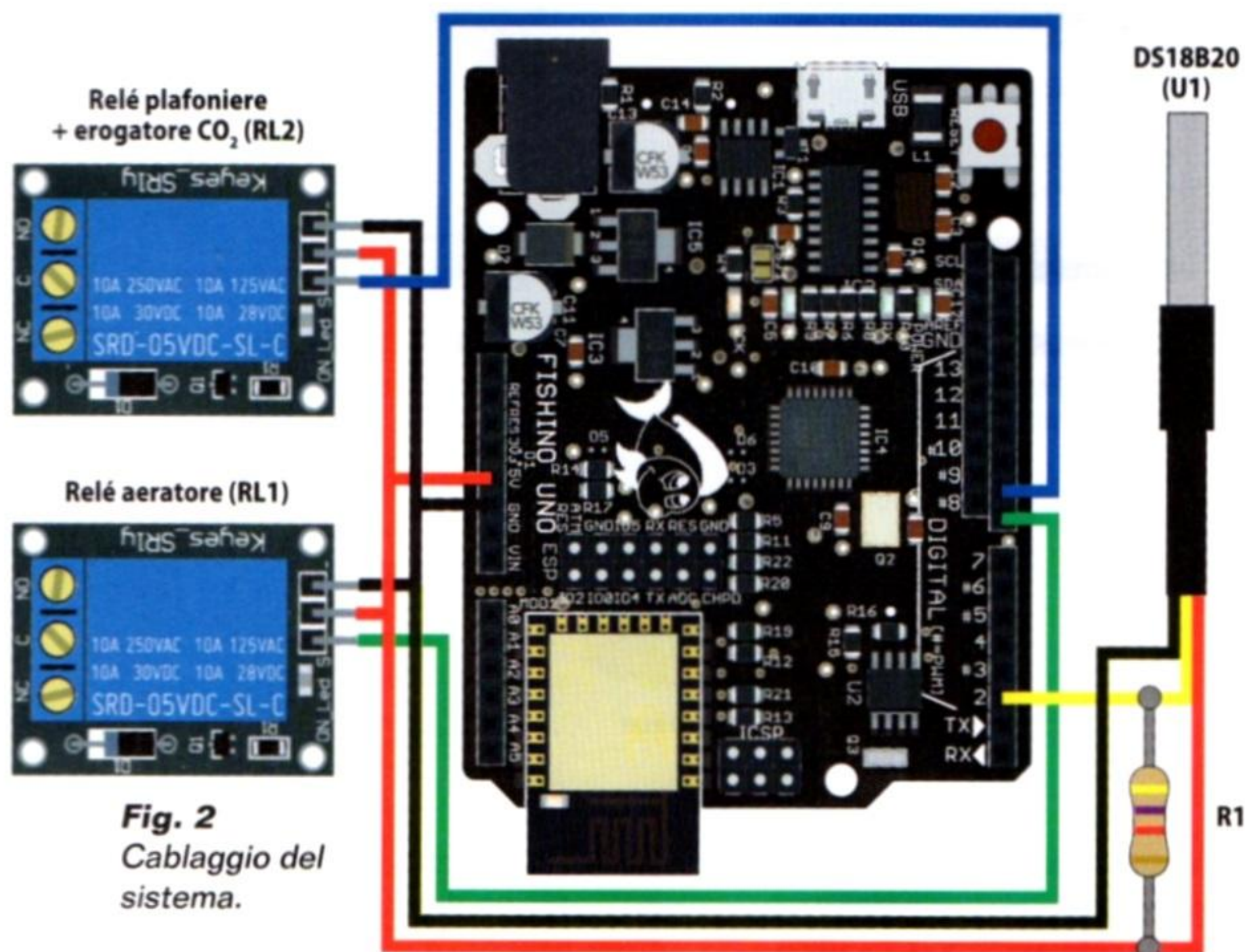


Fig. 2
Cablaggio del sistema.

della temperatura su un comodo grafico per sapere, ad esempio, la minima o la massima temperatura raggiunta dall'acqua. La temperatura rilevata, oltre ad esser salvata sul database, è usata anche per inviare, in caso di temperature critiche (troppo caldo o troppo freddo), una e-mail ad un indirizzo impostato così da essere avvisati e poter agire per tempo.

SCHEMA ELETTRICO

Nella rivista numero 198 abbiamo spiegato l'hardware di Fishino UNO, quindi qui ci limiteremo a spiegare la parte di circuito aggiuntiva richiesta dal progetto, il cui schema di cablaggio è mostrato dalla Fig. 2. Vediamo che la temperatura dell'acqua è acquisita da Fishino tramite una sonda di temperatura waterproof in acciaio inossidabile, realizzata con il sensore digitale di temperatura DS18B20 della Maxim (U1). U1 consente di misurare con precisione la temperatura in ambienti umidi, non solo se immerso in liquidi, con un range che va da -55°C a +125°C; naturalmente per un acquario questi

limiti sono più di quel che serve. Ogni sensore è dotato di un codice seriale a 64 bit memorizzato in una ROM al suo interno; ciò, unito al fatto che è richiesto un solo pin di comunicazione (1-Wire), permette di collegare più di un sensore a bus, in modo da misurare la temperatura in più punti dell'acquario ed ottenere così indicazioni più precise sulle condizioni dell'ambiente. La temperatura è misurata con una precisione di $\pm 0,5^\circ\text{C}$ (nel campo tra -10°C e $+85^\circ\text{C}$). La sonda si alimenta con una tensione che va da 3V a 5,5V.

La connessione della sonda a Fishino UNO si fa così:

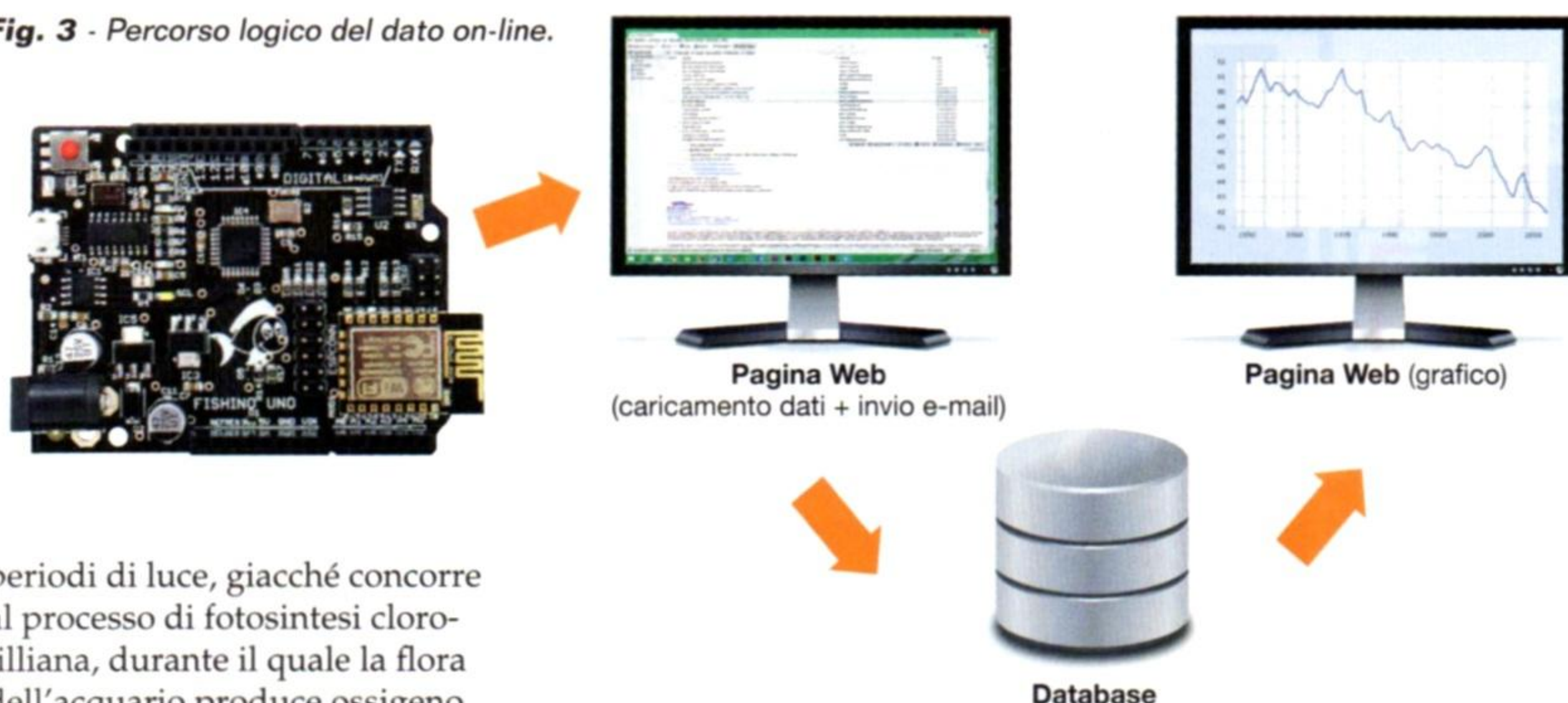
- il filo nero va al pin GND;
- il filo rosso va al pin 5V;
- il filo giallo (canale dati) va al pin digitale 2 e gli si deve collegare una resistenza (R1) da 4,7 k Ω verso il 5V (pull-up).

Per quanto riguarda RL1 e RL2, abbiamo utilizzato due schede dotate di relé monoscambio a 240V/10A, pilotabili con segnali TTL; ciascun modulo relé richiede un'alimentazione di 5 Vcc, prelevata da Fishino UNO (cui si connette anche la massa, con il filo nero), e rende accessibili i contatti N.A. (Normalmente Aperto - NO), COM (Comune) e N.C (Normalmente Chiuso). I moduli con RL1 e RL2 si pilotano con Fishino UNO tramite i pin digitali 8 e 9: il livello logico alto corrisponde a relé eccitato e lo zero a relé a riposo. Lo scambio dei relé va usato da interruttore, quindi C ed NO interromperanno un filo dell'alimentazione del carico per poi chiuderlo quando il relé viene attivato. Abbiamo previsto di collegare ad RL1 l'aeratore e di gestire con RL2 delle plafoniere di illuminazione e l'erogatore di CO₂. Per quanto riguarda l'apporto di anidride carbonica e di ossigeno siete liberi di usare il metodo con il quale siete abituati: potete, ad esempio, collegare al relé un'elettrovalvola normalmente chiusa che apre e chiude il tubo dentro il quale passa la CO₂ dopo aver attraversato il riduttore di pressione posto sopra la bombola. I carichi di plafoniere ed erogatore di anidride carbonica possono essere gestiti distintamente usando un relé aggiuntivo, ma, a nostro avviso, è più semplice usarne uno soltanto poiché, come ben saprete, l'apporto di CO₂ è necessario soprattutto durante i

Listato 1

```
CREATE TABLE acquario (
  Temperatura FLOAT NOT NULL,
  Timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP PRIMARY KEY
)
```


Fig. 3 - Percorso logico del dato on-line.



periodi di luce, giacché concorre al processo di fotosintesi clorofilliana, durante il quale la flora dell'acquario produce ossigeno assorbendo anidride carbonica grazie alla luce.

SOFTWARE

Visto l'hardware, passiamo alla parte software, che si divide sostanzialmente in due parti: la prima riguarda la programmazione di Fishino UNO, mentre la seconda è quella che permette il salvataggio dei dati su un

database relazionale MySQL, che ne permette poi il recupero per costruire un grafico sulla base della temperatura rilevata ogni mezz'ora. Potremmo vedere le due parti seguendo il percorso logico che seguirà il dato della temperatura dal momento dell'acquisizione a quello del

salvataggio fino ad arrivare all'output nel grafico. Conviene però cominciare dal fondo, ovvero da come avviene il salvataggio del dato su database, perché ciò vi aiuterà a capire il funzionamento dello sketch.

PAGINA WEB E DATABASE RELAZIONALE MYSQL

Per tenere monitorata la temperatura dell'acqua usando qualsiasi dispositivo dotato di connessione a Internet, è necessario rendere disponibile al mondo esterno ciò che Fishino UNO rileva; il modo più semplice per farlo è usare un sito web personale attraverso cui sarà possibile avere una sorta di finestra condivisa tra voi e Fishino. In Fig. 3 potete osservare il percorso effettuato dal dato della temperatura on-line: una prima pagina web si occupa della ricezione del dato e del salvataggio su database, mentre una seconda servirà per prelevare i dati dal database e mostrarli nel grafico.

Crea tabella

Nome tabella:
acquario

Struttura

Campo	Temperatura	Timestamp
Tipo	FLOAT	TIMESTAMP
Lunghezza/Set*1		
Predefinito ²	Nessuno	CURRENT_TIMESTAMP
Collation		
Attributi		
Null	☐	☐
Indice	---	PRIMARY
AUTO_INCREMENT	☐	☐
Commenti		

#	Campo	Tipo	Collation	Attributi	Null	Predefinito	Extra	Azione
1	Temperatura	float			No	Nessuno		Più ▼
2	Timestamp	timestamp			No	CURRENT_TIMESTAMP		Più ▼

Fig. 4
Campi della tabella "acquario".

Fig. 5 - Contenuto della tabella dopo la creazione.

Fig. 6 - Il contenuto della tabella "acquario" dopo aver inserito sei valori della temperatura.

← T →				Temperatura	Timestamp
☐	 Modifica	 Copia	 Elimina	27	2015-11-08 11:51:01
☐	 Modifica	 Copia	 Elimina	27.9	2015-11-08 11:51:05
☐	 Modifica	 Copia	 Elimina	27.91	2015-11-08 11:51:08
☐	 Modifica	 Copia	 Elimina	27.31	2015-11-08 11:51:13
☐	 Modifica	 Copia	 Elimina	26.31	2015-11-08 11:51:18
☐	 Modifica	 Copia	 Elimina	25.61	2015-11-08 11:51:25

Per prima cosa va quindi creato il sito web, se già non ne avete uno.

Per fare questo vi potete affidare ai tanti servizi di hosting disponibili a pagamento o gratuitamente; l'importante è che nel servizio sia inclusa la possibilità di disporre anche di un database MySQL. Una volta creato il sito web e attivata la base di dati, recatevi nella sezione nella quale sarà possibile gestire il vostro database (se ne avete più di uno sceglietene uno qualsiasi).

Una volta dentro, create una tabella (è comodo usare lo strumento di creazione sicuramente offerto) dal nome *acquario* con i seguenti campi (**Fig. 4**) rispettando maiuscole e minuscole:

- Temperatura
 - Tipo: FLOAT;
- Timestamp
 - Tipo: TIMESTAMP;
 - Predefinito: CURRENT_TIMESTAMP;
 - Indice: PRIMARY.

In questo modo avete creato una tabella con due parametri dove *Timestamp* (che, se non definito diversamente, è impostato automaticamente alla data e all'ora del server su cui è ospitato il database) è la chiave primaria, ovvero non potranno mai esserci due righe della tabella con lo stesso *Timestamp*. Se non volete usare lo strumento per la creazione delle tabelle, potete usare

la console (di solito la si trova sotto la voce *SQL*) e inserire il codice del **Listato 1**.

Poi cliccate su *Esegui* oppure *Salva*, a seconda dei casi. Il risultato sarà come quello mostrato in **Fig. 5**.

Per quanto riguarda la pagina web che verrà usata da Fishino per caricare la temperatura sul database, può essere utilizzato uno script PHP. La scelta è ricaduta sul PHP poiché, essendo un linguaggio di scripting server-side, si presta perfettamente al compito da svolgere: Fishino opera come client, si connette al server, invia la temperatura che viene elaborata e salvata su database. Per scrivere la pagina è sufficiente un editor di testo come il Blocco Note di Windows. Il file dovrà essere salvato, ad esempio come *in.php* (vi consigliamo di cambiare nome ma non sceglietene uno troppo lungo), e successivamente caricato sul vostro spazio web (preferibilmente nella directory radice) usando, ad esempio il pannello utente (o gestore file) del servizio di hosting. Vediamo insieme le parti di cui è composto lo script nel file *in.php*. Cominciate

aprendo il tag `<?php` (ricordatevi di chiuderlo con `?>` alla fine); da qui in poi seguirà il vostro script. Dovrete poi riportare (**Listato 2**) i parametri di connessione al vostro database (li trovate solitamente nel pannello di controllo): *\$username*, *\$password*, *\$db*, *\$tablename*, *\$host*. I nomi delle variabili (ricordiamo che nel linguaggio PHP si indicano con il simbolo *\$* davanti) possono essere anche cambiati, ricordandosi però di mantenere il nome scelto per tutto lo script. Alcuni servizi di hosting consigliano di lasciare vuoti i campi di username, password e host per due motivi: innanzitutto sono recuperati automaticamente, poi lo svantaggio del PHP è che spesso, in caso di errori di sintassi o di altro genere, viene stampato a schermo tutto il codice, compresi questi parametri personali. In ogni caso, se non fate errori, non c'è alcun rischio. La variabile *\$tablename* rappresenta il nome della tabella; noi la abbiamo chiamata "acquario" ma se avete scelto un nome differente allora dovrete sostituirlo con quello da voi impostato.

Per fare interpretare al server i dati che gli verranno passati da Fishino UNO, usiamo il metodo GET (avremmo potuto usare anche il metodo POST) con il quale si fanno richieste ad un Web Server. Il metodo GET consiste nell'accodare i dati che ci interessano all'indirizzo della pagina richiesta. Per farlo si deve aggiungere, dopo l'indirizzo della pagina, un punto in-



Fig. 7 - Esempio di e-mail inviata a seguito di una temperatura critica.

Listato 2

```
<?php
//-----
// ----- TODO -----
$username="username";
$password="password";
$db="database";
$tablename = "acquario";
$host="host";
//-----
```

Listato 3

```
$temperatura_acquario = $_GET['t'];
echo "Temperatura acquario = " . $temperatura_acquario;
```

Listato 4

```
if ($con = mysqli_connect($host, $username, $password))
{
    mysqli_select_db($con,$db) or die ("Unable to select database");
    $query = "INSERT INTO $tablename VALUES ($temperatura_acquario, now())";
    $result = mysqli_query($con,$query);
}
else
{
    echo('Unable to connect to database.');
```

terrogativo seguito dalle coppie *nome = valore* dei dati di interesse. Se vi è più di una coppia, esse vanno separate dal simbolo &. Il risultato sarà una variabile globale `$_GET` contenente l'array di tutte le coppie campo=valore che potremo usare nello script. Nel nostro caso avremo solo una coppia formata dalla temperatura con relativo valore. Vorremmo quindi passa-

re, ad esempio, la coppia *temperatura=25.6* (con il punto, non la virgola) al Web Server in modo tale da farglielo interpretare e, successivamente, caricare sul database. Per fare questo si dovrà assegnare alla variabile `$temperatura_acquario` il valore del campo *temperatura* passato con l'array `$_GET`. Se, ad esempio, chiamate il campo della temperatura con la

lettera *t*, allora si potrà selezionare questo specifico elemento dell'array rappresentato dalla variabile `$_GET` scrivendo `$temperatura_acquario = $_GET['t']` (ricordando gli apici prima e dopo *t*). In questo modo, digitando nella barra degli indirizzi del browser *www.miosito.it/in.php?t=25.6*, come risultato avrete che la variabile `$temperatura_acquario` sarà pari a 25,6. Se poi voleste visualizzare nel browser il dato appena inserito, basterà mostrarlo in output con la funzione *echo* (o *print*, o una funzione con lo stesso compito). È da notare che se vi dovessero essere errori di sintassi, di battitura o di altra natura nella query MySQL, questi non verrebbero notificati ma, semplicemente, non verrebbe caricato alcun dato sul database. In caso di errori vedreste semplicemente la temperatura in output ma, controllando nei dati contenuti nel database, vedreste una tabella vuota. Per verificare di non aver commesso errori, dopo aver fatto la prova appena descritta (inserire la temperatura di 25,6), controllate che il dato sia stato effettivamente salvato nel database. Se tutto è avvenuto senza errori allora la tabella non sarà vuota ma conterrà i dati appena inseri-

Listato 5

```
//Invio mail per temperature critiche
if ($temperatura_acquario < 23)
{
    $messaggio = "La temperatura dell'acquario e' troppo BASSA! Rilevata temperatura di " .
    $temperatura_acquario . "°C in data: " . date("Y-m-d H:i:s");
    mail('mail@mail.com','Temperatura acquario CRITICA',$messaggio);
}
if ($temperatura_acquario > 29)
{
    $messaggio = "La temperatura dell'acquario e' troppo ALTA! Rilevata temperatura di " .
    $temperatura_acquario . "°C in data: " . date("Y-m-d H:i:s");
    mail('mail@mail.com','Temperatura acquario CRITICA',$messaggio);
}
?>
```


ti. Vi consigliamo di provare ad inserire più di un dato, così da popolare la tabella come mostrato in Fig. 6.

Questo tornerà utile quando dovrete testare il corretto funzionamento dello script che mostrerà il grafico (ne parleremo più avanti). È giunto il momento di salvare i dati nel database (Listato 4).

A questo scopo, dovrete innanzitutto usare la funzione `mysqli` per far connettere la vostra applicazione Web al Database Manager MySQL e, successivamente, selezionare il database di interesse. Con `if ($con = mysqli_connect($host, $username, $password))` viene tentata la connessione e, in caso di fallimento, viene mostrato l'errore *Unable to connect to database*.

Se invece va a buon fine, viene tentato il collegamento al database con `mysqli_select_db($con, $db)`; anche qui è previsto un messaggio di errore in caso di fallimento.

La funzione `mysqli` accetta anche il nome del database come quarto parametro; potreste quindi usare un'unica funzione

Listato 6

```
<html>
<head>
<title>Acquario</title>
</head>
<body>
<h1 align="center">ACQUARIO </h1>
<p align="center">- based on Fishino UNO - </p>

<?php

    $max_value = 100; //Quanti valori mostra al massimo nel grafico

    //-----
    // ----- TODO -----
    $username="username";
    $password="password";
    $db="database";
    $tablename = "acquario";
    $host="host";
    //-----
```

Listato 7

```
$con = mysqli_connect($host, $username, $password);
mysqli_select_db($con, $db) or die('Errore connessione...');
```

per far connettere l'applicazione al Database Manager MySQL e selezionare un database: `mysqli_connect($host, $username, $password, $db)`.

La variabile `$con` è l'identificativo di connessione che servirà per comunicare con il database. Una volta avvenuta la connessione verrà mandata una query (una interrogazione di un data-

base) al database con la funzione `mysqli_query(identificativo, query)`, dove l'identificativo di connessione è `$con` (se mancante, verrà scelto l'ultimo identificativo usato) e la query è indicata come stringa in `$query`. Con questa query specifichiamo di inserire nella tabella `$tablename` (nel nostro caso, `acquario`) i valori di temperatura e orario nei campi

Listato 8

```
$q6 = "SELECT Temperatura, Timestamp FROM $tablename ORDER BY Timestamp DESC";

$i = 0;

$r6 = mysqli_query($con, $q6);
$n_rilevazioni = mysqli_num_rows($r6);
while((list($temp_water[$i], $temp_time[$i]) = mysqli_fetch_row($r6)) and $i<$max_value)
    $i = $i+1;

$i = $i - 1;

?>
```

Listato 9

```
<script type="text/javascript" src="https://www.google.com/jsapi"></script>
```



```

<script type="text/javascript">
  google.load('visualization', '1', {packages: ['corechart', 'line']});
  google.setOnLoadCallback(drawBackgroundColor);

  function drawBackgroundColor() {
    var data = new google.visualization.DataTable();
    data.addColumn('date', 'Time');
    data.addColumn('number', 'Temp');

    data.addRows([
      <?php
        $j = 0;
        while($i >= 0 and $temp_water[$i] != 0)
        {
          echo "[";
          echo "new Date(";
          $temp_date = date("Y,m-l,d,H,i,s", strtotime($temp_time[$i]));
          echo $temp_date;

          echo ")";
          echo ",";
          echo $temp_water[$i];
          echo "]";
          if ($i > 0)
          {
            echo ",";
          }
          $j = $j + 1;
          $i = $i - 1;
        }

      ?>

    ]);

    var options = {
      hAxis: {
        title: 't'
      },
      vAxis: {
        title: 'gradi'
      },
      focusTarget: 'category',
      curveType: 'function',
      lineWidth: 3,
      title: 'Temperatura acquario degli ultimi giorni',
      backgroundColor: 'white'
    };

    var chart = new google.visualization.LineChart(document.getElementById('chart_div'));
    chart.draw(data, options);
  }

</script>

<!-- Portions of this code are modifications based on work created and shared by Google and used according to
terms described in the Creative Commons 3.0 Attribution License. Code samples are licensed under the Apache
2.0 License. -->

```

del database. Con la funzione **now()** viene ritornato l'orario attuale del Web server nel formato **YYYY-MM-DD HH:MM:SS**. Siccome nella definizione della tabella del database avete impostato per il campo **Timestamp CURRENT_TIMESTAMP** come valore predefinito, anche passando **NULL** al posto di **now()**

non sorgerebbero problemi. Potreste aggiungere la possibilità di ricevere una e-mail in caso di temperature critiche (un esempio è in **Fig. 7**). Questo può essere fatto con la funzione **mail**, la cui sintassi è: **bool mail (string \$a , string \$oggetto , string \$messaggio [, string \$header_addizionali [, string \$parametri_addizionali**

]]). Tralasciando i parametri opzionali (tra parentesi quadre) è evidente come questa funzione permetta di inviare automaticamente il messaggio **\$messaggio** all'indirizzo **\$a** con oggetto della e-mail **\$oggetto**. La funzione può essere chiamata quando la temperatura è fuori da un certo range (nell'esempio, sotto i 23°C


```

<p align="right"><?php echo "Dati rilevati: #" . $n_rilevazioni; ?> </p>
<p>&nbsp;</p>

<?php
    echo "<br /><br /><strong>Ultima temperatura rilevata: " . $temp_water[0] . "&deg;C in data: " .
    $temp_time[0] . "</strong>";
?>

</body>
</html>

```

e sopra i 29°C).

GRAFICO DELLA TEMPERATURA

Per il grafico sul quale visualizzare le ultime temperature rilevate, la soluzione più semplice è sfruttare le molte librerie reperibili sul web; quella da noi utilizzata è la soluzione offerta da Google, che mette a disposizione gratuitamente un gran numero di grafici ottimizzati per i vari dispositivi, personalizzabili in molti aspetti e facili da utilizzare. I *Google charts*, la cui documentazione è reperibile all'indirizzo web <https://developers.google.com/chart/>, sono liberamente utilizzabili e forniti sotto licenza Creative Commons Attribution 3.0, mentre gli esempi di codice sono concessi sotto licenza Apache 2.0.

Il grafico che vi mostriamo nel nostro esempio è scelto dalla

categoria "Line Charts", con sfondo bianco, linea del grafico blu e con andamento curvilineo. Sull'asse delle ascisse rappresentiamo l'andamento del tempo, mentre sull'asse delle ordinate vi sarà la temperatura. Ogni singolo dato avrà poi un'etichetta, con data e temperatura in formato testuale, che comparirà al passaggio del cursore del mouse (o toccando lo schermo, per i dispositivi touch). Anche per visualizzare il grafico dovrete realizzare uno script PHP, chiamando il file *acquario_grafico.php*. Iniziate (**Listato 6**) con i tag di apertura di una normale pagina HTML (html, head, title, body) e aprite poi il tag PHP con *<?php*. Con la variabile *\$max_value* definite quanti dati mostrare nel grafico: sceglietene il valore considerando che verrà inviato e caricato un dato ogni

30 minuti quindi; per vedere il grafico di un giorno intero, il valore della variabile dovrà essere almeno pari a 48. Nel nostro esempio poniamo questo valore pari a 100, tanto da coprire poco più di 2 giorni.

Infine inserite i parametri di connessione al database come già avete fatto per il file *in.php*. Allo stesso modo facciamo collegare l'applicazione web al database: ciò si può fare come nel file *in.php* oppure in modo più abbreviato, come mostrato nel **Listato 7**.

Per prelevare le temperature dal database dovremo impostare la query *\$q6*; stavolta specifichiamo che vogliamo recuperare le temperature con relative date dalla tabella *acquario*, ordinate a partire dalla più recente. Dopo aver interrogato il database con la funzione *mysqli_query*

File	Dimensione	Data
 acquario_grafico.php	3 KB	08/11/2015
 in.php	1 KB	08/11/2015

Fig. 8 - Contenuto del "Gestore file" del sito web.

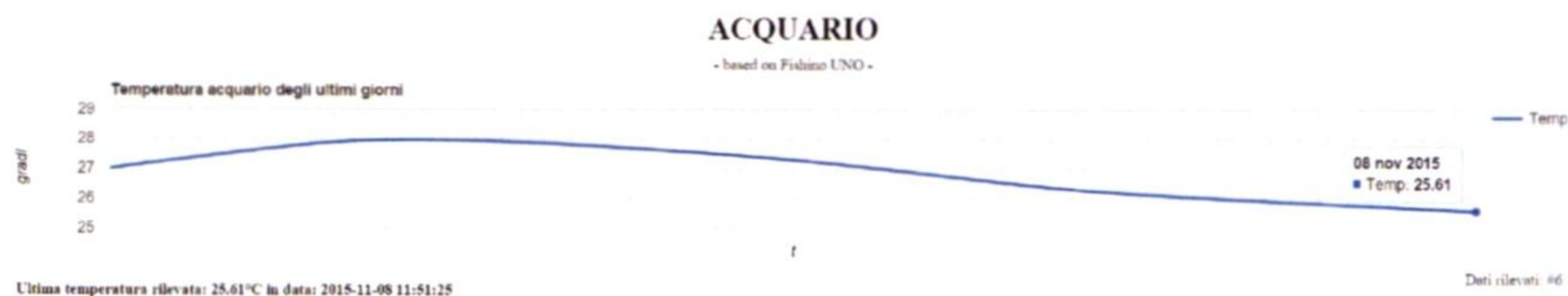


Fig. 9 - Aspetto del grafico con focus su uno dei dati.

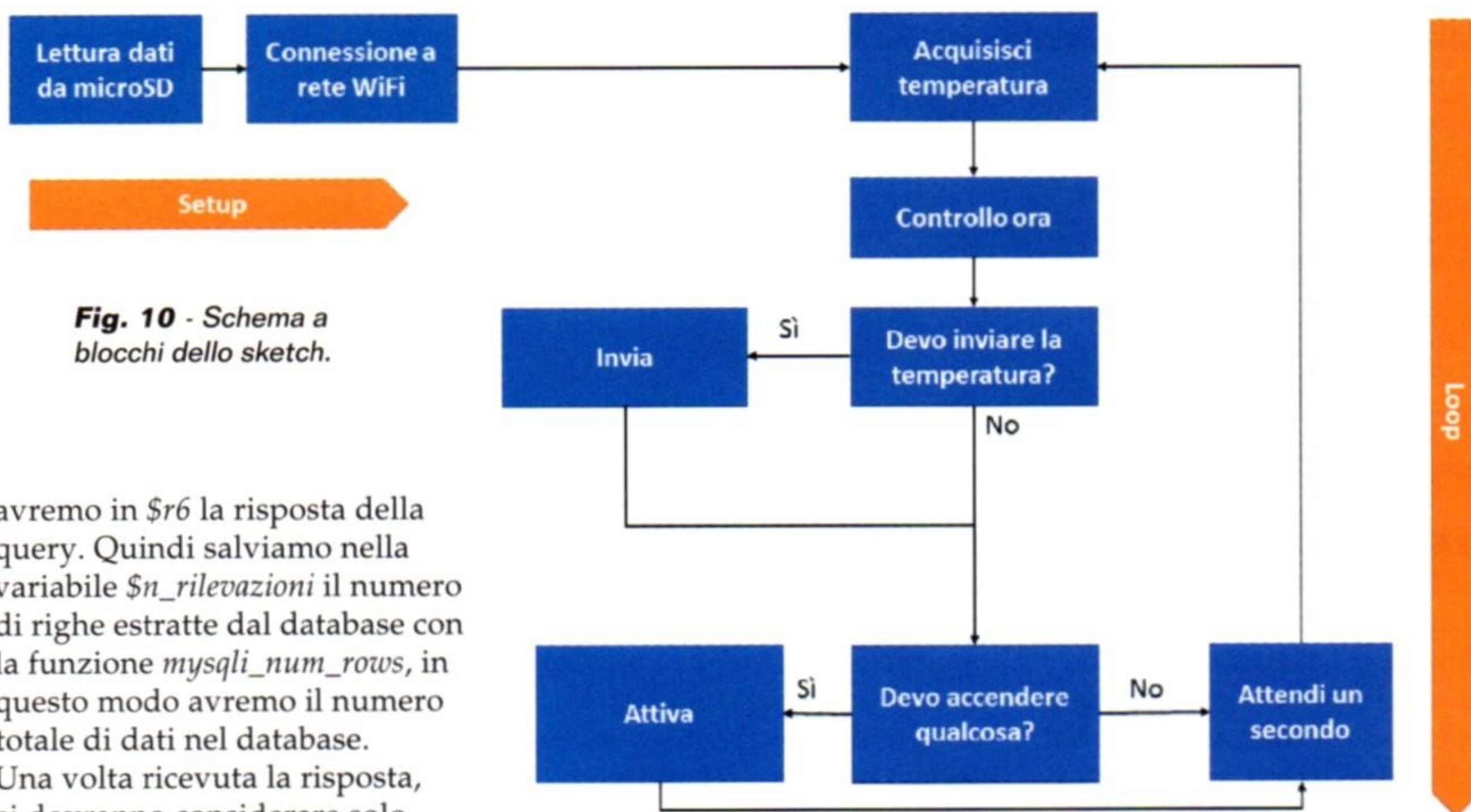


Fig. 10 - Schema a blocchi dello sketch.

avremo in `$r6` la risposta della query. Quindi salviamo nella variabile `$n_rilevazioni` il numero di righe estratte dal database con la funzione `mysql_num_rows`, in questo modo avremo il numero totale di dati nel database. Una volta ricevuta la risposta, si dovranno considerare solo le prime 100 righe di ciò che il database ha restituito. Per fare ciò, usate la funzione `mysql_fetch_row(risultato)` che ha come parametro il risultato dell'interrogazione del database (la variabile `$r6` nel nostro caso). Questa funzione carica in un array una riga del risultato associato all'identificativo specificato, oppure restituisce NULL se non ci sono più righe.

Ciascun elemento di questo array sarà formato dalla coppia temperatura-data. Usate il costrutto `list()` per assegnare il valore dei due campi a due variabili rappresentate ciascuna da una cella di un array (`$temp_water[$i]` e `$temp_time[$i]`, dove `$i` è una variabile inizializzata a 0 e incrementata di 1 ad ogni ci-

clo). Questo costrutto è usato per assegnare valori a una lista di variabili in una sola operazione. Con il ciclo `while` nel **Listato 8** viene specificato che, per ciascuna riga restituita dal database, si deve salvare la temperatura e la data nei due array indicati rappresentanti rispettivamente il vettore delle temperature e

quello delle date. Se il numero di righe restituite è maggiore del massimo degli elementi voluti, il ciclo termina quando si sono esauriti gli elementi; se invece il numero di righe è maggiore, allora il ciclo termina quando la variabile contatore `$i` raggiunge il valore massimo. Una volta fuori dal ciclo `while`,

Listato 12

```

File myFile;

//Pin DATA di DS18B20: pin 2
OneWire ds(2);

// Initialize the Fishino client library
FishinoClient client;

//Credenziali rete WiFi
#define MY_SSID "your_network"
#define MY_PASS "your_password"

/* Definisco i pin di attivazione dei relé.
 * BOLLE (Aeratore) = Pin 8
 * PLAFO (Plafoniera + CO2) = Pin 9
 */
#define BOLLE 8
#define PLAFO 9

//Numero massimo di fasce orarie nel file contenuto nella microSD
#define MAX_VAL 4

RTC_DS1307 RTC;

//Matrice usata per memorizzare gli orari letti da microSD
int orari_bolle_plaf[MAX_VAL][7];
  
```



plaf - Blocco note

File Modifica Formato

```

1300002200001
1300001330002
1930002330002
  
```

Fig. 11 - Esempio di file `plaf.txt`.

Listato 13

```
void setup() {
  Serial.begin(115200);

  //Inizializzo gli elementi della matrice a 0
  for (int t = 0; t < MAX_VAL; t++)
    for (int j = 0; j < 7; j++)
      orari_bolle_plaf[t][j] = 0;

  //Leggo il file nella microSD e inserisco gli orari nella matrice
  leggi_SD();

  //Relé non eccitati
  pinMode(PLAFO, OUTPUT);
  pinMode(BOLLE, OUTPUT);
  digitalWrite(PLAFO, LOW);
  digitalWrite(BOLLE, LOW);

  // only for Leonardo needed
  // necessario solo per la Leonardo
  while (!Serial)
    ;
  // initialize SPI
  // inizializza il modulo SPI
  SPI.begin();
  SPI.setClockDivider(SPI_CLOCK_DIV2);

  // reset and test WiFi module
  // resetta e testa il modulo WiFi
  while (!Fishino.reset())
    Serial << F("Fishino RESET FAILED, RETRYING....\n");
  Serial << F("Fishino WiFi RESET OK\n");

  // go into station mode
  // imposta la modalità stazione
  Fishino.setMode(STATION_MODE);

  // Initialize the Wifi connection.
  // Inizializza la connessione WiFi
  Serial << F("Setting up the Wifi connection...\n");

  // try forever to connect to AP
  // tenta la connessione finché non riesce
  while (!Fishino.begin(MY_SSID, MY_PASS) == STATION_GOT_IP)
    Serial << F("ERROR CONNECTING TO AP, RETRYING....\n");

  Serial << F("SUCCEFULLY CONNECTED TO AP\n");

#ifdef IPADDR
  Fishino.config(ip);
#else
  Fishino.staStartDHCP();
#endif
  delay(1000);
}
```

sarà necessario decrementare di 1 la variabile *\$i* in modo da avere l'indice dell'array *\$temp_water* riferito all'ultimo valore inserito (se le righe restituite sono più di 100, ovviamente il valore di *\$i* sarà 100). In questo modo avete due array: uno contenente tutte le temperature da rappresentare nel grafico, l'altro con le date relative all'acquisizione. È il momento di inserire questi valori in un grafico; per sfruttare quello

offerto da Google aggiungete le librerie necessarie (**Listato 9**). Successivamente basta impostare qualche parametro legato all'aspetto del grafico e fornire i dati da rappresentare. L'aspetto più interessante è il passaggio dei dati, che avviene sotto forma di tabella dove le colonne costituiscono i campi da rappresentare nell'ordine ascisse e ordinate, mentre le righe sono i dati veri e propri.

Nel **Listato 10** vediamo infatti la variabile *data*, rappresentante i dati, che permette di aggiungere colonne con il metodo *addColumn(tipo, nome)* e righe con il metodo *addRows*. Per evitare di dover aggiungere manualmente una riga per ogni elemento si può sfruttare un ciclo che aggiunga iterativamente righe separate da virgola fino a che non arriva al penultimo elemento, dopo il quale la virgola non è più necessaria. Alla fine del ciclo si avrà un risultato del tipo: *data.addRows([[data, temperatura], [data, temperatura], [data, temperatura]])*. Associando il grafico al contenitore *chart_div*, sarà possibile mostrarlo in output:

```
<div id="chart_div"></div>
```

Prima di vedere il risultato finale vi mostriamo come sia possibile visualizzare in output il numero totale di dati nel database e l'ultima temperatura rilevata con il relativo orario (**Listato 11**). Una volta salvato il file (ricordatevi di chiudere tutti i tag aperti), caricatelo sul vostro sito web tramite il gestore file (**Fig. 8**). Il grafico sarà poi visibile (**Fig. 9**) digitando ad esempio www.mio-sito.it/acquario_temperatura.php.

LO SKETCH

Passiamo ora a schematizzare in pochi passi la logica dello sketch e lo facciamo con l'aiuto della **Fig. 10**. Inizialmente si includono le necessarie librerie:

```
#include <Wire.h>
#include <RTCLib.h>
#include <Flash.h>
#include <FishinoUdp.h>
#include <FishinoSockBuf.h>
#include <Fishino.h>
#include <SPI.h>
#include <SD.h>
```


Listato 14

```

/* Funzione che legge il file plaf.txt nella microSD
 * e ne estrae gli orari inserendoli nella matrice
 * degli orari.
 * CS - pin 4*/
void leggi_SD()
{
    String orari = "";
    if (!SD.begin(4)) {
        return;
    }
    int i;
    char c;
    myFile = SD.open("plaf.txt");
    if (myFile) {
        i = 0; //Numero di righe trovate nel file
        while (myFile.available()) {
            c = myFile.read();
            orari += c;
            if (c == '\n') //C'è ancora una riga
                ++i;
        }
        myFile.close();
    }
    int x = 0;
    int cont = 0;
    for (int j = 0; j <= i; j++)
    {
        while (orari[cont] != '\n')
        {
            switch (x)
            {
                case 0:
                    orari_bolle_plaf[j][0] += (orari[cont] - 48) * 10; //ORE_ON
                    break;
                case 1:
                    orari_bolle_plaf[j][0] += (orari[cont] - 48);
                    break;
                case 2:
                    orari_bolle_plaf[j][1] += (orari[cont] - 48) * 10; //MINUTI_ON
                    break;
                case 3:
                    orari_bolle_plaf[j][1] += (orari[cont] - 48);
                    break;
                case 4:
                    orari_bolle_plaf[j][2] += (orari[cont] - 48) * 10; //SECONDI_ON
                    break;
                case 5:
                    orari_bolle_plaf[j][2] += (orari[cont] - 48);
                    break;
                case 6:
                    orari_bolle_plaf[j][3] += (orari[cont] - 48) * 10; //ORE_OFF
                    break;
                case 7:
                    orari_bolle_plaf[j][3] += (orari[cont] - 48);
                    break;
                case 8:
                    orari_bolle_plaf[j][4] += (orari[cont] - 48) * 10; //MINUTI_OFF
                    break;
                case 9:
                    orari_bolle_plaf[j][4] += (orari[cont] - 48);
                    break;
                case 10:
                    orari_bolle_plaf[j][5] += (orari[cont] - 48) * 10; //SECONDI_OFF
                    break;
                case 11:
                    orari_bolle_plaf[j][5] += (orari[cont] - 48);
                    break;
                case 12:
                    orari_bolle_plaf[j][6] += (orari[cont] - 48); //1 (plafoniere) o 2 (bolle)
                    break;
            }
            ++x;
            ++cont;
        }
        ++cont;
        x = 0;
    }
    orari = "";
}

```

```

#include <OneWire.h>
#include <avr/io.h>
#include <avr/wdt.h>

```

Molte di esse saranno già presenti se avete scaricato l'IDE di Arduino mentre altre, come quelle di Fishino (scaricabili dal sito www.fishino.it o www.fishino.com), RTCLib.h per usare il modulo Real Time Clock (scaricabile da <https://github.com/adafruit/RTCLib>) e OneWire (scaricabile ad esempio da <http://playground.arduino.cc/Learning/OneWire>) dovranno essere aggiunte nella directory *Libraries* di Arduino.

Successivamente andranno definite alcune variabili (**Listato 12**): va dichiarato, ad esempio, il pin al quale è collegata la sonda DS18B20 (nel nostro caso il 2), va inizializzata la libreria client di Fishino e va definito il nome della rete WiFi con relativa password (*MY_SSID* e *MY_PASS*). Si dovranno poi definire i pin con i quali si attiveranno e disattiveranno i due relé. Con *MAX_VAL* potete specificare quante fasce orarie massime volete considerare, non è consigliabile inserire un valore alto per poi inserire pochi orari. Ad esempio se vorrete accendere una sola volta la plafoniera di illuminazione e, magari, due volte l'aeratore, allora potreste mettere come *MAX_VAL* un valore poco più grande di 3 (ad esempio 4 o 5) così da non sprecare memoria e da poter aggiungere in futuro qualche fascia oraria senza ricompilare lo sketch ma semplicemente modificando il file sulla microSD.

Infine, sarà usata una matrice *orari_bolle_plaf[MAX_VAL][7]* per poter salvare gli orari contenuti nella microSD in modo da non doverla aprire ad ogni

Listato 15

```
// Funzione per calcolare la temperatura dell'acqua in gradi Celsius tramite il DS18B20
float getTemp() {
    byte data[12];
    byte addr[8];

    if ( !ds.search(addr)) {
        //no more sensors on chain, reset search
        ds.reset_search();
        return -1000;
    }

    if ( OneWire::crc8( addr, 7) != addr[7]) {
        return -1000;
    }

    if ( addr[0] != 0x10 && addr[0] != 0x28) {
        return -1000;
    }

    ds.reset();
    ds.select(addr);
    ds.write(0x44, 1); // start conversion, with parasite power on at the end

    byte present = ds.reset();
    ds.select(addr);
    ds.write(0xBE); // Read Scratchpad

    for (int i = 0; i < 9; i++) { // we need 9 bytes
        data[i] = ds.read();
    }

    ds.reset_search();

    byte MSB = data[1];
    byte LSB = data[0];

    float tempRead = ((MSB << 8) | LSB); //using two's compliment
    float TemperatureSum = tempRead / 16;

    return TemperatureSum;
}
```

ciclo, ma solo una volta sola. Questa matrice avrà tante righe quante sono le fasce orarie (le restanti saranno poste a zero) e le colonne indicheranno rispettivamente:

1. ora di accensione;
2. minuto di accensione;
3. secondo di accensione;
4. ora di spegnimento;
5. minuto di spegnimento;
6. secondo di spegnimento;
7. carattere per distinguere tra illuminazione ed erogatore di anidride carbonica o aeratore.

SETUP()

Nel **Listato 13** vediamo l'inizializzazione di Fishino nel *Setup*: si può notare come Fishino sia in modalità stazione. La matrice degli orari è inizialmente riempita di zeri ma, tramite la funzione **leggi_SD()**, viene progressivamente popolata dagli orari definiti da microSD.

LEGGI_SD()

La microSD è letta tramite la funzione **leggi_SD()** che non necessita di alcun parametro e si limita ad aprire il file *plaf.txt* (**Fig. 11**) contenuto nella microSD. All'interno di questo file saranno contenuti gli orari di accensione e spegnimento dei due relé nel formato *HHMMSSHHMMSSn* dove *n* è un carattere 1 o 2 a seconda che si tratti di un orario riferito al relé dell'illuminazione e dell'anidride carbonica oppure di quello dell'aeratore. Se, ad esempio, volessimo accendere l'aeratore tra le 10 e le 11 del mattino, dovremmo scrivere: *1000001100001*. Aggiungete una nuova riga di testo se e solo se volete aggiungere un nuovo orario, non lasciate altrimenti righe vuote. Una volta letti tutti i caratteri del file *plaf.txt* e averli inseriti nella stringa *orari*, questa viene analizzata e, in base alla posizio-

Listato 16

```
/* Funzione per decidere se il tempo attuale sia in un intervallo o meno.
 * Accetta 6 parametri:
 * - Prima ora
 * - Primo minuto
 * - Seconda ora
 * - Secondo minuto
 * - Ora attuale
 * - Minuto attuale
 * Restituisce true o false se il calcolo ha esito affermativo o meno*/
boolean ora_giusta(int oral, int minutol, int ora2, int minuto2, int ora_ adesso, int minuto_ adesso)
{
    //Considero subito una risposta affermativa
    boolean risposta = true;

    //Vado a cercare se sono fuori dall'intervallo
    if (ora_ adesso < oral || ora_ adesso > ora2)
        risposta = false;
    if (oral == ora2 && minutol == minuto2)
        risposta = false;
    if (ora_ adesso == oral && minuto_ adesso < minutol)
        risposta = false;
    if (ora_ adesso == ora2 && minuto_ adesso >= minuto2)
        risposta = false;
    if (oral > ora2 || (oral == ora2 && minutol > minuto2))
        risposta = false;

    return risposta;
}
```



```

void loop() {
  //Imposto il WatchDog a 8 secondi
  wdt_enable(WDTO_8S);
  // Server address:
  char server[] = "www.miosito.it";
  float temperatura = 0.0;
  while (temperatura < 1 || temperatura > 84) //Evita di stampare dati impossibili
    temperatura = getTemp();
  //Inizializza stringa da inviare al server
  String temperature = "";
  //Data attuale
  DateTime now = RTC.now();
  //Queste variabili non sono necessarie, sono usate solo per comodità
  int ore = now.hour();
  int minuti = now.minute();
  int secondi = now.second();
  int anno = now.year();
  //Ad ogni inizio ora e ogni mezza...
  if ((minuti == 0 || minuti == 30) && (secondi == 0 || secondi == 1))
  {
    // close any connection before send a new request.
    // This will free the socket on the WiFi module
    // chiude ogni ventuale connessione prima di inviare una nuova richiesta
    // quest libera il socket sull modulo WiFi
    client.stop();
    delay(1000);
    //Preparo la stringa da inviare al server
    temperature = "GET /in.php?t=";
    temperature += temperatura;
    temperature += " HTTP/1.1";
    // if you get a connection, report back via serial:
    // se ottiene una connessione, stampa i dati sulla seriale
    if (client.connect(server, 80))
    {
      //Serial.println("connected to server");
      // Make a HTTP request:
      client.println(temperature);
      //TODO
      client.println("Host: www.miosito.it");
      client.println("User-Agent: ArduinoWiFi/1.1");
      client.println("Connection: close");
      client.println();
    }
  }
  /* Variabili per decidere se accendere o spegnere i relé:
  * false = spento
  * true = acceso
  * Inizialmente li considero spenti*/
  boolean bolle_on = false, plaf_on = false;
  /*Scorro la matrice degli orari per controllare se deve essere acceso qualcosa
  *(nel caso si trovi qualcosa da accendere pongo a true le variabili bolle_on o plaf_on)*/
  for (int k = 0; k < MAX_VAL; k++)
  {
    if (orari_bolle_plaf[k][0] + orari_bolle_plaf[k][1] + orari_bolle_plaf[k][3] + orari_bolle_plaf[k][4] != 0)
    {
      if (ora_justa(orari_bolle_plaf[k][0], orari_bolle_plaf[k][1], orari_bolle_plaf[k][3], orari_bolle_plaf[k][4], ore, minuti) == true)
      {
        switch (orari_bolle_plaf[k][6])
        {
          case 1:
            plaf_on = true;
            break;
          case 2:
            bolle_on = true;
            break;
        }
      }
    }
  }
  //Accesi (true) o spenti (false)?
  if (plaf_on == false)
    digitalWrite(PLAFO, LOW);
  else
    digitalWrite(PLAFO, HIGH);
  if (bolle_on == false)
    digitalWrite(BOLLE, LOW);
  else
    digitalWrite(BOLLE, HIGH);
  //Resetto il Fishino una volta al giorno
  if (ore == 23 && minuti == 59 && (secondi >= 0 || secondi <= 5))
    delay(15000); //Lascio scadere gli 8 secondi del WatchDog
  //Attendo un secondo
  delay(1000);
  //Resetto il contatore del WatchDog
  wdt_reset();
}

```


ne del carattere in esame, viene inserito in una cella della matrice il valore calcolato differenziando tra decine e unità (**Listato 14**). Ad esempio la stringa "34" viene trasformata nel numero 34 moltiplicando il 3 per 10, aggiungendolo al 4, il tutto dopo aver convertito da *char* a *intero*.

GETTEMP()

La funzione **getTemp()**, nel **Listato 15**, non è stata modificata dalla funzione che si trova anche nella libreria OneWire. Il suo scopo è quello di calcolare la temperatura in base a quanto misurato dal sensore DS18B20. La funzione restituisce un float rappresentante la temperatura in gradi Celsius.

ORA_GIUSTA

La funzione **ora_giusta** (**Listato 16**) accetta come parametri tre orari formati da ore e minuti e controlla se il terzo orario si trova tra i primi due.

La funzione non controlla se i due orari sono invertiti ma si potrebbe aggiungere tale opzione. In ogni caso è sufficiente rispettare l'ordine con il quale li si inserisce.

Inizialmente, nella funzione, la variabile *risposta* è posta a true. Successivamente si verifica se l'orario in esame si trovi fuori dall'intervallo analizzato, in caso affermativo la variabile *risposta* è posta a false.

Alla fine dei confronti viene restituito il valore di *risposta*. Se tale valore è rimasto a true allora significa che l'orario analizzato si trova tra gli altri due.

LOOP()

Passiamo ora ad analizzare il corpo principale dello sketch, la funzione **Loop()** (**Listato 17**). All'inizio del ciclo viene impostato il WatchDog a 8 secondi così

da resettare Fishino in caso di blocchi o cicli infiniti causati da errori imprevisti.

Viene poi definito il server al quale connettersi, basta indicare in *server[]* l'indirizzo del vostro sito Web sul quale avete caricato il file *in.php*.

In seguito viene letto il valore della temperatura dalla sonda, evitando errori di misura con un ciclo while.

Ogni 30 minuti deve avvenire la connessione al server per inviare la temperatura, quindi si chiudono le eventuali connessioni ancora aperte e si prepara la stringa da inviare.

La stringa sarà composta dal GET, che abbiamo già trattato, e dal percorso del file *in.php* sul sito Web (se lo avete messo nella directory radice sarà sufficiente inserire */in.php*) seguito dal campo *t* (quello che verrà usato dallo script PHP per recuperare il valore della temperatura) con relativo valore. Infine viene effettuata la connessione al Web server e inviata la temperatura con il metodo GET.

Ricordatevi di cambiare anche l'indirizzo host in *client.println("Host: www.miosito.it")* inserendo, ancora una volta, l'indirizzo del server.

Successivamente si scorre la matrice degli orari per decidere se i relé devono essere accesi o spenti e, per far questo, si usano due variabili *bolle_on* e *plaf_on* inizialmente poste a false che, se la funzione *ora_giusta* ritorna true, allora vengono poste a true.

Alla fine del ciclo vengono eccitati solo i relé la cui variabile booleana relativa è posta a true, altrimenti vengono disattivati.

Per sicurezza, una volta al giorno (alle 23:59) Fishino è resettato, così da azzerare tutti i contatori e non incorrere in blocchi inutili. Il reset è ottenuto facendo attende-

re 15 secondi così da far scadere il WatchDog.

Questa parte può ovviamente essere evitata.

Viene poi resettato il contatore del WatchDog dopo aver atteso un secondo. Infine viene rieseguito il ciclo di *loop()*.

Se non avete mai impostato l'orario dell'RTC di Fishino potete farlo facilmente caricando lo sketch *Softrtc* che trovate negli esempi della libreria *RTCLib*.

SVILUPPI FUTURI

Il sistema presentato svolge solo alcune delle funzioni che potrebbe compiere nel settore dell'acquariofilia; con qualche sensore in più e alcune modifiche al software, potreste ad esempio utilizzare il valore della temperatura per far gestire a Fishino l'aspetto del raffreddamento e del riscaldamento. Sulla base del file *plaf.txt*, potreste inserire un ulteriore file nella microSD e usarlo per definire il colore di strisce LED RGB per dare un tono allegro all'ambiente. ■



per il MATERIALE

Tutto il materiale necessario per realizzare questo progetto è disponibile presso Futura Elettronica. La board Fishino (cod. FISHINOONO) viene fornita montata e collaudata al prezzo di Euro 36,00. La sonda di temperatura waterproof con DS18B20 (cod. SONDADS18B20) è in vendita a Euro 7,00. Il modulo 1 relé 5 Vdc 10A viene fornito montato al prezzo di Euro 5,00. 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
<http://www.futurashop.it>



COLIBRI': DRIVER DI POTENZA PER RGBW

Permette di pilotare i nuovissimi diodi luminosi a 4 colori, variandone agevolmente la luminosità e la colorazione.

di MASSIMO DEL FEDELE

Di recente sono apparsi sul mercato alcuni LED multicolore di potenza che, a differenza dei più comuni RGB, incorporano in più un LED bianco: vengono chiamati RGBW (Red, Green, Blu, White) proprio per questo. Sono stati creati perché sebbene per ottenere una luce bianca sia sufficiente accendere insieme LED rossi, verdi e blu (la proporzione tra la luminosità di rosso, verde e blu determina la tonalità del bianco, che può essere calda o fredda) l'aggiunta di un LED bianco permette di ottenere dei bianchi molto più "puliti" rispetto a quanto si può ottenere dalla miscelazione dei tre colori base; è inoltre possibile utilizzare tutti e quattro i insieme per ottenere una potenza luminosa maggiore.

I LED di tipo RGBW sono reperibili tipicamente in potenze come 4 x 1 watt (4 watt totali) e 4 x 3 watt (12 watt totali). C'è da dire che le potenze reali sono effettivamente inferiori rispetto a quanto dichiarato dai produttori, ed è sempre meglio guardare la massima corrente di pilotaggio piuttosto che la potenza, se non altro perché i LED

si comandano in corrente e perché i costruttori forniscono i grafici della luminosità in funzione della corrente diretta (I_f).

Vista la disponibilità di questi nuovi LED a quattro colori, abbiamo progettato un driver di potenza per alimentarli a partire da segnali PWM di controllo che possono essere forniti, ad esempio, da un microcontrollore, ma anche da una scheda Arduino. In queste pagine descriveremo il progetto, proponendo altresì uno sketch per la gestione da Fishino Uno, che è la nostra scheda Arduino Uno compatibile.

Prima di analizzare lo schema elettrico del circuito, torniamo un attimo sulle considerazioni fatte pocanzi circa la potenza e la corrente, per spiegare come mai ai fini della valutazione di un LED RGBW si debba prendere in considerazione la corrente diretta. Prendiamo un LED quadricolore da 4 watt, potenza distribuita tra quattro LED che usualmente hanno una corrente di pilotaggio di 300 mA (corrente diretta); siccome la caduta di tensione sui LED varia in base al colore, abbiamo

CARATTERISTICHE TECNICHE

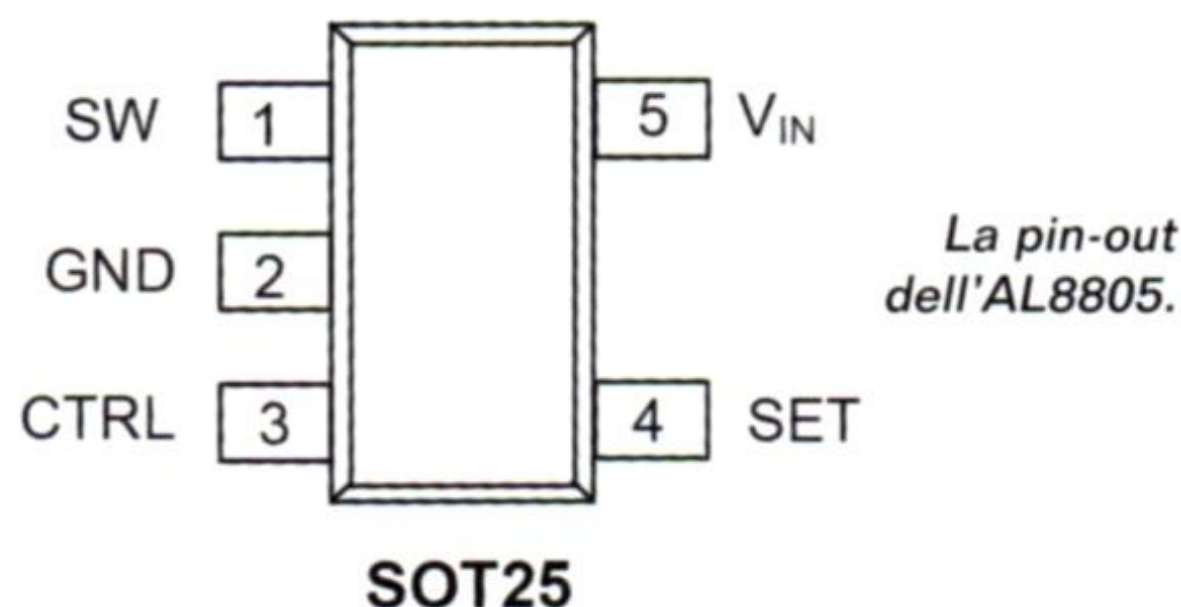
- Tensione di alimentazione: 6÷36 Vcc
- Funzionamento switching
- Corrente massima in uscita: 1 A per canale
- Controllo di luminosità (dimming): sia analogico che digitale
- Numero di canali per modulo: 4
- Dimensioni: 27,6 x 37,3 mm

potenze differenti per i vari LED, ossia:

$$P_{\text{rosso}} = (2,2 + 2,5V) \cdot 300 \text{ mA} = 0,66 + 0,75 \text{ W}$$

$$P_{\text{verde}} = (2,5 + 2,8V) \cdot 300 \text{ mA} = 0,75 + 0,84 \text{ W}$$

$$P_{\text{blu, bianco}} = (3,2V + 3,4V) \cdot 300 \text{ mA} = 0,96 + 1,02 \text{ W}$$



dove tra parentesi sono indicate le cadute di tensione minima e massima sui diodi. Considerando i valori massimi, abbiamo una potenza totale pari a:

$$P_{\text{tot}} = 0,75 + 0,84 + 2,04 \text{ W} = 3,63 \text{ W}$$

quindi abbastanza vicina a quanto dichiarato.

Per i LED da 12 watt le cose cambiano un po', visto che la corrente di funzionamento dichiarata ammonta a 700 mA; ora le potenze diventano 1,75W per il LED rosso, 1,96W per il LED verde e 2,38W per blu e blu. Con questi valori, la potenza massima complessiva diviene:

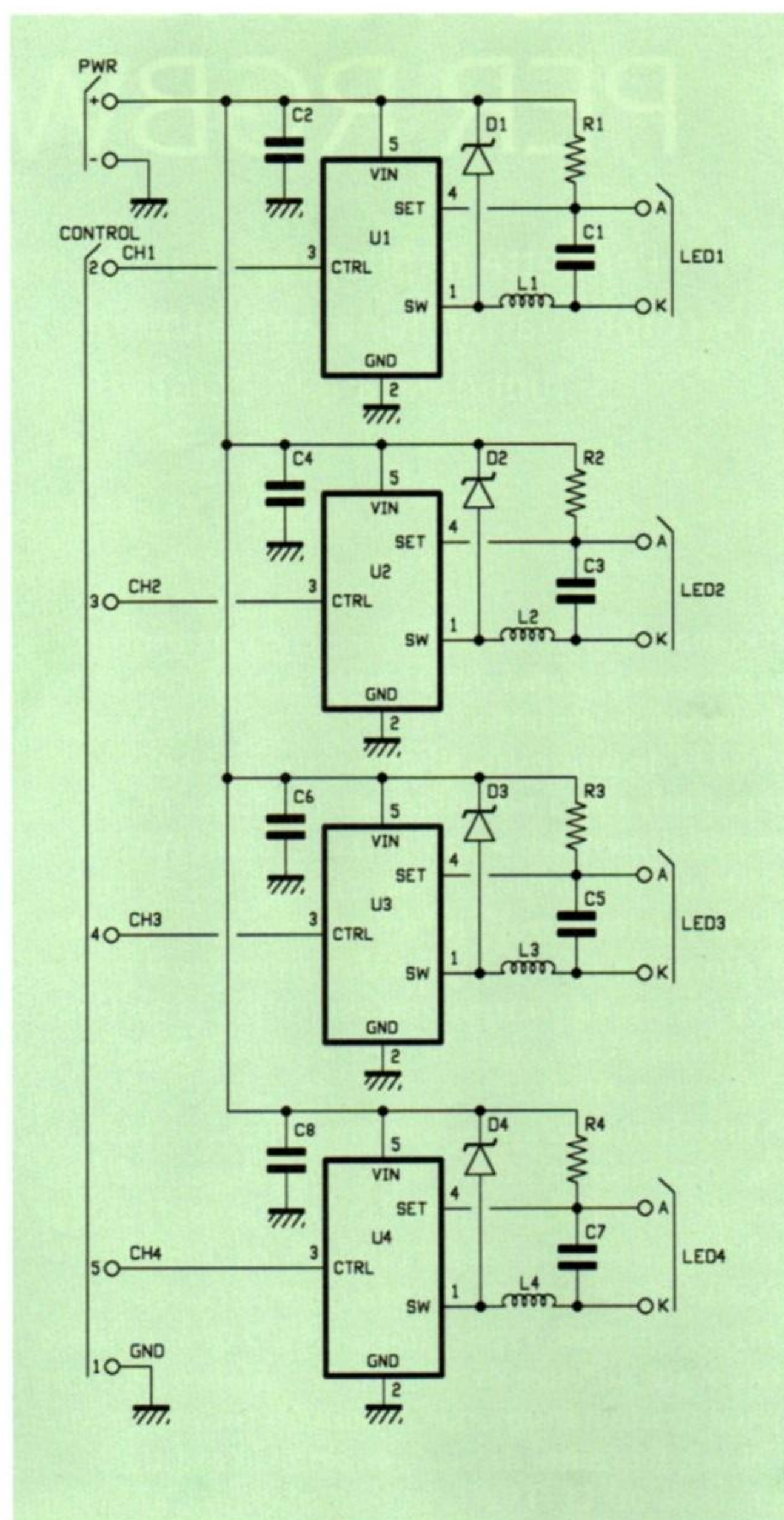
$$P_{\text{tot}} = 8,47 \text{ W}$$

Questo valore è ben inferiore a quanto dichiarato, anche se rimane una potenza notevole.

La potenza dichiarata per i LED RGBW è quindi da intendersi quella massima pilotando ciascun LED alla massima potenza, perciò con correnti di valore differenti.

IL PROGETTO

Come ben noto, i LED vanno pilotati a corrente costante e non a tensione costante, per ottenerne il massimo rendimento ed evitare di "bruciarli" (rammentate che la variazione della corrente nei diodi è esponenziale, quindi piccole variazioni di tensione determinano ampie variazioni di corrente). La tecnica principe per modularne la corrente e quindi la luminosità consiste nell'alimentarli con un segnale PWM, vale a dire una tensione di forma d'onda rettangolare di cui si modula il duty-cycle. Questa tecnica è prediletta perché consente di ottenere rendimenti elevatissimi, dato che per fornire l'alimentazione ai LED - sotto forma di impulsi - è sufficiente utilizzare transistor funzionanti in modo on-off, beneficiando delle minime resistenze serie dei moderni MOSFET e ottenendo così perdite di potenza irrisorie. Se i LED venissero pilotati da regolatori di tensione lineari, quindi con transistor funzionanti da amplificatori, il rendimento scenderebbe tanto più

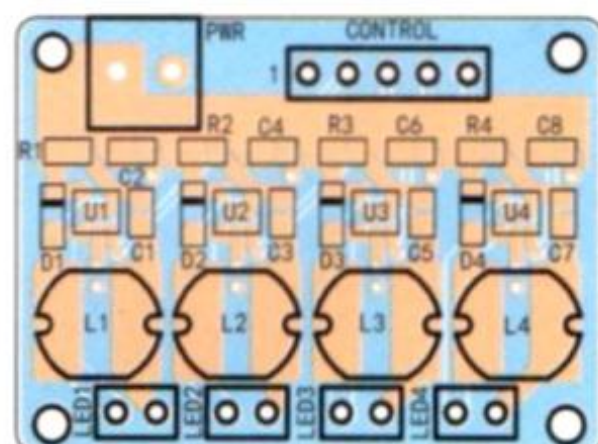
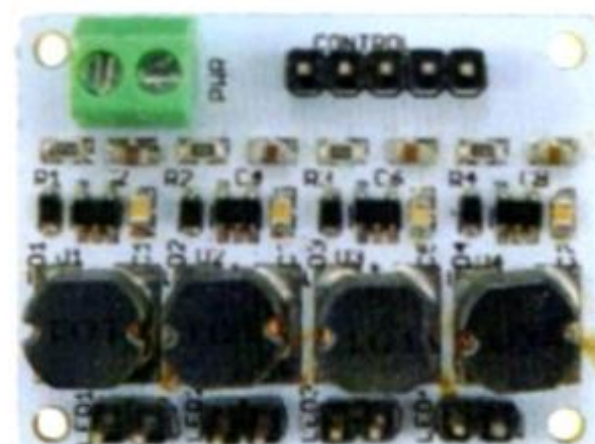


Elenco Componenti:

R1÷R4: 0,3 ohm (0805)
 C1: 1 µF ceramico (0805)
 C2: 2,2 µF ceramico (0805)
 C3: 1 µF ceramico (0805)
 C4: 2,2 µF ceramico (0805)
 C5: 1 µF ceramico (0805)
 C6: 2,2 µF ceramico (0805)
 C7: 1 µF ceramico (0805)
 C8: 2,2 µF ceramico (0805)
 D1÷D4: BAT20JFILM
 U1÷U4: AL8805W5-7
 L1÷L4: Induttanza 33 µH

Varie:

- Strip maschio 2 vie (4 pz.)
- Strip maschio 5 vie (1 pz.)
- Plug alimentazione SMD
- Circuito stampato S1242



quanto la tensione fornita ai diodi sarebbe più bassa di quella di alimentazione, dato che aumenterebbe la caduta di tensione sui transistor a parità di corrente e con essa la potenza dissipata. Alla perdita di rendimento si affiancherebbe quindi la necessità di adottare dei dissipatori, con l'inevitabile incremento di costi e dimensioni dei circuiti driver.

In questo articolo presentiamo quindi un driver per LED a 4 canali con ottime caratteristiche e in grado di pilotare elementi con un assorbimento massimo di 1 Ampere per canale; il nostro circuito è predisposto per ricevere segnali di comando PWM (o anche digitali, a livello logico TTL) ed è in grado di accettare una tensione in ingresso da 6 a 36 volt.

L'elevata tensione di alimentazione supportabile permette inoltre di pilotare più LED in serie, così da realizzare lampade e faretti; per esempio, con una tensione di 36 volt è possibile pilotare un massimo di $36V/3,4V = 10$ LED bianchi in serie per ogni canale.

SCHEMA ELETTRICO

Come si nota, il driver è composto da quattro sezioni perfettamente identiche tra loro, quindi ne descriveremo una sola, fermo restando che quanto detto per essa varrà per le altre.

Il tutto ruota attorno ad un circuito integrato driver creato appositamente per il controllo di LED di potenza, IC1, che è un AL8805 prodotto dalla Diodes. Questo è in pratica un convertitore step-down (buck converter) realizzato in modo da fornire un'uscita a corrente costante, di valore regolabile, perché i LED si controllano in corrente, adeguando

la tensione fornita ai LED al fine di mantenere il valore di intensità corrispondente alla luminosità desiderata: più corrente viene richiesta, maggiore è la tensione fornita e viceversa.

All'interno dell'integrato, lo stadio di uscita che alimenta il LED è composto da MOSFET che fa capo al piedino SW (switch), mentre il piedino SET viene utilizzato per il controllo della corrente in uscita, impostata tramite la resistenza R1 secondo la formula seguente:

$$I_{LED} = \frac{V_{THD}}{R_1}$$

dove I è la corrente che si vuol far circolare nei LED e V_{THD} è una tensione di riferimento interna all'integrato, pari a 100 mV.

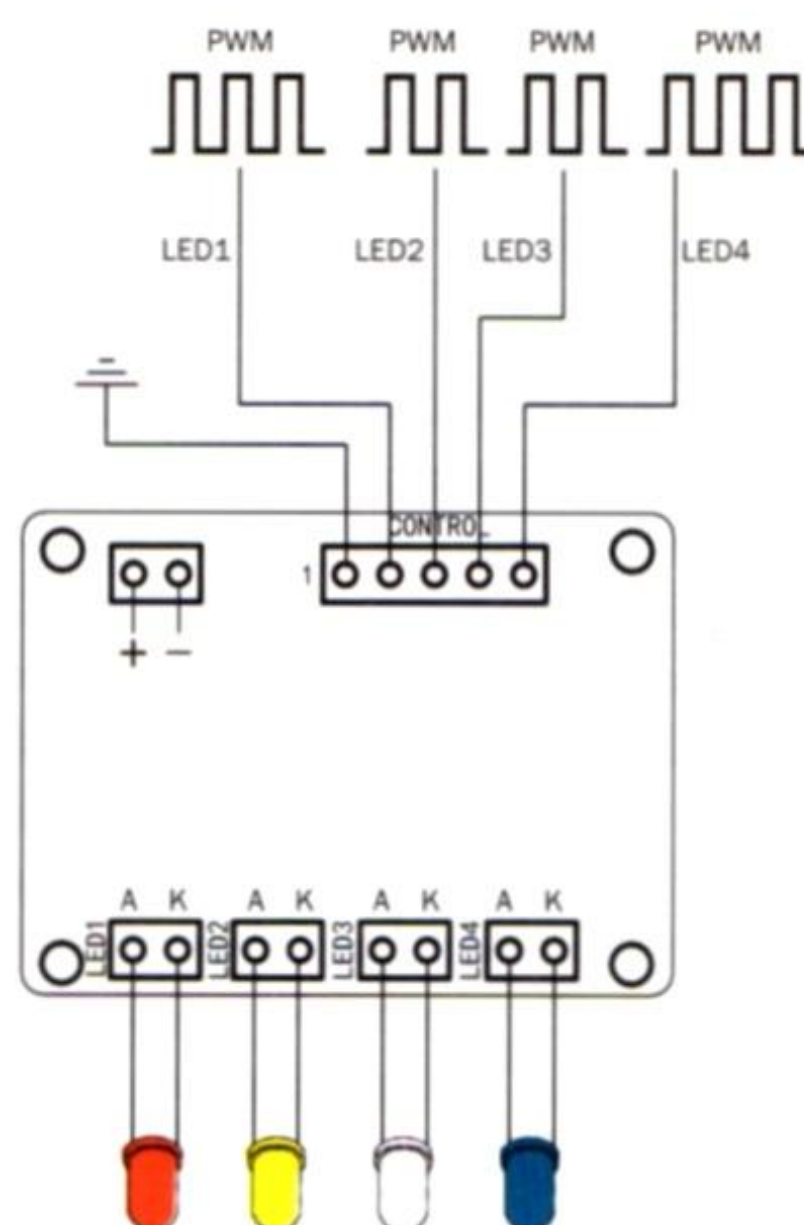
Quindi, volendo far circolare 300 mA nei LED, adatta ai LED da 4 watt di cui abbiamo parlato, occorre inserire una resistenza pari a:

$$R_1 = \frac{V_{THD}}{I_{LED}} = \frac{0.1V}{0.3A} = 0.33\ ohm$$

che è il valore da noi inserito nella scheda, arrotondandolo a 0,3 ohm. Per ottenere una corrente di 700 mA, adatta ai LED da 12 watt, occorre inserire una resistenza pari a:

$$R_1 = \frac{V_{THD}}{I_{LED}} = \frac{0.1V}{0.7A} = 0.14\ ohm$$

Questo valore è facilmente approssimabile tramite



Schema di collegamento del driver e comando tramite i segnali PWM.

Listato 1 MAIN LOOP

```
void loop(void)
{
    // ciclo per luminosità del bianco crescente
    for(int w = 0; w < 256; w++)
    {
        // ciclo per la luminosità (VALUE) crescente
        for(int v = 100; v < 256; v++)
        {
            // ciclo per la tonalità (HUE)
            for(int h = 0; h < 256; h++)
            {
                showColor(h, 255, v, w);
                delay(del);
            }
        }

        // ciclo per la luminosità (VALUE) decrescente
        for(int v = 254; v >= 100; v--)
        {
            // ciclo per la tonalità (HUE)
            for(int h = 0; h < 256; h++)
            {
                showColor(h, 255, v, w);
                delay(del);
            }
        }
    }

    // ciclo per luminosità del bianco decrescente
    for(int w = 254; w >= 0; w--)
    {
        // ciclo per la luminosità (VALUE) crescente
        for(int v = 100; v < 256; v++)
        {
            // ciclo per la tonalità (HUE)
            for(int h = 0; h < 256; h++)
            {
                showColor(h, 255, v, w);
                delay(del);
            }
        }

        // ciclo per la luminosità (VALUE) decrescente
        for(int v = 254; v >= 100; v--)
        {
            // ciclo per la tonalità (HUE)
            for(int h = 0; h < 256; h++)
            {
                showColor(h, 255, v, w);
                delay(del);
            }
        }
    }
}
```

una seconda resistenza da 0,3 ohm saldata sopra quella presente sul PCB, in modo da realizzare un parallelo da 0,15 ohm totali.

L'induttanza L1 e il diodo Schottky D1 costituiscono la rimanente parte del convertitore buck contenuto nell'integrato. Il converter lavora ad una frequenza massima di 1 MHz e consente quindi l'utilizzo di componenti di valori contenuti e piccole dimensioni, sia per l'induttanza che per il condensatore di filtro C1, che serve unicamente a ridurre il ripple in uscita (l'ondulazione della tensione continua fornita al LED) e ad attenuare eventuali interferenze prodotte dal convertitore, per quanto i LED a stretto rigore non ne abbiano bisogno.

Per ultimo, il condensatore C2 costituisce un'indispensabile filtro/disaccoppiamento in ingresso, in modo da far "vedere" all'integrato una tensione continua priva di disturbi ed evitare malfunzionamenti che possono derivare dalla presenza di spike. Parlando dell'impostazione della corrente in uscita effettuata tramite R1, abbiamo ommesso di dire che questa è da intendersi come la corrente massima, quando il chip è impostato per la massima luminosità. La corrente nel LED è in realtà modulabile tramite l'ingresso CTRL, il quale, grazie ad un paio di meccanismi interessanti, permette il controllo completo della luminosità sia tramite un valore di tensione analogico che con un segnale rettangolare in PWM.

Listato 2 CONVERSIONE DA COLORE HSV A RGB

```
RgbColor HsvToRgb(HsvColor hsv) // conversione da colore HSV a colore RGB
{
    RgbColor rgb;
    unsigned char region, p, q, t;
    unsigned int h, s, v, remainder;
    if (hsv.s == 0)
    {
        rgb.r = hsv.v;
        rgb.g = hsv.v;
        rgb.b = hsv.v;
        return rgb;
    }
    h = hsv.h;
    s = hsv.s;
    v = hsv.v;
    region = h / 43;
    remainder = (h - (region * 43)) * 6;
    p = (v * (255 - s)) >> 8;
    q = (v * (255 - ((s * remainder) >> 8))) >> 8;
    t = (v * (255 - ((s * (255 - remainder)) >> 8))) >> 8;
    switch (region)
    {
        case 0:
            rgb.r = v;
            rgb.g = t;
            rgb.b = p;
            break;
        case 1:
            rgb.r = q;
            rgb.g = v;
            rgb.b = p;
            break;
        case 2:
            rgb.r = p;
            rgb.g = v;
            rgb.b = t;
            break;
        case 3:
            rgb.r = p;
            rgb.g = q;
            rgb.b = v;
            break;
        case 4:
            rgb.r = t;
            rgb.g = p;
            rgb.b = v;
            break;
        default:
            rgb.r = v;
            rgb.g = p;
            rgb.b = q;
            break;
    }
    return rgb;
}
```

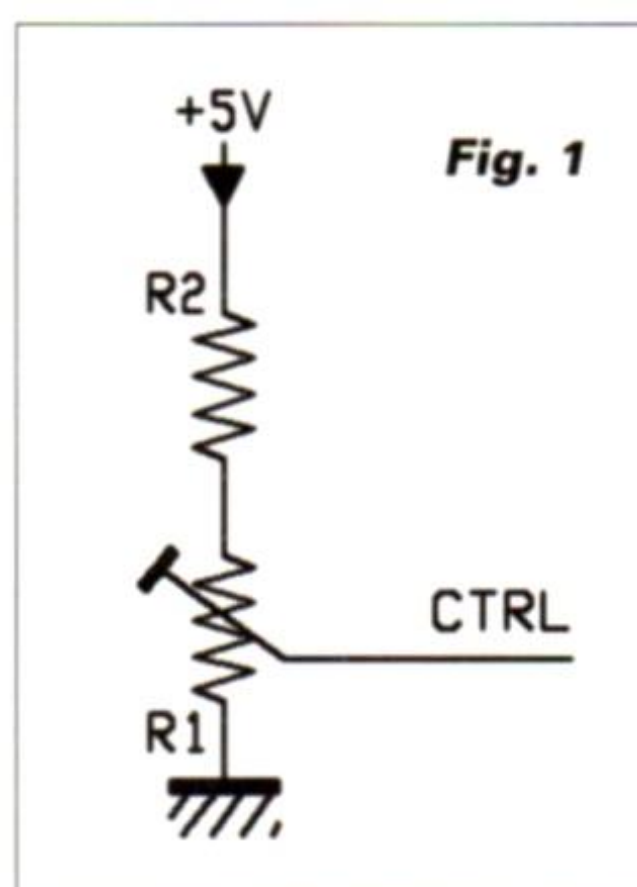
In pratica, il controllo si può effettuare così:

- quando al piedino CTRL è applicata una tensione continua inferiore a 0,4 volt, il convertitore si spegne e l'uscita va a zero volt;
- applicando a CTRL una tensione continua di valore compreso tra 0,4 e 2,5 volt, il convertitore varia la corrente in uscita da zero al massimo impostato tramite la R1, proporzionalmente alla tensione in ingresso;
- se la tensione continua applicata a CTRL supera i 2,5 volt, il convertitore mantiene l'uscita al massimo preimpostato tramite R1;
- applicando, invece, un segnale PWM sull'ingresso CTRL, con l'accortezza che il valore basso del

medesimo sia minore di 0,4 volt ed il segnale alto maggiore di 2,5 volt (ottenibile con logiche sia a 3,3 volt che a 5 V) l'uscita viene impostata secondo il duty-cycle del segnale stesso, con una risoluzione di 1.024 passi.

Risulta quindi semplicissimo realizzare un controllo analogico tramite un potenziometro, come mostrato in **Fig. 1**. Ruotando il potenziometro (siglato R1 nella predetta figura) è possibile variare la luminosità dal minimo al massimo, in maniera molto semplice e senza bisogno di circuiti di pilotaggio basati su microcontrollore; ciò permette di implementare la funzione dimmer con il minimo

hardware possibile. In questo schema semplificato non si tiene conto della soglia inferiore di 0,4 volt, per cui il potenziometro avrà una zona "morta" all'inizio nella quale il LED risulterà sempre spento. In alternativa è possibile inserire un'ulteriore resistenza tra il potenziometro e la massa in modo



da avere una tensione minima su CTRL pari a 0,4 volt. Con i valori nello schema è sufficiente inserire una resistenza di circa 1,5-1,8 kohm. Passando dall'analogico al digitale, con lo sketch del listato 2 possiamo ottenere tutta la gamma di colori sul LED, variabili con continuità.

La variazione non viene eseguita come di consueto nello spazio colore **RGB** (ovvero facendo variare i singoli colori in 3 cicli nidificati), cosa poco gradevole cromaticamente, ma lavorando nello spazio **HSV** (o **HSL**), ovvero **Hue-Saturation-Value**, **Tonalità-Saturazione-Luminosità**.

Per evitare una durata troppo elevata del ciclo è stata impostata la **Saturazione (Saturation)** al massimo valore (e quindi i colori saranno molto "carichi") e vengono fatte variare solo **Tonalità (Hue)** e **Luminosità (Value)**, con in più un ciclo esterno che aggiunge una componente bianca all'effetto.

Lo sketch è abbastanza semplice; tramite 3 cicli nidificati vengono fatti variare i valori del **Bianco (ciclo esterno)**, della **Luminosità (ciclo intermedio)** e della **Tonalità (ciclo interno)**.

I cicli aggiuntivi sono dovuti al fatto che si vogliono evitare "sbalzi" nei valori passando dal massimo (255) direttamente allo zero; i valori di **Bianco** e **Luminosità** vengono quindi fatti variare *da 0 a 255* e *da 255 a 0*, gradualmente. Il valore della **Tinta (Hue)**, trovandosi su uno spazio circolare, non ha bisogno di questo accorgimento.

La variabile 'del' controlla, come detto sopra, il ritardo nel ciclo più interno e quindi la velocità delle variazioni cromatiche.

Un'ultima nota: il controllo tramite Arduino prevede l'utilizzo delle uscite in PWM che sono piuttosto poche; con un Arduino UNO è possibile controllare solo un LED RGBW in questo modo, cosa piuttosto limitativa.

Per questo ed altri motivi, in un successivo articolo presenteremo uno shield d'espansione dotato di

16 canali PWM, controllabile tramite interfaccia I²C-Bus, che permetterà di utilizzare ben 4 driver in contemporanea, pilotando così quattro LED RGBW o ben 16 LED monocromatici contemporaneamente.

REALIZZAZIONE PRATICA

Chiudiamo spiegando come costruire il driver per LED RGBW, che richiede un circuito stampato a doppia ramatura e fori metallizzati e che fa uso in buona parte di componenti SMD; per questa ragione, la sua costruzione richiede un minimo di manualità con le tecniche di montaggio superficiale e un po' di attrezzatura specifica che comprende un saldatore da non più di 20 watt di potenza, a punta fine, del filo di lega saldante da non più di 0,5 mm di diametro, della pasta flussante, una lente d'ingrandimento e una pinzetta per posizionare i componenti sulla basetta.

Il montaggio si inizia con le resistenze e i condensatori in formato 0805 e prosegue con i diodi e gli integrati, quindi con le induttanze; per agevolare la saldatura ed evitare che la lega saldante cortocircuiti i contatti molto vicini degli AL8805, conviene spalmare della pasta flussante sulle piazzole prima di saldare detti componenti. Il flussante è comunque utile per tutti gli elementi del circuito. La morsettiera è a montaggio tradizionale, come i pin-strip che utilizziamo per le connessioni ai segnali di controllo (o potenziometri di dimming) e ai LED. Completato il montaggio, date uno sguardo con la lente di ingrandimento per vedere se non ci sono cortocircuiti e nel caso rimuoveteli. Ora il driver è pronto per essere utilizzato, dato che non richiede alcuna operazione di taratura.



per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono si possono acquistare presso Futura Elettronica. Il master del circuito stampato può essere scaricato dal sito della rivista così come il listato completo. Il driver di potenza per RGBW (cod. COLIBRI) è disponibile a Euro 19,90. Il modulo LED RGBW 4x1W (cod. RGBWLED1) è in vendita a Euro 2,50, mentre il modulo LED RGBW 4x3W (cod. RGBWLED3) costa Euro 4,80.

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

IL CUSCINO DI S. VALENTINO

di BORIS LANDONI

Ricamiamo un cuscino con un messaggio diretto alla persona amata fatto di filo elettroluminescente pilotato da una scheda che lo farà accendere creando un effetto suggestivo.

La festa degli innamorati di tutto il mondo (ed anche, è inevitabile, del commercio che gli gira intorno...) è, come ogni anno, il momento per pensare a qualcosa di originale che possa fare colpo o stupire. I più fantasiosi possono attingere all'instancabile vena creativa di chi produce gadget d'ogni genere, ma noi elettronici non perdiamo l'occasione per strabiliare unendo questo e quel componente per creare animazioni di vario genere. Come ogni anno, cerchiamo di proporre un progetto a tema e stavolta abbiamo pensato a qualcosa di davvero originale, che coniuga la tradizione con la tecnologia: si tratta di un cuscino con "cucite" degli EL Wire. Il cuscino naturalmente lo scegliete voi e potete applicargli poi gli EL Wire, che sono delle strisce elettroluminescenti, disponibili anche in forma di filamenti (sempre elettroluminescenti)

che utilizziamo per la prima volta con questo progetto. Nel nostro progetto utilizziamo proprio dei filamenti, cogliendo l'occasione per spiegarvi cosa sono e in che modo si accendono. Gli EL Wire, in un numero massimo di 6, vengono controllati da una scheda che descriveremo nei prossimi



paragrafi e che è basata su un microcontrollore PIC e degli integrati driver specifici.

IL PROGETTO

Ma andiamo con ordine: siamo partiti dall'idea di realizzare una scritta "d'amore" con degli EL Wire; la soluzione più comoda sarebbe stata quella di adottare l'immane scheda Arduino, visto che in commercio esistono diversi shield che permettono di realizzare vari effetti luminosi e che contengono un driver specifico funzionante a 110 Vca; tale tensione è richiesta dai filamenti o strip elettroluminescenti per accendersi. L'alta tensione poi viene pulsata ed eventualmente parzializzata mediante fototriac. Ovviamente questa soluzione va bene se gli EL Wire da pilotare sono molto lunghi e quindi al driver da utilizzare è richiesta una buona potenza. Il difetto di tale soluzione è che lo shield non integra l'elevatore di tensione, ma

richiede che l'alta tensione gli sia fornita da un dispositivo esterno (che può essere opportunamente calibrato in base alle esigenze). Volendo una soluzione più economica e compatta (Arduino e gli shield occupano spazio e nel nostro cuscino non ce n'è tantissimo...) abbiamo cercato cosa offriva il mercato ed abbiamo trovato una famiglia di driver per EL wire sviluppata da Microchip (www.microchip.com/ParamChartSearch/Chart.aspx?branchID=9020); alcuni driver sono stati sviluppati -e non è casuale- dopo l'acquisto di Supertex (azienda che produce e commercializza driver per fogli e strip elettroluminescenti) da parte della stessa Microchip. Quindi dove altro avremmo potuto orientare la nostra ricerca? Tra questi driver ne esistono alcuni con costi molto bassi e che non necessitano di bobine esterne per il funzionamento dei converter necessari all'elevamento di tensione; tra essi abbiamo trovato quello

da noi utilizzato nel progetto, che è l'HV852.

Questo integrato è l'ideale per il nostro progetto e se proprio vogliamo, possiamo trovargli un piccolo difetto: quello di non poter pilotare grossi carichi, quindi la lunghezza massima di un EL wire deve essere non superiore a circa 1 metro. E qui apriamo una parentesi: le strisce o filamenti elettroluminescenti sono formati da due superfici variamente sagomate, una delle quali rivestita da materiale elettroluminescente, tra le quali si applica una differenza di potenziale di valore tale da innescare la fluorescenza; in altre parole, quando il campo elettrico supera un certo livello la struttura atomica del materiale fosforescente viene alterata e gli elettroni periferici, debolmente legati al nucleo, vengono estratti dall'azione del campo elettrico, ma poi all'inversione di polarità della tensione di alimentazione (che è alternata) ritornano nei loro legami e restituiscono l'energia ricevuta sotto forma di fotoni, generando quindi della luce. Questo fenomeno comporta un certo assorbimento di corrente che dipende dalla superficie degli elettrodi, perciò, a parità di larghezza, più un EL Wire è lungo o esteso in superficie, maggiore è la corrente che richiede. L'adozione dei driver Microchip risparmia hardware, perché utilizzando dei driver comuni bisogna innanzitutto generare il segnale di controllo, poi produrre i 110 Vca e parzializzare la tensione alternata mediante dei TRIAC disaccoppiati tipicamente mediante fotoaccoppiatori; tutto questo può essere semplificato grazie all'HV852, driver che permette di fare a meno di TRIAC (e quindi fotoaccoppiatori ed eventuali DIAC) e dei driver di potenza. Con l'eliminazione di tutti questi

Il driver EL Wire di Microchip

Per pilotare gli EL Wire serve una tensione relativamente alta, idealmente di un centinaio di volt, e una corrente di pochissime decine di milliampere, sufficienti a sviluppare potenze discrete. Nel nostro progetto la tensione viene fornita da un componente Microchip siglato HV852, che contiene un elevatore di tensione che alimenta un driver a ponte H realizzato a MOSFET, i cui transistor vengono fatti commutare grazie a un segnale di clock sdoppiato fornito da un oscillatore interno (o prelevabile dall'esterno tramite il piedino CLKIN) e poi elevato di livello grazie a un traslatore. L'oscillatore può lavorare a una frequenza compresa tra 50 e 500

Hz, definita dalla resistenza collegata tra il piedino REL e il positivo di alimentazione del chip; in tale range di frequenza deve rientrare l'eventuale segnale fornito da una fonte di clock esterna. Se si opta per il generatore di clock interno, la frequenza di lavoro (f) va calcolata con la formula seguente:

$$f = \frac{(1,5M\Omega) \times (250Hz)}{REL}$$

Ricordiamo che f è la frequenza della tensione alternata fornita tra i punti VA e VB, che costituiscono le uscite del ponte e che alimentano l'EL Wire.

componenti siamo riusciti ad ottenere un circuito in grado di pilotare fino a sei EL wire, il tutto gestito da un PIC16F1574, in dimensioni un po' più ridotte di quelle di una carta di credito.

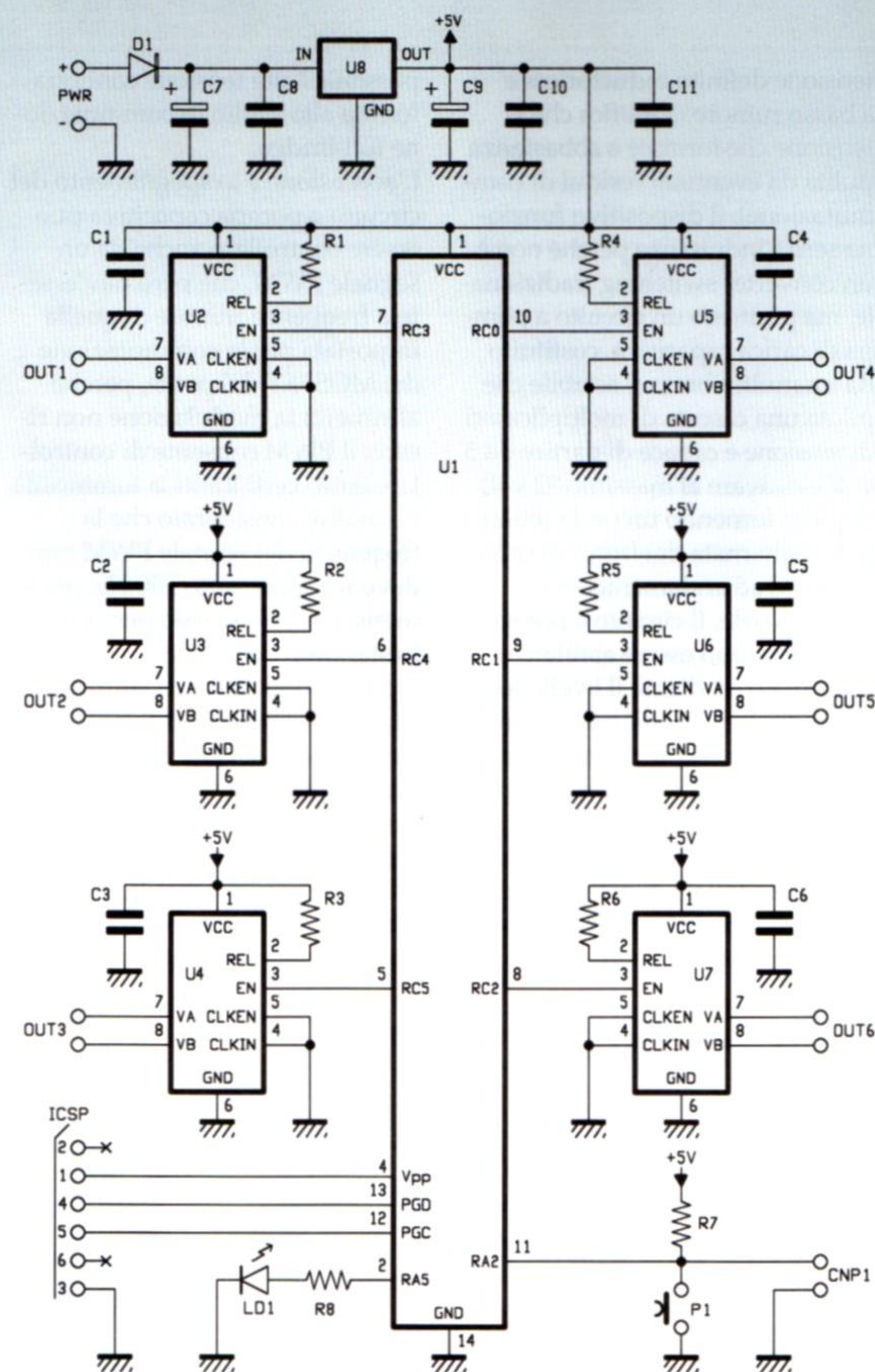
SCHEMA ELETTRICO

Passiamo ad analizzare il circuito elettronico, che è basato sul PIC16F1574 perché ha le caratteristiche che fanno al caso nostro: è basato su un'architettura ad 8 bit con una buona potenza di calcolo (elabora fino a 8 MIPS) e dispone di ben 4 moduli PWM indipendenti a ben 16 bit, con cui è possibile pilotare i driver per ottenere effetti luminosi particolari. Per il momento abbiamo progettato il firmware senza utilizzare i PWM, ma semplicemente costruiamo una sequenza di attivazione delle linee di I/O del microcontrollore, tutte sul registro RC, che sono RC0, RC1, RC2, RC3, RC4 ed RC5. Tutte sono inizializzate come uscite.

La linea RA2 viene inizializzata come ingresso e serve al firmware per rilevare la pressione del pulsante P1, ovvero della chiusura dei contatti CNP1, che può avvenire ad opera di un dispositivo esterno che provvede al telecomando; a riposo, la linea viene mantenuta a livello logico alto dal resistore di pull-up R7. Il pulsante e in generale l'RA2 del micro, è disponibile (previa implementazione nel firmware) per scegliere tra più effetti luminosi composti dai sei EL Wire.

La RA5 viene inizializzata come uscita per il comando del LED LD1, che provvederà alle segnalazioni di stato durante il funzionamento del circuito.

Il firmware viene caricato on-board una volta montato il microcontrollore, tramite il consueto connettore ICSP, cui fanno capo le linee Vpp (tensione di programmazione) PGD (clock della connes-



sione dati seriale) e PGD (canale dati per la programmazione). L'alimentazione dell'intero circuito viene prelevata dai contatti +/- PWR, cui si fornisce una tensione di valore compreso tra 9 e 12 volt; il diodo D1 protegge dall'inversione di polarità il resto del circuito, ovvero il regolatore integrato U8, il quale, contornato

dai condensatori che filtrano l'alimentazione da disturbi impulsivi ed eventuali ripple (C7 e C8 valgono per l'ingresso e C9-C10 per l'uscita), fornisce 5 volt stabilizzati al microcontrollore e ai sei driver per EL Wire.

E veniamo proprio ai driver, per spiegare come vengono utilizzati: ognuno contiene un elevatore di

tensione definito inductorless e a basso rumore (significa che la tensione che fornisce è abbastanza pulita da eventuali residui di commutazione); il dispositivo funziona senza induttanze perché non è un converter switching tradizionale, ma piuttosto un circuito a pompa di carica capacitiva, costituito da un multivibratore astabile che pilota una cascata di moltiplicatori di tensione e capace di partire da 5 volt e ricavare al massimo 92 volt di picco fornendo un'onda rettangolare alternata dai fronti di salita e discesa ad andamento simil-esponenziale. Il circuito a pompa capacitiva può essere abilitato e disattivato mediante il livello logico applicato al piedino EN (l'uno logico abilita e lo zero disabilita) il che permette di accenderlo e spegnerlo all'occorrenza e comunque di gestirne l'attività mediante un semplice segnale digitale. L'elevatore di tensione è retroazionato, in modo da stabilizzare quanto

possibile l'alta tensione continua fornita allo stadio di commutazione full-bridge.

L'accensione e lo spegnimento del circuito a pompa capacitiva può essere controllato anche da un segnale PWM, che però dev'essere a frequenza minore di quella impostata per la commutazione dei MOSFET del ponte, perché altrimenti la modulazione non riesce; il PWM consente di controllare entro certi limiti la luminosità. Va inoltre considerato che la frequenza del segnale PWM non deve scendere sotto i 50 Hz, altrimenti il nostro occhio percepisce lo sfarfallio della luce.

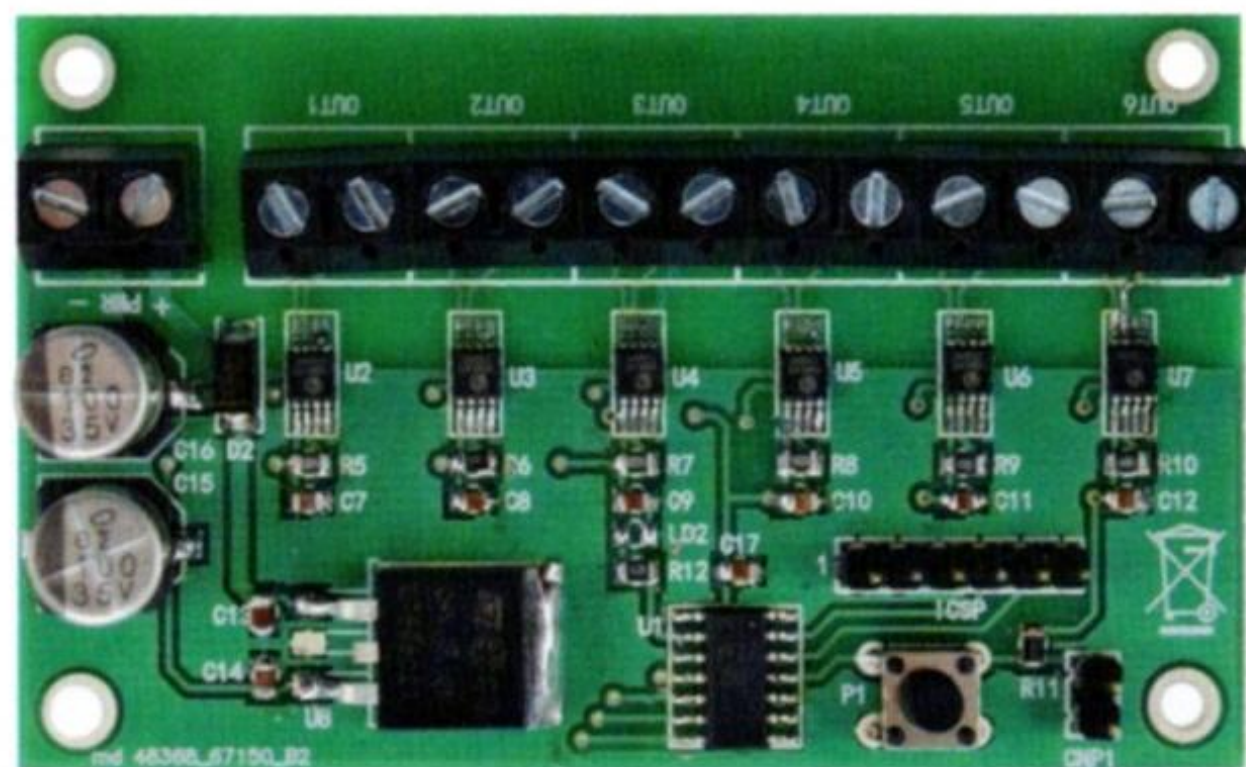
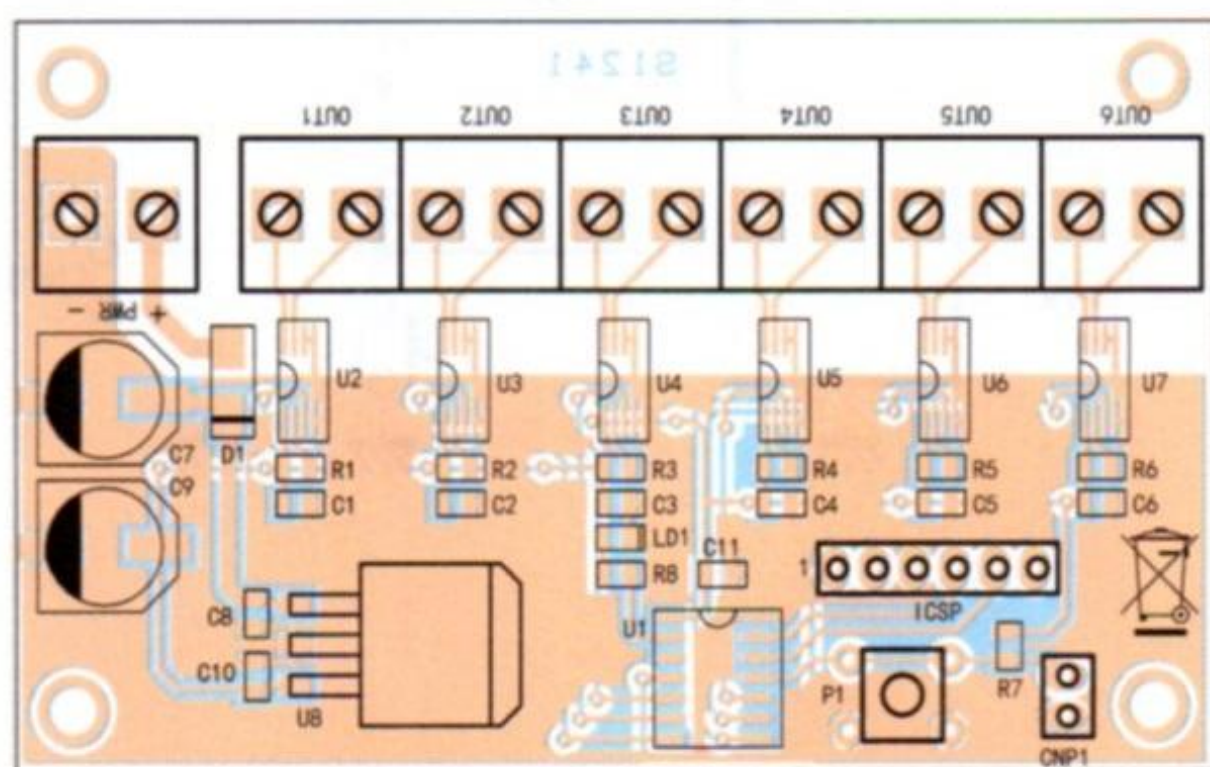
Il circuito elevatore di tensione alimenta un ponte a MOSFET alle cui uscite va collegata la lampada: la struttura a ponte consente di trasformare la tensione continua fornita dall'elevatore in alternata, dato che può invertire la polarità tra i propri capi di uscita facendo accendere alternativamente

il transistor superiore di un lato e l'inferiore di quello opposto e viceversa.

Siccome nasce per pilotare EL Wire, il driver funziona a basse frequenze: tipicamente tra 50 e 500 Hz; il clock può essere fornito dall'esterno tramite il piedino CLKIN (il segnale dev'essere un'onda rettangolare a livello TTL) e in questo caso va posto a livello logico alto il piedino CLKEN (che abilita la ricezione del clock dall'esterno e blocca il multivibratore interno) e a massa il pin REL, ovvero si può ricavare internamente definendone la frequenza (f) mediante la resistenza applicata tra il piedino REL e il positivo di alimentazione. REL può assumere valori che vanno da 7,5 Mohm a 0,75 Mohm e nel nostro caso vale 375 Hz.

L'integrato HV852 (l'elevatore di tensione e il driver a ponte) è dimensionato per alimentare EL Wire che abbiano una superficie

[piano di **MONTAGGIO**]



Elenco Componenti:

R1 ÷ R6: 1 Mohm (0805)

R7: 4,7 kohm (0805)

R8: 470 ohm (0805)

C1 ÷ C6: 100 nF ceramico (0805)

C7, C9: 100 µF 16 VL elettrolitico (E)

C8, C10, C11: 100 nF ceramico (0805)

LD1: LED verde (0805)

D1: GF1M (DO-214AC)

P1: Microswitch

U1 ÷ U6: HV852MG-G

U7: PIC16F1574-I/SL (MF1241)

U8: L7805CD2T-TR

Varie:

- Morsetto 2 poli passo 5mm (7 pz.)
- Strip maschio 2 vie passo 2.54mm
- Strip maschio 6 poli 2.54mm
- Circuito stampato S1241

massima di 14,5 cm², ovvero la cui capacità sia di valore non superiore a 5,3 nF. Si parla di capacità perché i fogli e le strip elettroluminescenti, essendo composti da due elettrodi affacciati l'uno verso l'altro, si comportano elettricamente come condensatori e quindi hanno una loro capacità (C) che, come per i condensatori, è data dalla formula:

$$C = \epsilon \times s / d$$

dove ϵ è la costante dielettrica assoluta del materiale interposto fra gli elettrodi, s la superficie di questi ultimi e d la distanza tra essi. Restando ϵ costante, come fissa è la distanza, chiaro che aumentando la superficie o comunque la lunghezza dell'EL Wire, aumenta la capacità.

Il limite alla capacità e quindi alla superficie dipende altresì dal fatto che il driver fornisce tensione alternata e che in alternata i condensatori conducono elettricità, assorbendo quindi una corrente che dipende dalla frequenza dell'alternata stessa; per l'esattezza, va di pari passo con la frequenza. Se la tensione di alimentazione fosse sinusoidale, l'impedenza (in realtà è una reattanza capacitiva) presentata dagli EL Wire sarebbe pari a:

$$X_c = 1 / 6,28 \times f \times C$$

dove X_c è la reattanza capacitiva, f la frequenza e C la capacità. Frequenza a parte, è evidente che l'assorbimento dipende, a parità di tensione applicata, dalla superficie degli elettrodi, dato che essa determina la corrente che scorre. Nel nostro circuito i driver sono tutti configurati allo stesso modo, ovvero per funzionare con il clock interno (fissato in 375 Hz); l'alimentazione di ciascuno è filtrata mediante un condensatore,

Abbreviazione di Electroluminescent Wire, il termine indica delle strip o dei filamenti elettroluminescenti formati da un filo di rame (core) attorno al quale è deposto uno strato di fosfori, su quale sono avvolti a spirale due sottili fili di rame (sottili perché non deve disturbare l'emissione della luce da parte dei fosfori) che costituiscono gli elettrodi. Il tutto è passivato da uno strato trasparente in plastica e da uno strato colorato in PVC. Tale struttura può anche essere simile a quella dei fogli elettroluminescenti o sviluppata sotto forma di striscia.

Siccome le colorazioni ottenibili dai fosfori sono limitate, per ottenerne di diverse si miscelano, nello strato in PVC, sostanze organiche in grado di generare una luce di un determinato colore se stimulate da quella dei fosfori all'interno della struttura. Lo strato organico consente ad esempio di ottenere colori come rosso e viola quando viene eccitato da una luce -prodotta dall'interno- di colore blu.

Applicando una tensione alternata ai filamenti, si crea un campo elettrico tra i due, che causa lo scorrimento di una corrente durante la variazione di tensione; gli elettroni che la compongono derivano dal fatto che il campo elettrico energizza gli elettroni periferici degli atomi costituenti i fosfori e gli elettroni periferici vengono momentaneamente allontanati; siccome la tensione di alimentazione fornita è alternata, quando essa scende di potenziale fino a invertirsi, gli elettroni tornano al loro posto e restituiscono l'energia immagazzinata sotto forma di fotoni (particelle

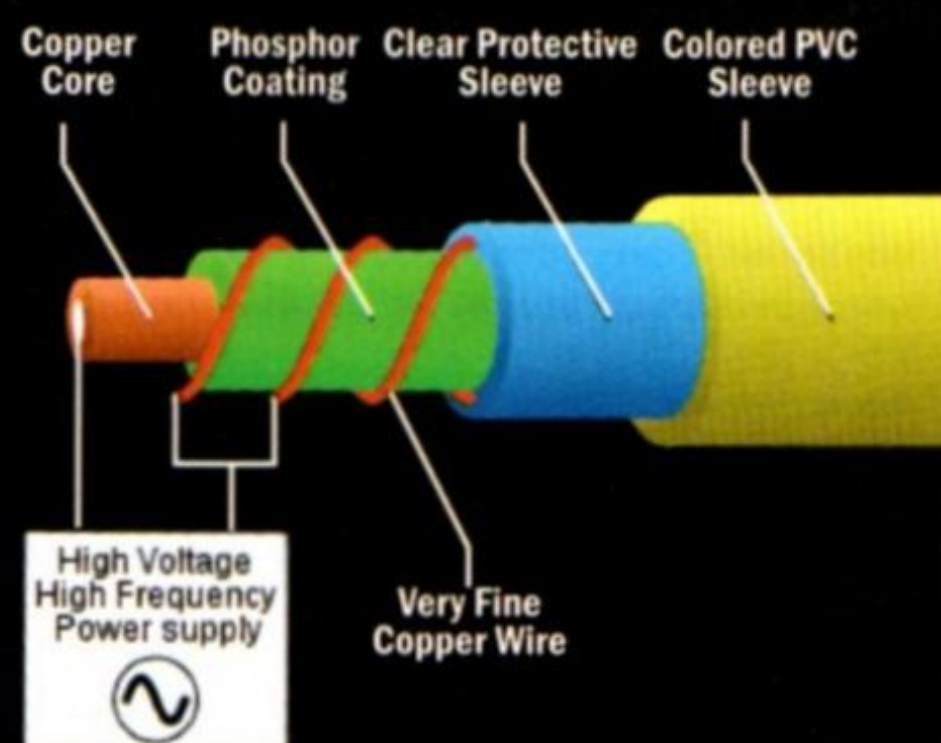
elementari di luce) che formano la luce visibile. La colorazione della luce dipende dal tipo di materiale fosforescente utilizzato, ovvero dall'energia di ionizzazione (quella che serve a estrarre gli elettroni dai loro atomi e ceduta poi per la liberazione dei fotoni): è tanto più vicina al viola quanto maggiore è l'energia e tende al rosso se l'energia è minore. Nel caso sia presente lo strato esterno fosforescente deposto per cambiare la colorazione, i fotoni emessi dai fosfori all'interno colpiscono gli atomi del materiale esterno eccitandoli e provocando l'emissione di un fotone del colore previsto dal fosforo esterno.

Tipicamente la capacità delle EL Wire a filamento è di 3,3 nF ogni metro di lunghezza.

Per funzionare le EL Wire necessitano di una tensione intorno ai 100V.

In commercio si trovano diversi dispositivi che utilizzano questi EL Wire: li vende ad esempio Futura Elettronica (www.futurashop.it/index.php?route=product/search&filter_name=elettrolu).

Composizione di un tipico filamento EL Wire.



Listato 1

```

/*
 * File:   main.c
 * Author: boris-landoni
 *
 * Created on 13 agosto 2015, 14.16
 */

/* -- includes -- */
#include <stdio.h>
#include <stdlib.h>
#include <xc.h>
// #pragma config statements should precede project file includes.
// Use project enums instead of #define for ON and OFF.

// CONFIG1

// PIC16F1574 Configuration Bit Settings

// 'C' source line config statements

#pragma config FOSC = INTOSC // Oscillator Selection Bits
                                // (INTOSC oscillator; I/O function on CLKIN pin)
#pragma config WDTE = OFF // Watchdog Timer Enable (WDT disabled)
#pragma config PWRTE = OFF // Power-up Timer Enable (PWRT disabled)
#pragma config MCLRE = OFF // MCLR Pin Function Select (MCLR/VPP pin
                                // function is digital input)
#pragma config CP = OFF // Flash Program Memory Code Protection
                                // (Program memory code protection is disabled)
#pragma config BOREN = OFF // Brown-out Reset Enable (Brown-out Reset disabled)
#pragma config CLKOUTEN = OFF // Clock Out Enable (CLKOUT function is disabled.
                                // I/O or oscillator function on the CLKOUT pin)

// CONFIG2
#pragma config WRT = OFF // Flash Memory Self-Write Protection (Write protection off)
#pragma config PPS1WAY = ON // PPSLOCK bit One-Way Set Enable bit
                                // (PPSLOCKED Bit Can Be Cleared & Set Once)
#pragma config PLLEN = OFF // PLL Enable (4x PLL enabled)
#pragma config STVREN = ON // Stack Overflow/Underflow Reset Enable
                                // (Stack Overflow or Underflow will cause a Reset)
#pragma config BORV = LO // Brown-out Reset Voltage Selection
                                // (Brown-out Reset Voltage (Vbor), low trip point selected.)
#pragma config LPBOREN = OFF // Low Power Brown-out Reset enable bit (LPBOR is
                                // disabled)
#pragma config LVP = OFF // Low-Voltage Programming Enable (High-voltage
                                // on MCLR/VPP must be used for programming)

/* -- defines -- */
#define DELAY_VALUE 3000
/* -- defines -- */
#define FOREVER 1

#define OUT1 PORTCbits.RC3
#define OUT2 PORTCbits.RC4
#define OUT3 PORTCbits.RC5
#define OUT4 PORTCbits.RC0
#define OUT5 PORTCbits.RC1
#define OUT6 PORTCbits.RC2
#define LED PORTAbits.RA5

unsigned char portValue; // our counter variable
unsigned char state; // state variable
/*
void interrupt myIsr(void)
{
    // only process timer-triggered interrupts
    if(INTCONbits.TMR0IE && INTCONbits.TMR0IF) {
        //portValue++;
        state++;
        INTCONbits.TMR0IF = 0; // clear this interrupt condition
    }
}

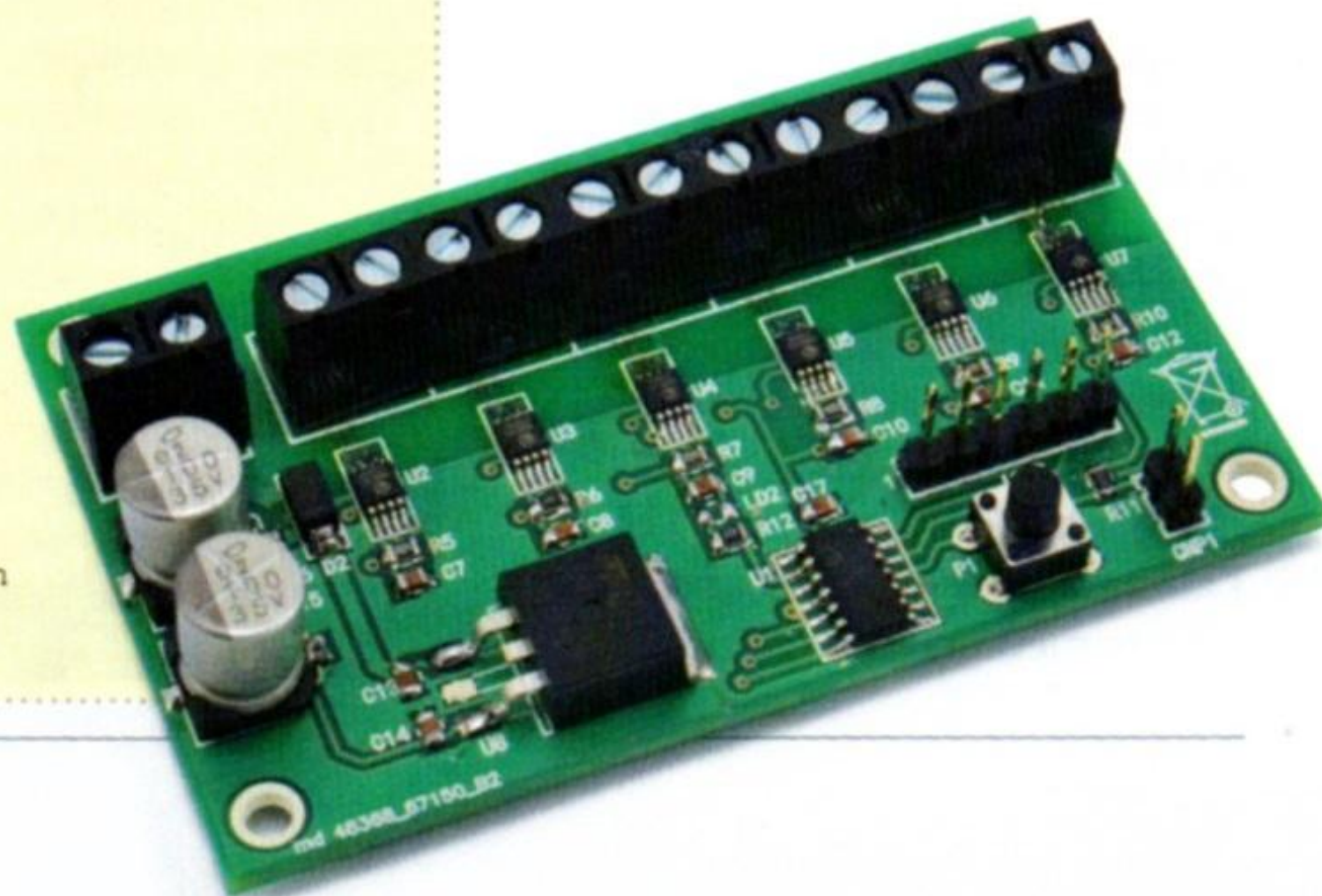
```

in modo da evitare che residui della commutazione degli stadi a ponte di ciascuno si propaghino sulla linea dei 5 volt del circuito, disturbando il funzionamento del microcontrollore.

L'uscita di ciascun driver è collegata a una morsettiere siglata OUT, cui si collegano direttamente i fili della EL Wire.

REALIZZAZIONE PRATICA

Passiamo adesso alla costruzione del circuito, che prevede, per limitare le dimensioni, l'utilizzo di componentistica SMD; fanno eccezione le morsettiere, i pin-strip e il pulsante da c.s. L'utilizzo di componenti SMD impone l'uso di un saldatore a punta molto fine (la potenza non dovrebbe superare i 20 watt) di lega saldante in filo del diametro non superiore a 0,5 mm, nonché di pasta fluxante per agevolare le saldature e limitare la possibilità che si fornino baffi di stagno. Una volta ottenuta la basetta (si ricava per fotoincisione partendo dalle tracce lato rame scaricabili dal nostro sito www.elettronica.in.it) potete disporvi per primi gli elementi passivi, vale a dire diodi e condensatori, lasciando da parte gli elettrolitici, almeno per il momento. Si procede poi con gli integrati HV852 e il microcontrollore, da centrare bene nelle rispettive piazzole, quindi saldare partendo dai piedini posti sugli angoli. A questo punto si



saldano il regolatore 7805, che va appoggiato alla propria piazzola e stagnato bene dietro dopo aver saldato i contatti E ed U (la massa è la placca metallica posteriore, da saldare, appunto, sulla piazzola di massa), il pulsante, i pin strip per la connessione ICSP e l'ingresso CNP1, i due elettrolitici e per finire, le morsettiere a passo 5 mm. Completato il montaggio si può collegare il programmatore e avviare la programmazione del microcontrollore tramite il connettore ICSP.

A questo punto il circuito è pronto per essere utilizzato; per alimentarlo è sufficiente un alimentatore da 9 volt che eroghi una corrente di almeno 200 milliampere, visto che ogni driver assorbe massimo 30 mA (cosa normale, visto che le EL Wire lavorano ad alta tensione). Il circuito, visto che dissipa pochissimo e quindi scalda appena, può essere collocato sul retro (chiuso in un contenitore piatto) o in una tasca ricavata nel cuscino. Le EL Wire vanno collegate con fili che garantiscano il necessario isolamento elettrico, dato che la tensione fornita dal circuito è di oltre 90 volt: fili da 1 mm di diametro garantiscono una guaina di spessore sufficiente ad assicurare l'isolamento.

Attenzione alle giunte: se ne fate, ricopritela con abbondante nastro isolante o con due strati di guaina termorestringente.

Nel nostro progetto abbiamo composto, con sei spezzoni di EL Wire a filamento, la scritta "Ti Amo" circondata da un cuore sopra i cui vertici sono applicate due "onde" la cui accensione idealmente simula la pulsazione; dunque, per ottenere l'effetto desiderato bisogna collegare OUT 1 al filamento con cui componete la scritta "Ti", OUT2 a quello della lettera "A", OUT3 a quello della scritta "mo", OUT4 al filamento che delinea il

Listato 1 - segue

```

}
*/

int main (void)
{
    /*
        //T0CON = 0b10001000;           // enable the timer as 16 bit...
                                         // internal clock, no prescaler

        OPTION_REG = 0b00000111;
        INTCONbits.TMR0IE = 1;         // enable interrupts for timer 0
        ei();                           // enable all interrupts
    */

    //OSCCONbits.IRCF = 0b101;         // Binary value of the three ICRF bits
                                         // to select 4 MHz internal oscillator

    ANSELA= 0;
    ANSELB= 0;
    /* delay counter */
    unsigned long counter = DELAY_VALUE;

    /* set ddr register */
    TRISC = 0x0000;
    TRISA = 0x0000;

    OUT1 = 0; //Ti
    OUT2 = 0; //A
    OUT3 = 0; //mo
    OUT4 = 0; //cuore sx
    OUT5 = 0; //cuore dx
    OUT6 = 0; //pulsazione
    LED = 0;

    /* endless loop */
    while (FOREVER)
    {
        _delay(100000);
        //state++;
        state = 1;
        switch (state)
        {
            case 1:
                OUT1 = ~OUT1;
                LED = ~LED;
                break;
            case 2:
                OUT2 = ~OUT2;
                OUT3 = ~OUT3;
                break;
            case 3:
                OUT4 = ~OUT4;
                OUT5 = ~OUT5;
                break;
            case 4:
                OUT6 = ~OUT6;
                break;
            default:
                state = 0;
                break;
        }
    }

    /* execution should never reach this line */
    return (EXIT_SUCCESS);
}

```


Listato 2 - SKETCH ARDUINO

```
// EL test code

#define heart 6
#define heartup 4
#define ti 7
#define amo 5

unsigned long runtime;
unsigned long time_ta;
unsigned long time_heart;
boolean stateheart=0;
boolean stateta=0;

void setup(){
  for(int i = 4; i<8; i++){
    pinMode(i, OUTPUT);
  }
  time_ta=millis();
}

int tmp = 0;

void loop()
{
  runtime=millis();

  if(runtime>time_ta+200){
    if (stateta==1){
      digitalWrite(7, LOW);
      delay(50);
      digitalWrite(5, LOW);
      stateta=0;
    }
    else {
      digitalWrite(7, HIGH);
      delay(50);
      digitalWrite(5, HIGH);
      stateta=1;
    }
    time_ta=millis();
  }

  if(runtime>time_heart+1500){
    if (stateheart==0){
      digitalWrite(6, HIGH);
      delay(150);
      digitalWrite(4, HIGH);
      stateheart=1;
    }
    else
    {
      digitalWrite(6, LOW);
      delay(150);
      digitalWrite(4, LOW);
      stateheart=0;
    }
    time_heart=millis();
  }
}
```

Anche con Arduino

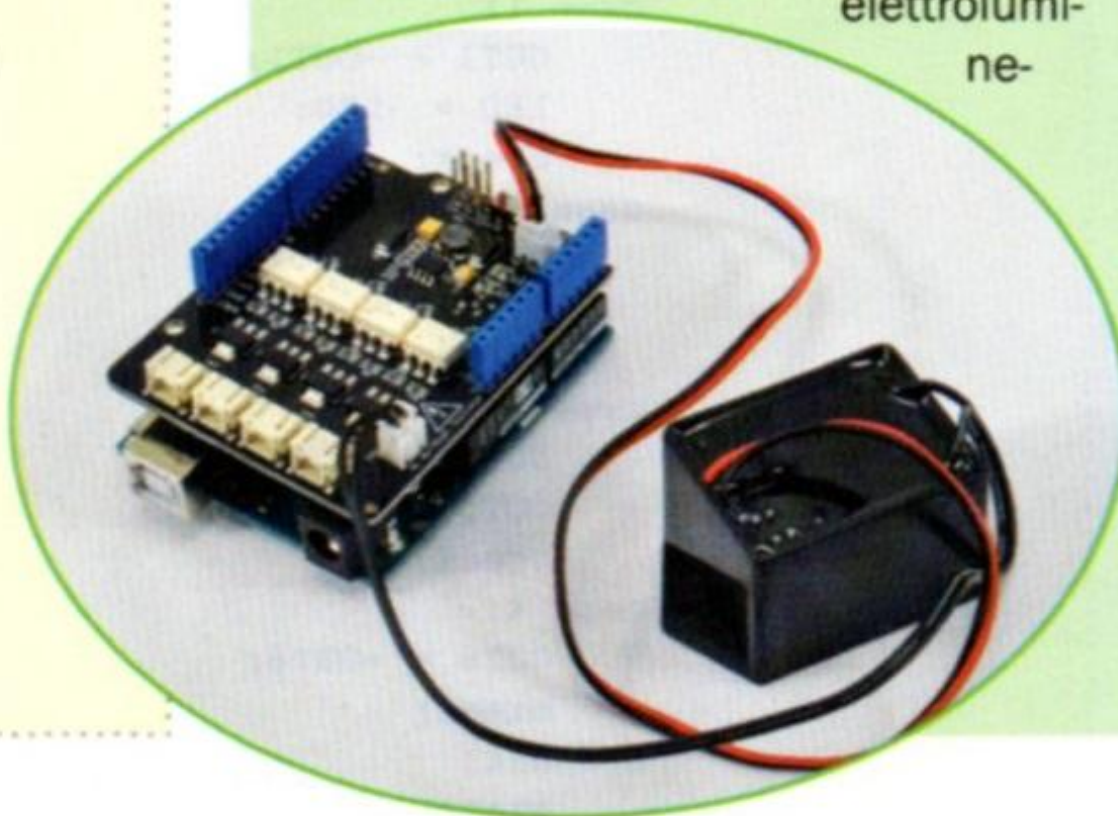
Chi dispone di una scheda Arduino può trovare interessante realizzare con esso, invece che con il circuito basato sul microcontrollore Microchip, il nostro gadget; la cosa è possibile e l'abbiamo sperimentata. Chiaramente cambia il firmware, che peraltro diventa semplicissimo (lo trovate nel **Listato 2**) proprio grazie all'adozione di Arduino e del suo IDE, che semplifica la programmazione: infatti si riduce tutto a una definizione delle variabili e degli I/O, quindi ad un setup. L'hardware del caso è composto da una scheda Arduino (può essere la Uno o la Mega, ma anche la nostra Fishino Uno) e da un apposito shield (EL-shield) commercializzato dalla Futura Elettronica (www.futurashop.it/el-shield-per-arduino-2510-elshield) cui va abbinato un inverter disponibile sempre da Futura Elettronica (www.futurashop.it/potente-inverter-5v-per-elshield-2510-pow10301p).

Lo schema elettrico dello shield è visibile in questo riquadro ed è composto da un regolatore integrato che prende tensione dalla linea Vin e ricava 5 volt che invia alla linea EXT_5V e da essa, tramite una sezione dell'interruttore di accensione SW1 (SW1A) al connettore di alimentazione dell'inverter esterno, che si connette a INV_5V.

L'EL Shield permette di controllare da Arduino fino a 4 strip o filamenti elettrolumi-

scenti con tensione di funzionamento sino a 160 Vca; nel caso specifico, lo shield può pilotare strisce lunghe fino a 15 metri (ricordate il discorso già fatto per i driver Microchip riguardo alla relazione tra lunghezza e capacità). Lo shield dispone di 4 fototriac MOC3063 e 4 TRIAC di potenza in SMD siglati BT134W-600E (600 V, 1 A). Lo shield riceve i segnali di comando dai pin digitali di Arduino e li usa per alimentare le sezioni di ingresso dei quattro MOC3063, cui è affidato il compito di alimentare l'utilizzatore del rispettivo canale. La presenza dei fototriac garantisce il trasferimento degli impulsi da Arduino a una linea ad alta tensione, mantenendo l'isolamento galvanico richiesto per operare in sicurezza. Il MOC3063 è un particolare fotoaccoppiatore con fototriac d'uscita, che integra un circuito di zero-crossing detection, ossia un rivelatore di passaggio per lo zero dell'onda sinusoidale o comunque dell'alternata, in modo da garantire che la conduzione del fototriac interno avvenga sempre lontano dal picco e preferibilmente dopo l'inversione di polarità. Ciò previene i guasti del fototriac che possono sopravvenire per sovraccarico e allunga la durata dei componenti.

Lo shield è predisposto per il collegamento a un inverter specifico per i tubi elettroluminescenti: questo si collega con l'ingresso al connettore INV_5V e l'uscita all'INV_OUTPUT (sono entrambi JST 2.0). Siccome lo shield dispone di sole 4 uscite, inevitabilmente dovete collegare in maniera differente gli EL Wire e, più precisamente: al CH1 la Ti, a CH2 la A e la mo, a CH3 le due strisce o filamenti dei lati del cuore e alla CH4 la striscia sopra il cuore. La sequenza di accensione sarà quindi CH1,



cuore sinistra, OUT5 a quello che delinea il cuore a destra e OUT6 al filamento sovrapposto, che fa la pulsazione.

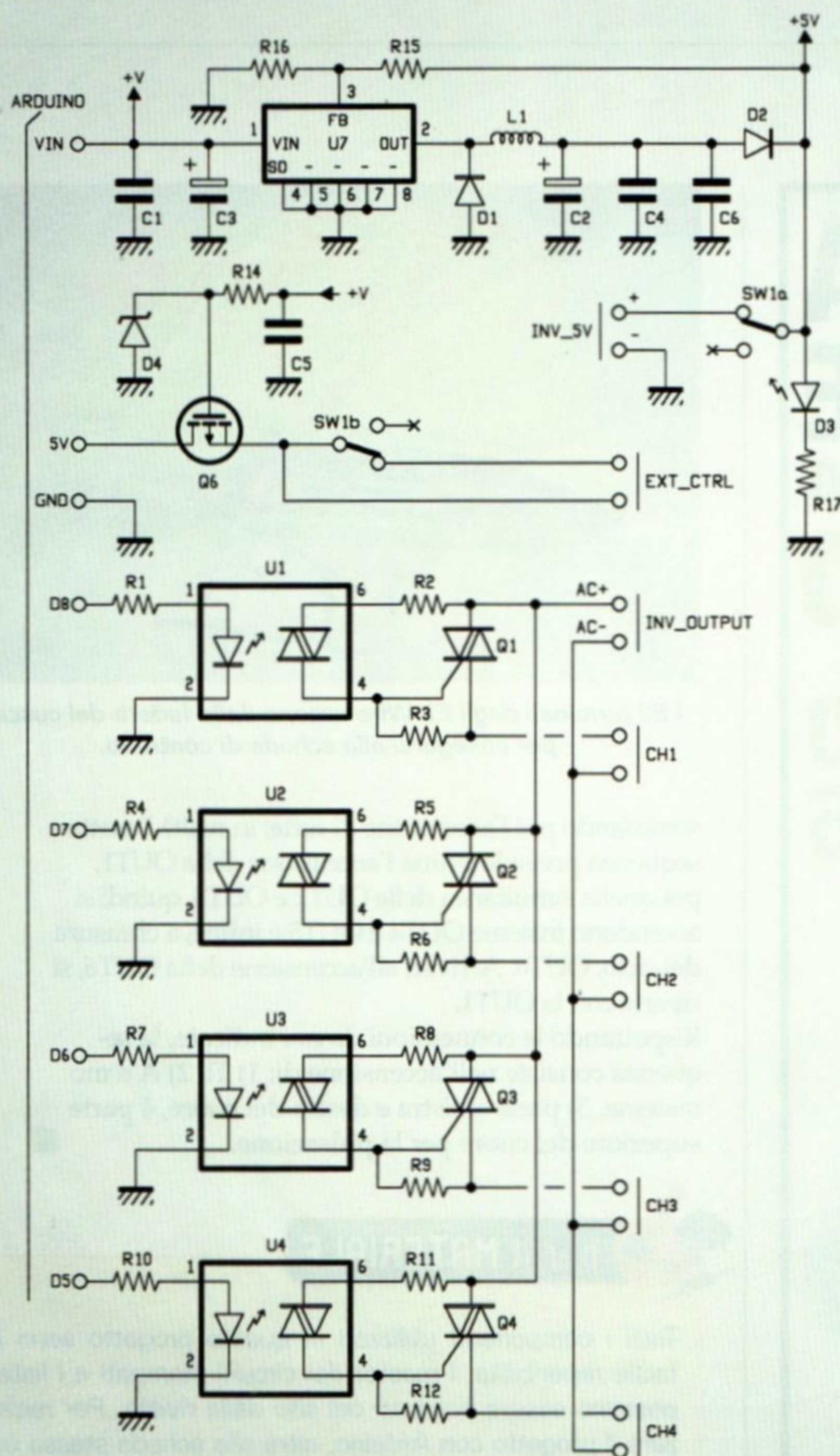
Per fare le scritte, armatevi di pazienza e puntiglio e piegate -senza spezzarle- le EL Wire disegnando la scritta "Ti", poi la "A", poi la

"mo", quindi le metà del cuore e infine la parte alta della pulsazione. Per far passare il filamento e nascondere il collegamento tra le lettere della scritta Ti, tra quelle della mo e tra i segmenti della parte superiore, forate il rivestimento del cuscino e infilateli per

poi farli fuoriuscire dove serve. Traete spunto dalle foto del prototipo, se volete idee. 3,,5, 6, 9, 10, 11

IL FIRMWARE

È giunto il momento di analizzare il firmware per il microcontrollore che, come accennato, per il mo-



Elenco Componenti:

R1: 1 kohm
 R2, R3: 100 ohm
 R4: 1 kohm
 R5, R6: 100 ohm
 R7: 1 kohm
 R8, R9: 100 ohm
 R10: 1 kohm
 R11, R12: 100 ohm
 R13: -
 R14: 4,7 kohm
 R15: 3 kohm
 R16: 1 kohm
 R17: 2,2 kohm
 C1: 100 nF ceramico
 C2: 220 μ F 16 VL tantalio
 C3: 22 μ F 50 VL tantalio
 C4: 10 μ F ceramico
 C5: 10 nF ceramico
 C6: 100 nF ceramico
 Q1÷Q4: BT134W-600D
 Q5: -
 Q6: SI2305
 U1÷U4: MOC3063
 U5: -
 U6: -
 U7: AP1509SOP-8L
 L1: Bobina 22 μ H
 D1: SS54
 D2: -
 D3: LED verde
 D4: MNSZ4689
 SW1: Deviatore 2 vie

Varie:

- Strip M/F 6 vie (2 pz.)
- Strip M/F 8 vie (2 pz.)
- Strip M/F 2x3 vie
- Connettore Filo-Scheda 2 vie 90° (4 pz.)
- Connettore Filo-Scheda 2 vie dritto (2 pz.)
- Circuito stampato

CH1+CH2, CH3, CH1+CH3+CH4.
Chi di voi se la sente può realizzare il controllo in PWM per variare la luminosità delle EL Wire, fermo

restando che degli I/O digitali previsti, solo D5 e D6 possono essere assegnati al PWM (Arduino può mandare il PWM su D3, D5,

D6, D9, D10 e D11); quindi U1 e U2 vanno riassegnati, ad esempio a D9 e D10, scrivendo lo sketch di conseguenza.

mento abbiamo scritto senza utilizzare i PWM, ma semplicemente per creare e ripetere ciclicamente una sequenza di attivazione delle linee di I/O del microcontrollore RC0 (OUT1), RC1 (OUT2), RC2 (OUT3), RC3 (OUT4), RC4 (OUT5) ed RC5 (OUT6). Nulla vi vieta, se

ve la sentite, di implementare il comando in PWM in modo da ottenere, ad esempio, la dissolvenza, tuttavia ricordate quanto detto in precedenza circa la frequenza del segnale PWM e la sua relazione con quella del circuito a pompa capacitiva. Il firmware da caricare

nel PIC lo trovate nel **Listato 1**. e funziona così: sostanzialmente il microcontrollore inizializza un timer per tenere conto della durata dei cicli, quindi attiva le uscite secondo una determinata combinazione, le uscite di comando delle EL Wire, con la cadenza di 1 secondo,

37^a MOSTRA REGIONALE DELL'ELETTRONICA SCANDIANO

(REGGIO EMILIA)

20/21 FEBBRAIO
CENTRO FIERISTICO 2016



**MERCATINO DELLE PULCI
RADIOAMATORIALI**

**TELEFONIA - COMPONENTISTICA
COMPUTER - HI-FI CAR - RADIANTISMO
CB e OM - VIDEOREGISTRAZIONE**

ORARI

**Sabato 20 Febbraio 9 - 18,30
Domenica 21 Febbraio 9 - 18**



COMUNE DI
SCANDIANO

INGRESSO euro 7,50 **RIDOTTO** euro 5
GRATUITO per disabili e ragazzi fino a 14 anni

**INFO: 0522/764302-857436
www.fierascandiano.it**



I fili terminali degli EL Wire escono dalla federa del cuscino per collegarsi alla scheda di controllo.

sommando poi l'accensione di tutte; in realtà l'esatta sequenza prevede prima l'accensione della OUT1, poi quella simultanea delle OUT2 e OUT3, quindi si accendono insieme OUT4 e OUT5 e infine, a chiusura del ciclo, OUT6. Arrivati all'accensione della OUT6, si riparte con la OUT1.

Rispettando le connessioni da noi indicate, la sequenza consiste nell'accensione di: 1) Ti, 2) A e mo insieme, 3) parte sinistra e destra del cuore, 4 parte superiore del cuore per la pulsazione. ■

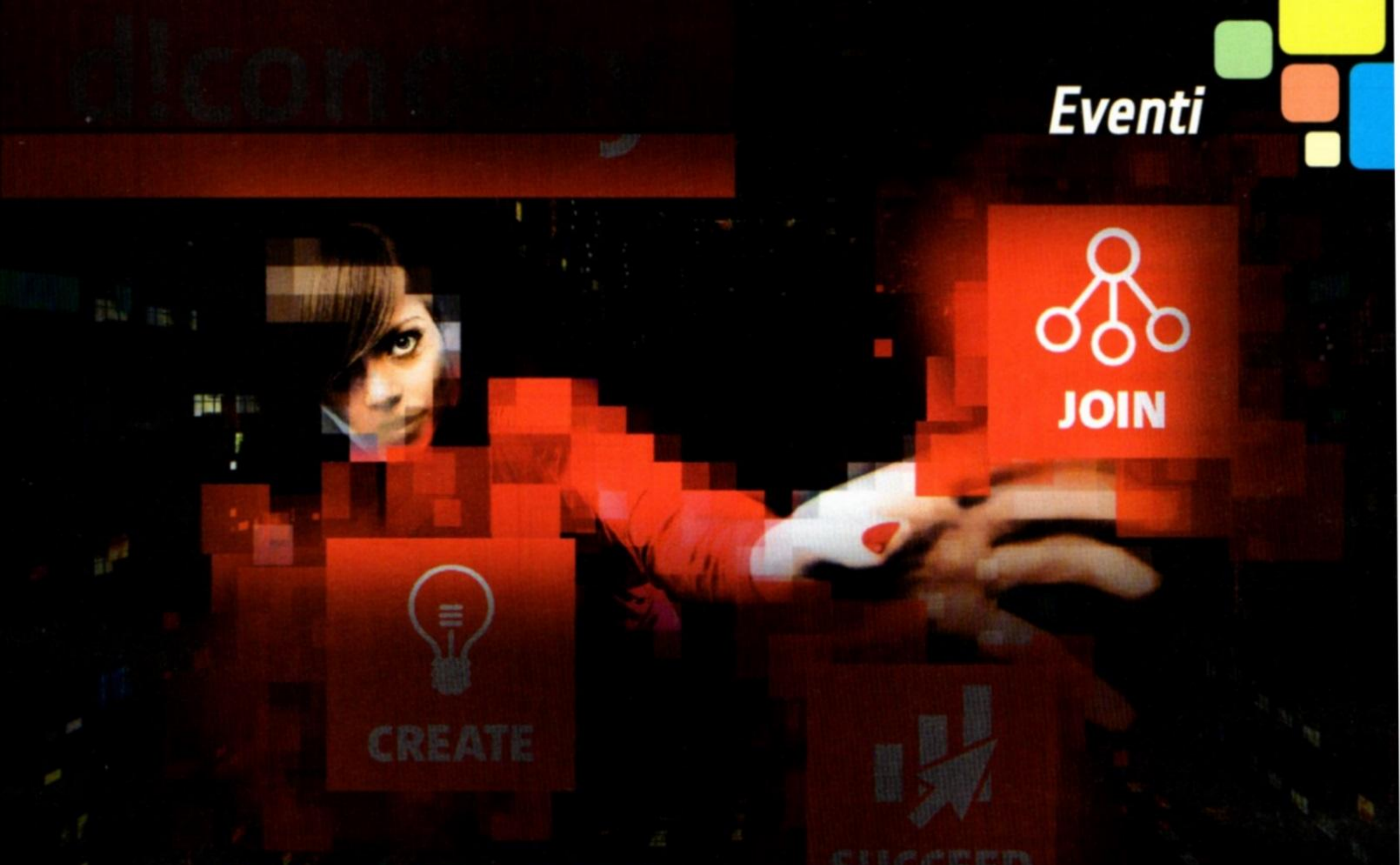


per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono di facile reperibilità. I master dei circuiti stampati e i listati possono essere scaricati dal sito della rivista. Per realizzare il progetto con Arduino, oltre alla scheda stessa occorrono: la scheda El shield per Arduino (cod. ELSHIELD) in vendita al prezzo di Euro 33,00, l'inverter 5V (cod. POW10301P) è venduta al prezzo di Euro 12,00. È disponibile, inoltre, una vasta gamma di cavi elettroluminescenti. Alcuni esempi: cavo elettroluminescente rosso (cod. CAVOELERED), cavo elettroluminescente rosa (cod. CAVOELEPINK), cavo elettroluminescente arancione (cod. CAVOELEARA), cavo elettroluminescente bianco (cod. CAVOELEBIANCO), cavo elettroluminescente blu (cod. CAVOELEBLU) tutti disponibili a Euro 15,00 ciascuno. Tutti questi prodotti si possono acquistare presso Futura Elettronica.

Il materiale va richiesto a:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - Fax: 0331-792287 - www.futurashop.it



IL CeBIT ACCELERA SULLA STRADA DELLA DIGITALIZZAZIONE

di ARSENIO SPADONI

Il tema conduttore della prossima edizione del CeBIT in programma ad Hannover dal 14 al 18 marzo 2016 (d!conomy: join – create – succeed) sancisce la definitiva trasformazione della manifestazione tedesca nel più importante evento mondiale dedicato alla digitalizzazione.

La digitalizzazione sta investendo ormai tutti i settori dell'economia e della società ed è destinata a modificare sempre più rapidamente e in profondità il nostro modo di vivere e di lavorare. Con il tema dell'edizione 2016 "d!conomy: join – create – succeed" il CeBIT punta i suoi riflettori sulle persone e sul ruolo centrale che esse hanno nei processi decisionali e gestionali della trasformazione digitale. La decisione di mantenere anche per il prossimo anno lo stesso tema

conduttore della scorsa edizione – la d!conomy – è stata presa di stretta intesa con l'industria digitale. "La digitalizzazione abbraccia tutti i settori dell'economia e della società. Ben lontano dall'essere un fenomeno circoscritto e di breve durata, è destinato a creare a lungo termine grandi opportunità. Per le aziende vale dunque sicuramente la pena di porsi di fronte a queste nuove prospettive per assicurarsi un futuro di successo." afferma Oliver Frese del CdA di Deutsche Messe. Con lo slogan "d!conomy: join –

Topic Cluster

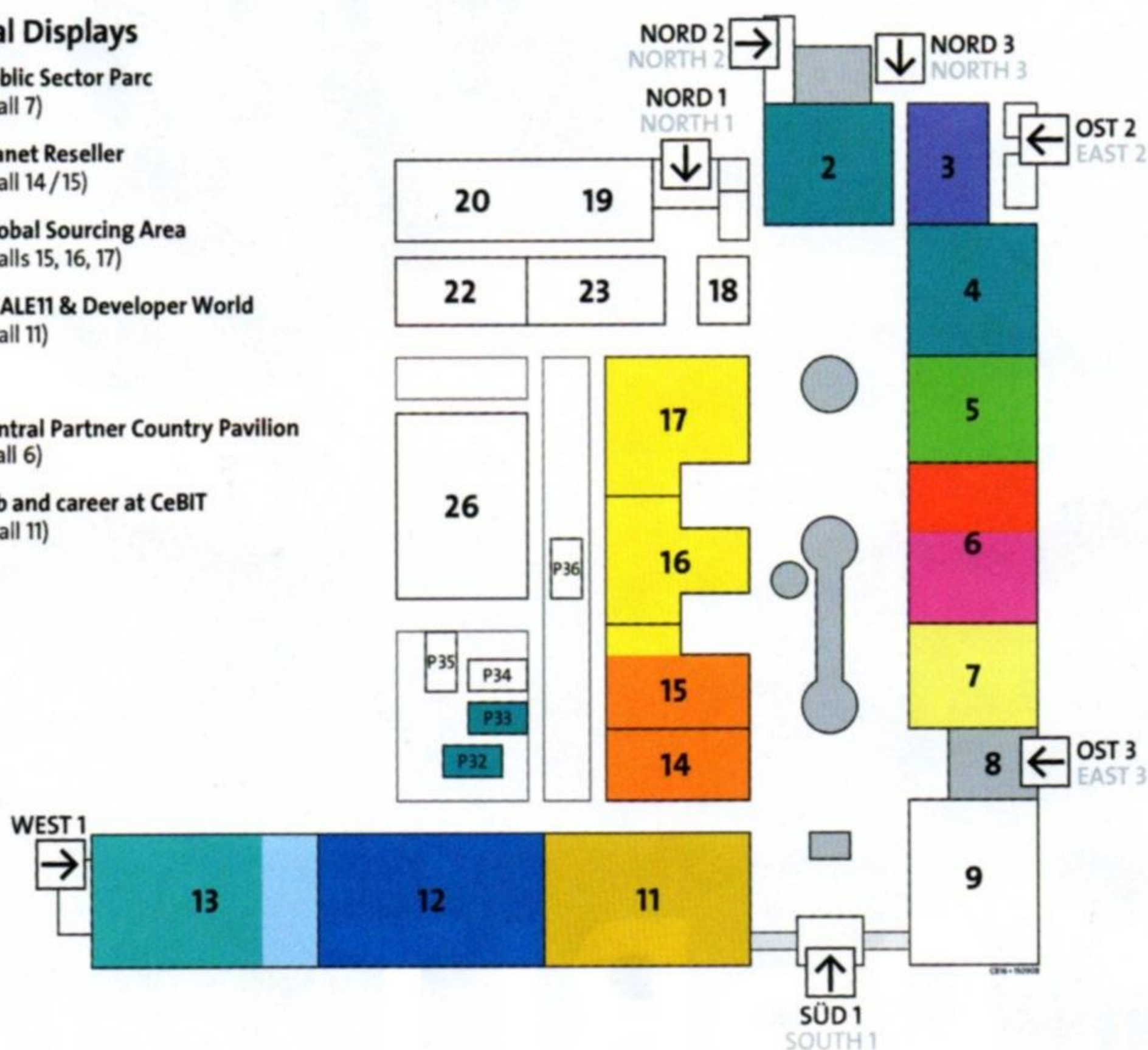
- Digital Business Solutions (Halls 2, 4, Pavilions 32, 33)
- ECM, Input/Output Solutions (Hall 3)
- Business Process Applications (Hall 5)
- Research & Innovation (Hall 6)
- Business Security (Hall 6)
- DatacenterDynamics at CeBIT (Hall 12)
- IoT SOLUTIONS (Hall 13)
- Communication & Networks (Hall 13)

Conference & Lounge Area

- CeBIT Global Conferences (Hall 8)
- CeBIT Executive Club (Hall 8)

Special Displays

- Public Sector Parc (Hall 7)
- Planet Reseller (Hall 14/15)
- Global Sourcing Area (Halls 15, 16, 17)
- SCALE11 & Developer World (Hall 11)
- Central Partner Country Pavilion (Hall 6)
- Job and career at CeBIT (Hall 11)



create – succeed” gli organizzatori descrivono un processo in atto, interpellando così ogni responsabile decisionale, ma anche ogni azienda – indipendentemente dalla sua appartenenza settoriale – in ogni singola fase della trasformazione digitale. “Recenti studi dimostrano che soprattutto le piccole e medie imprese esitano ancora ad accostarsi al trend digitale”. Secondo uno studio della società tedesca di consulenza strategica Roland-Berger, sino ad oggi solo un’azienda su due ha preso in esame le implicazioni della digitalizzazione per il proprio modello di business e circa due terzi delle aziende interpellate hanno ammesso di avere una bassa preparazione digitale. “La digitalizzazione influirà sugli anelli della catena del valore di ogni settore dell’industria. Bisogna poi considerare anche l’effetto delle proposte dirompenti di aziende giovani e dinamiche che turbano mercati ormai consolidati con i loro modelli

di business digitale trasformandosi in nuovi concorrenti”. Il tema “*d!conomy: join – create – succeed*” sottolinea anche la natura universale dei vantaggi offerti dalla digitalizzazione, vantaggi che possono essere condivisi da industria, amministrazione pubblica, politica e società. “Ecco perché, all’inizio dell’era digitale, è necessario dare un inquadramento generale alla trasformazione digitale, tenendo conto dei suoi aspetti più importanti, ad esempio i problemi della sicurezza dei dati e della privacy. Devono anche essere sviluppati nuovi modelli di business, nuove interfacce con i clienti e nuovi processi aziendali e di produzione, oltre a nuove forme di comunicazione e di lavoro”, spiega Frese, aggiungendo che “le aziende che vogliono continuare a rimanere competitive in un contesto di mercato così dinamico non possono non cogliere le possibilità offerte dalla trasformazione digitale.”

conclude Frese.

Le informazioni, le ispirazioni e le innovazioni necessarie per cogliere queste opportunità si trovano al CeBIT. Come salone leader mondiale del business digitale, il CeBIT offrirà infatti anche nel 2016 una panoramica completa delle possibilità e delle sfide della digitalizzazione, riservando inoltre spazio alla discussione sui temi più importanti – e senza ignorare temi controversi – connessi ai nuovi sviluppi. Il salone prevede anche numerosi incontri e dibattiti nell’ambito delle **CeBIT Global Conferences** che consentono di conoscere nuove visioni e nuovi concetti proposti da importanti leader dell’industria. I visitatori potranno anche toccare con mano modelli di business dirompenti, ad esempio nella cornice di **SCALE 11**, l’area speciale riservata alle startup.

LA TRASFORMAZIONE DIGITALE DELL'ECONOMIA, DELLA PUBBLICA AMMINISTRAZIONE E DELLA SOCIETÀ

Mentre altre manifestazioni di settore si concentrano su di una specifica area, come l'elettronica di consumo o gli ultimissimi smartphone, solo il CeBIT mette al centro della sua attenzione i tre importanti trend della digitalizzazione: la digitalizzazione di imprese e settori industriali, la digitalizzazione della pubblica amministrazione, la digitalizzazione della società.

Grazie al suo concept, il CeBIT è forse l'unica manifestazione a livello mondiale ad affrontare sia gli aspetti economici sia gli aspetti di politica sociale internazionale della digitalizzazione. Al centro della sua attenzione ci sono la trasformazione digitale di tutti i settori industriali, Internet delle Cose e la sicurezza dell'economia digitalizzata e della società. Per questo anche politici e scienziati di tutto il mondo guardano al salone di Hannover come alla piattaforma di riferimento per lo sviluppo di nuove idee e per la

discussione su nuove opportunità e sfide.

L'abbinamento tra ambito espositivo e ambito congressuale consente di presentare, discutere e spiegare tutti gli aspetti della digitalizzazione in un'ottica applicativa. Le numerose conferenze, i convegni e forum offrono ampio spazio per discutere delle opportunità che si offrono e del contesto economico generale e di politica sociale che esse richiedono.

L'edizione 2015 ha registrato rispetto all'anno precedente un incremento del sei per cento circa nell'affluenza dei visitatori e nella superficie espositiva. Soprattutto nei settori chiave del CeBIT – quelli che hanno un ruolo centrale nella digitalizzazione dell'economia, ad esempio Enterprise Content Management (ECM), Enterprise Resource Planning (ERP), Business Security, Communication & Networks o Human Resources (HR), gli espositori si sono detti molto soddisfatti dei risultati ottenuti.

Paese Partner dell'edizione 2016 sarà la **Svizzera**. Il CeBIT pun-

terà così i suoi riflettori su una nazione che è punto di riferimento mondiale nell'impiego delle tecnologie digitali. Con il loro eccellente know-how e con le loro soluzioni altamente innovative, le aziende e le istituzioni svizzere daranno importanti impulsi al settore e alle industrie presenti. La Svizzera è uno dei Paesi più innovativi del mondo, è hub di fama internazionale nella ricerca e nello sviluppo, oltre ad essere uno dei Paesi più benestanti della Terra – ed è ai primi posti anche nella classifica dei valori connessi alle strategie digitali. Secondo le stime dello *European Information Technology Observatory (EITO)*, il mercato svizzero del software dovrebbe crescere del 5,5 per cento nel 2015, e quello dei servizi per l'IT del 3,3 per cento. Oltre al Paese Partner, anche la **Commissione Europea** arricchirà il CeBIT con la propria presenza. Dopo l'avvio dell'Agenda Digitale Europea "Digital Single Market", l'Unione Europea organizzerà infatti proprio al CeBIT, sotto la direzione del Commissario UE per l'economia e la società





digitali **Günther Oettinger**, una conferenza annuale nel cui ambito esponenti della politica e dell'economia si confronteranno sui progressi del mercato digitale interno. L'attuazione dell'Agenda Digitale è uno dei progetti più importanti e più ambiziosi della Commissione Europea. Obiettivo dell'Agenda è infatti quello di riunire i 28 mercati nazionali europei in un mercato digitale unico. La strategia prevede tra l'altro reti di telecomunicazione europee, servizi digitali transfrontalieri e creazione di innovative startup. Questa conferenza si terrà ogni anno il primo giorno di fiera nella cornice delle **CeBIT Global Conferences** e avrà come temi centrali l'ampliamento delle infrastrutture, la standardizzazione, la sicurezza e le startup. Nel 2016 le CeBIT Global Conferences affronteranno nell'arco di quattro giorni e in tre spazi diversi, i temi: d!conomy, Internet di ogni cosa, Security & Privacy e Tecnologie dirompenti. Oltre al Commissario UE Günther H. Oettinger, hanno già confermato la loro presenza sul fronte dei relatori Christian Fredrikson, CEO di F-Secure, Shoei Yamana, Presidente di Konica Minolta, Inc. (Giappone) e August-Wilhelm

Scheer, CEO del Gruppo Scheer.

IL NUOVO FORMAT DI SCALE11

Con un importante ampliamento di SCALE 11 – la piattaforma per le startup che ha fatto il suo riuscito debutto lo scorso marzo – il CeBIT rafforza la propria posizione di evento leader per l'innovazione e per la proposta di modelli dirompenti di business nell'economia digitale. *"Intendiamo fare di SCALE 11 una piattaforma ancora più importante per l'ecosistema che ruota intorno a innovazioni, giovani imprenditori, aziende consolidate e investitori"*, afferma Oliver Frese. Nel 2016, SCALE11, che il CeBIT promuove in stretta collaborazione con l'Associazione federale tedesca delle startup (BVDS), si terrà all'insegna dello slogan: *"enable the next..."*. Dice Frese: *"Abbiamo investito tempo e impegno nel nuovo format. Al prossimo CeBIT i visitatori troveranno una interessante e stimolante vetrina delle startup all'insegna della creatività"*. Complessivamente parteciperanno al salone oltre 350 giovani aziende, almeno 250 delle quali occuperanno l'area di SCALE 11. Lo slogan *"enable the next..."* riflette alla perfezione, secondo Frese, l'obiettivo di SCALE11: facilitare il prossimo salto di qualità delle startup, ac-

celerare il prossimo stadio dello sviluppo, farsi trampolino di lancio per la prossima "big thing". SCALE11 sarà composta fondamentalmente da cinque spazi. Prima di tutto ci sarà una piattaforma espositiva riservata alle startup. Ad essa si affiancherà una piattaforma per conferenze con relatori internazionali di alto livello. Uno degli highlight sarà lo Startup City Summit, dove rappresentanti delle startup di metropoli internazionali quali Londra, Parigi e Tel Aviv sveleranno come creare ambienti di business favorevoli allo sviluppo di giovani imprese.

Un terzo spazio sarà quello dove giovani imprenditori proporranno i loro modelli innovativi di business a visitatori qualificati del CeBIT e investitori, confrontandosi con finanziatori e banche per trovare fondi attraverso i quali consentire alla loro azienda di crescere e di trovare nuovi partner. Questa tribuna ospiterà anche la fase finale di un concorso europeo per startup dal titolo *"Best of CeBIT"*. All'interno di un secondo evento, il *"Founders Fight Club"*, coppie di fondatori di startup metteranno vivacemente a confronto le loro idee su un vero e proprio ring da boxe. Il quarto spazio di SCALE 11 prevede un programma di workshop organizzati in un'area dedicata del padiglione 11, dove si approfondiranno i colloqui sulle possibilità di finanziamento. Qui i giovani imprenditori potranno fornire maggiori informazioni sui propri modelli di business, o aprirsi a nuovi mercati internazionali confrontandosi direttamente con nuovi potenziali clienti. Per finire – come quinto e ultimo spazio di SCALE 11 – si svolgeranno ogni sera nel padiglione 11 momenti di networking faccia a faccia tra key player delle

comunità delle startup. Sono previsti anche incontri veloci – speed dating – e incontri tra giovani imprenditrici (Female Founders).

La creazione di tre ampi cluster renderà la struttura del padiglione delle startup estremamente trasparente. Il primo comprenderà network di fondatori – aziende affermate e consulenti che si sono messi insieme per istruire e supportare le startup. Il secondo sarà composto da rappresentanti di città internazionali collegate allo sviluppo dell'imprenditoria giovane che presenteranno numerose aziende e riuscite idee di business nate all'interno dei loro ecosistemi di startup. Il terzo cluster prevede invece la presenza di numerose startup che proporranno proprie presentazioni, sfruttando la possibilità di inserirsi in una delle 15 aree a tema previste, quali Big Data, Digital Health, eCommerce, FinTech, Internet of Things o Logistics Media & AdTech. Ognuna di queste aree avrà un "Community Captain" – un player ormai affermato che assumerà il ruolo di mentore per le startup partecipanti. Il team del CeBIT utilizzerà peraltro i dati di registrazione dei partecipanti per assegnare i community captain alle startup il più presto possibile, in modo che il networking nell'ambito delle aree a tema possa prendere il via già prima dell'inizio del CeBIT. Durante il salone, poi, si terrà, la sera di martedì 15 marzo – sempre nel padiglione 11, anche un "Captains Dinner".

Conclude Frese: "Organizziamo SCALE 11 promettendo sicuri risultati. Alle startup promettiamo che potranno testare i loro modelli di business su clienti potenziali in condizioni reali e che troveranno nuovi partner che le aiuteranno a fare un importante passo avanti. Alle

aziende affermate promettiamo che a SCALE 11 scopriranno innovazioni digitali e interessanti e promettenti nuovi modelli di business".

I DATA CENTER DEL FUTURO

La digitalizzazione rivoluziona anche il mercato dei data center, costringendo le aziende a verificare lo stato dei loro sistemi IT per adeguarli alle nuove esigenze. Il CeBIT 2016 metterà in mostra l'intera offerta per i "Data Center del futuro" richiamando su di essa l'attenzione in ambito sia congressuale sia espositivo. Secondo un'indagine condotta dalla società di ricerche di mercato IDC in ambito europeo, i data center in uso presso le piccole e medie imprese hanno di media sette anni di vita. Ne deriva un importante fabbisogno di modernizzazione, perché, in tempi di crescita esponenziale delle quantità di dati da gestire, di sempre più forte pressione dei costi e di accresciute esigenze di sicurezza, l'impiego di infrastrutture obsolete può diventare un vero e proprio problema.

Per questo motivo il CeBIT, in collaborazione con lo specialista britannico delle conferenze sui data center "DatacenterDynamics" (DCD), dedicherà il prossimo anno maggiore attenzione al tema con un apposito spazio, unico nel suo genere, interamente rivolto al data center del futuro all'interno del padiglione 12.

L'offerta degli sviluppi proposti spazierà da centri dati prefabbricati (container based) a soluzioni per l'approvvigionamento intelligente di energia elettrica, la ventilazione e il cablaggio, a componenti singoli per rack. Ai temi "classici" connessi ai data center, si affiancheranno inoltre per la prima volta anche soluzioni di stoccaggio, tecnologie per i server, computing ad alte presta-

zioni e servizi IT cloud-based. Un altro importante settore espositivo sarà quello delle tecnologie per la messa in sicurezza fisica dei centri dati. L'offerta di soluzioni per la protezione dei dati e per la sicurezza virtuale sarà proposta nell'area "Business Security" del padiglione 6.

PIÙ CONFERENZE SPECIALISTICHE E PRESENTAZIONI DI PRODOTTO

Oltre agli ultimissimi prodotti e alle soluzioni da toccare con mano, il pubblico degli addetti ai lavori potrà fruire di un programma di conferenze ancora più ricco. Per la prima volta si discuterà su due diversi palchi dei temi della costruzione, della programmazione e della gestione di centri dati state of the art. Alla conferenza "DCD Converged", più specificamente orientata su temi infrastrutturali, si affiancherà una conferenza dedicata alle problematiche connesse al software e all'open source.

Il titolo del nuovo appuntamento è "StackingIT".

Relatori di alto profilo discuteranno, ad esempio, in questa cornice di vantaggi e svantaggi degli standard open compute e open stack, che gruppi mondiali come Facebook utilizzano già oggi e che assicurano un'ulteriore spinta all'innovazione. Nel 2016 entrambi i palchi, e quindi anche l'intero programma delle conferenze, saranno per la prima volta accessibili gratuitamente ai visitatori del CeBIT.

La manifestazione di marzo verrà preceduta dal CeBIT Preview, il 20 gennaio, con la presentazione alla stampa da parte dei principali espositori delle innovazioni e soluzioni digitali e dei nuovi modelli di business che proporranno in fiera.

Appuntamento ad Hannover dal 14 al 18 marzo.



Concludiamo la panoramica sul software libero di modellazione ed editing tridimensionale con la creazione di modelli divisi in parti da assemblare. Seconda ed ultima puntata.

CONOSCIAMO OpenSCAD

a cura di L.I.F.O. (Laboratorio Informatico Free&Open)

OpenSCAD è un programma libero di modellazione 3D i cui punti di forza sono nella preparazione di modelli "tecnici" più che "artistici": oggetti che richiedono misure precise, geometrie ben definite e magari ottenibili per ripetizione di componenti simili.

Nella puntata precedente di questo tutorial abbiamo visto che in OpenSCAD si descrivono gli oggetti a partire da primitive di base come parallelepipedi, sfere e cilindri, mediante operazioni insiemistiche (unioni, intersezioni...) e trasformazioni geometriche. Limitandoci a primitive ed operazioni semplici, avevamo realizzato una scatola parametrica adatta a contenere dei circuiti elettronici. In questa seconda ed ultima puntata proviamo a costruire qualcosa di più complesso

come i meccanismi: alcuni ingranaggi ed una struttura per contenerli e supportarli.

Contrariamente al progetto presentato nel fascicolo precedente, questa volta non stamperemo la scatola tutta in un pezzo, bensì separando le varie facce, unendole poi ad incastro; questo esercizio ci serve principalmente per poter vedere diversi metodi di modellazione, ma ha anche il vantaggio di poter sfruttare per ogni parte l'orientamento migliore al fine della praticità di stampa e della solidità del pezzo stampato.

LIBRERIE

Innanzitutto creiamo un file `lib/sizes.scad` nel quale inseriremo le varie misure necessarie al progetto: in questo modo potremo dividerle

Listato 1 - lib/sizes.scad

```
// Gears settings
gears_circular_pitch = 200;
gears_bore_diameter = 3;
gears_hub_diameter = 7;
gears_thickness = 5;
gears_hub_thickness = 10;
gears_rim_thickness = 8;
gears_number_of_teeth_big = 39;
gears_number_of_teeth_small = 9;

// Axis settings
axis_cylinder_r = 4;
axis_cylinder_h = 10;
axis_cube_l = 5.6;
axis_cube_h = 10;

// Handle settings
handle_d = 40;
handle_r1 = 16;
handle_r2 = 10;
handle_h = 5;
handle_hole_h = 15;
tolerance = 0.5;
handle_knob_r = 6;
handle_knob_h = 20;

// Box sizes

gears_d = gears_circular_pitch * (gears_number_of_teeth_big + gears_number_of_teeth_small) / 360;
gears_h = gears_circular_pitch * gears_number_of_teeth_big / 180;
gears_len = gears_circular_pitch * gears_number_of_teeth_big / 180 + gears_d;
gears_width = gears_rim_thickness * 2;

box_wall = 2;
box_length = gears_len + 10;
box_width = gears_width + axis_cylinder_h - box_wall + 1;
box_height = gears_h + 10;
box_clear = 0.15;
```

tra i vari file del progetto, in modo da poterle così modificare facilmente in corso d'opera.

Il file corrispondente, presente nel **Listato 1**, è la versione finale, ma nel corso dell'articolo scriveremo man mano in questo file le misure di cui abbiamo bisogno.

Questo file verrà incluso in tutti i file successivi tramite l'istruzione `include <lib/sizes.scad>` (**Listato 2**) che equivale, sul piano logico, a copiare ed incollare i contenuti di `lib/sizes.scad` nel punto in cui si trova l'istruzione; in questo modo potremo sempre utilizzare le

variabili definite nella libreria `lib/sizes.scad`. Fate attenzione al fatto che l'istruzione `include` non va usata all'interno di moduli o di gruppi di istruzioni, dato che con ogni probabilità includere in quel punto i contenuti del file causerebbe degli errori di sintassi.

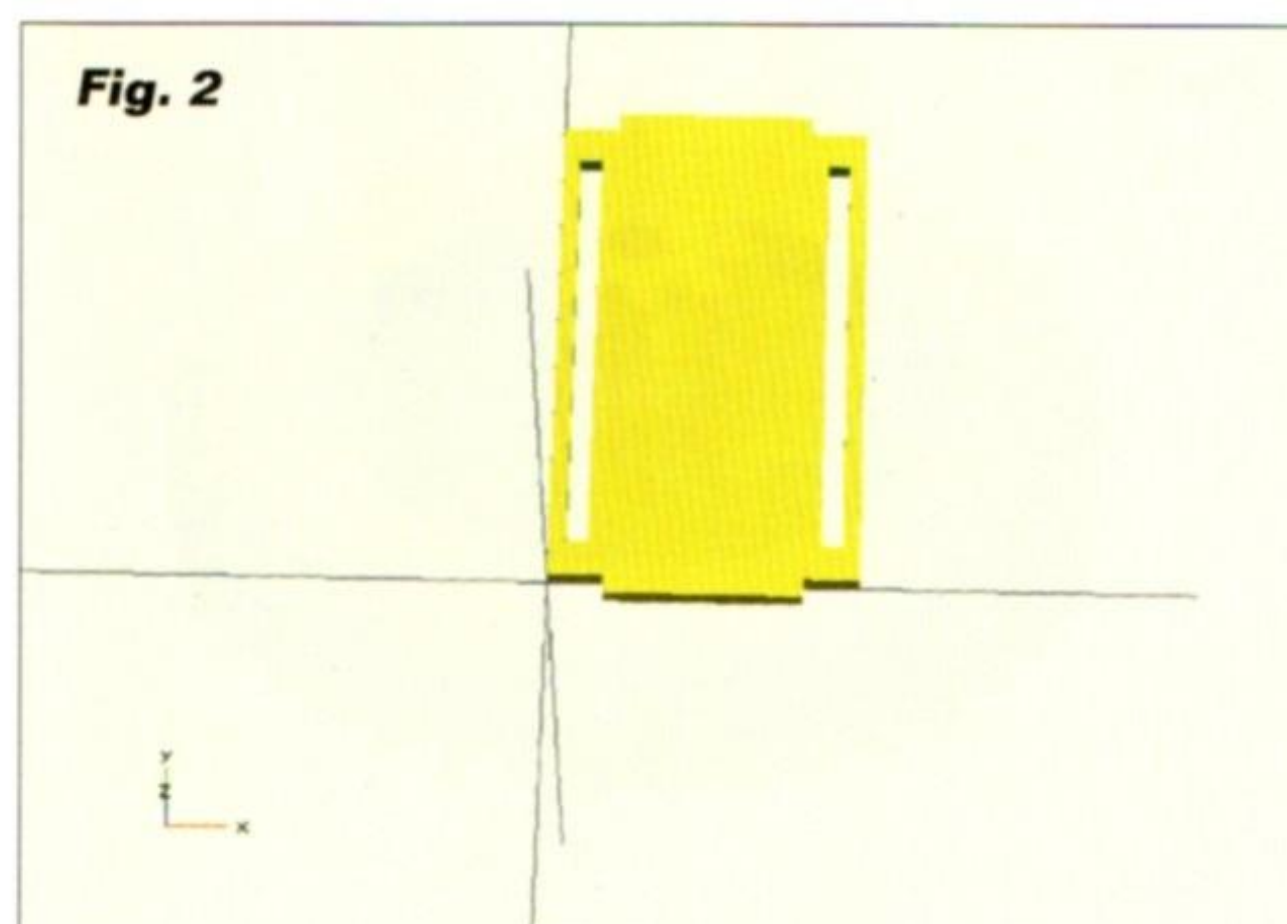
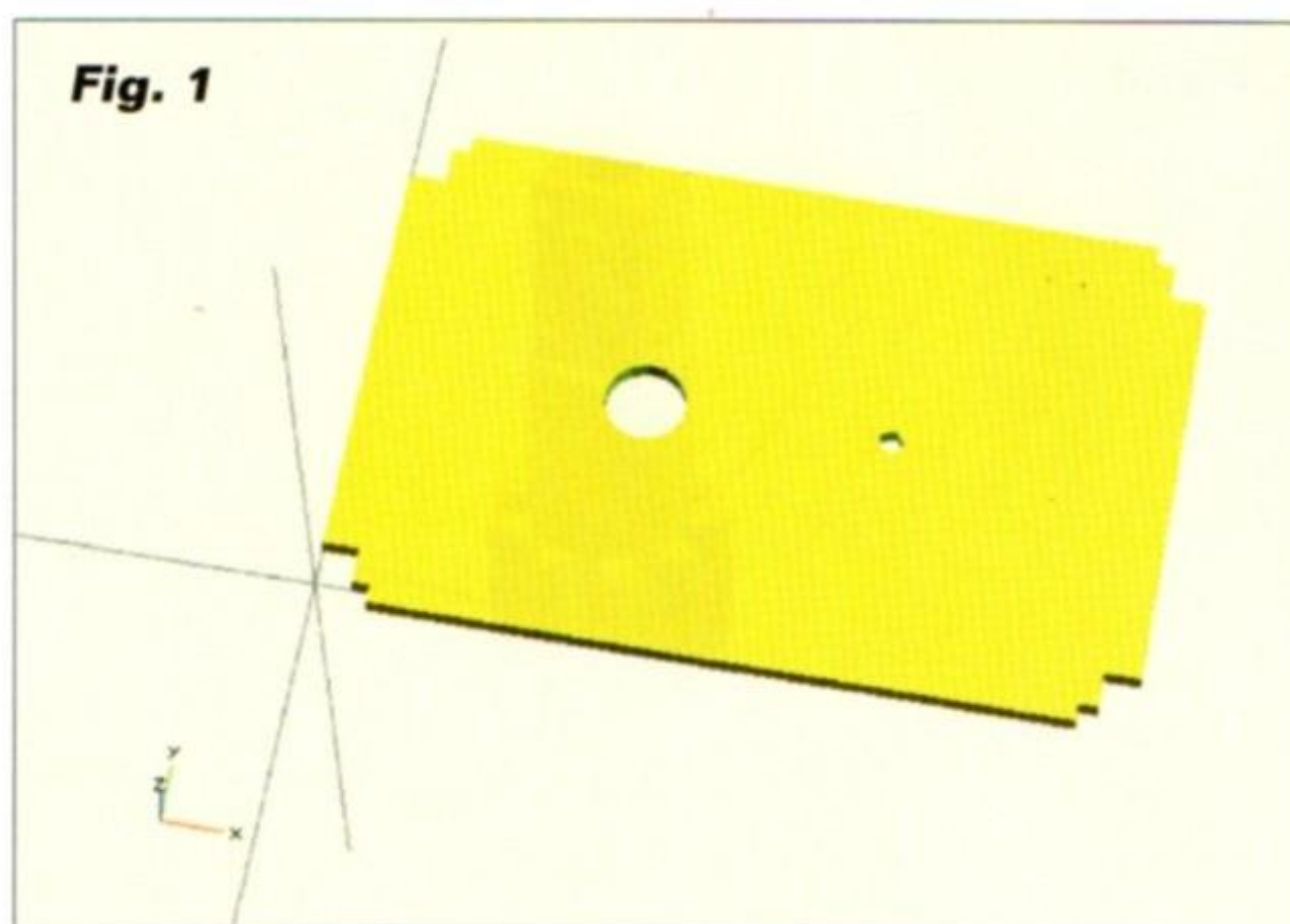
In generale, è utile usare un file da includere per suddividere un progetto complesso in parti più maneggevoli, ma soprattutto per condividere parti di codice comuni a più file, come le variabili di questo caso.

Col tempo, capiterà poi di creare moduli riuti-

Listato 2 - lib/axis.scad

```
include <lib/sizes.scad>

module axis() {
    difference() {
        union() {
            cylinder(r=axis_cylinder_r, h=axis_cylinder_h);
            translate([-axis_cube_l/2, -axis_cube_l/2, axis_cylinder_h - 0.01])
                cube([axis_cube_l, axis_cube_l, axis_cube_h]);
        }
        translate([0, 0, -1]) cylinder(r=gears_bore_diameter/2,
            h=axis_cylinder_h+axis_cube_h+2);
    }
}
```

lizzabili in più progetti: anche questi possono essere raccolti in un file da includere nei rispettivi progetti, creando così una vera e propria libreria, personale o da condividere in Internet. Le pareti principali della scatola, `long_side.scad` e `short_side.scad` sono disegnate semplicemente come unione di cubi, dai quali vengono rimossi degli altri cubi per formare gli incastri. Il valore `box_clear` serve per aggiungere una tolleranza (clearance) agli incastri, e ovviamente dipende dalle caratteristiche della stampante usata.

LIBRERIE DI TERZE PARTI

Oltre alla possibilità di crearsi una propria libreria di parti, OpenSCAD permette di integrare nei nostri progetti numerose librerie esistenti, scaricabili da Internet. Una di queste è MCAD, distribuita assieme ad OpenSCAD, che comprende componenti utili per la realizzazio-

ne di modelli meccanici; useremo una parte di tale libreria per realizzare una coppia di ruote dentate.

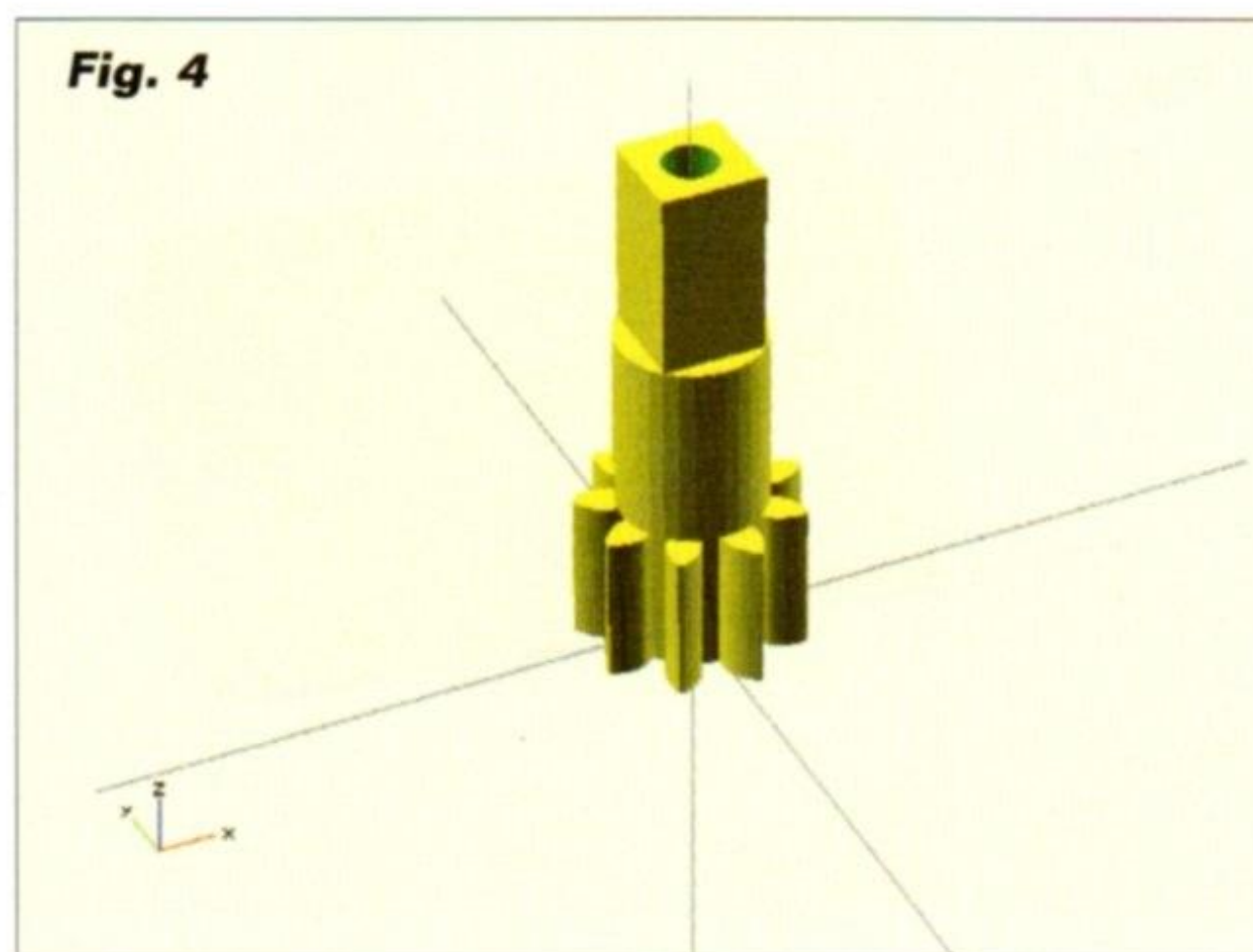
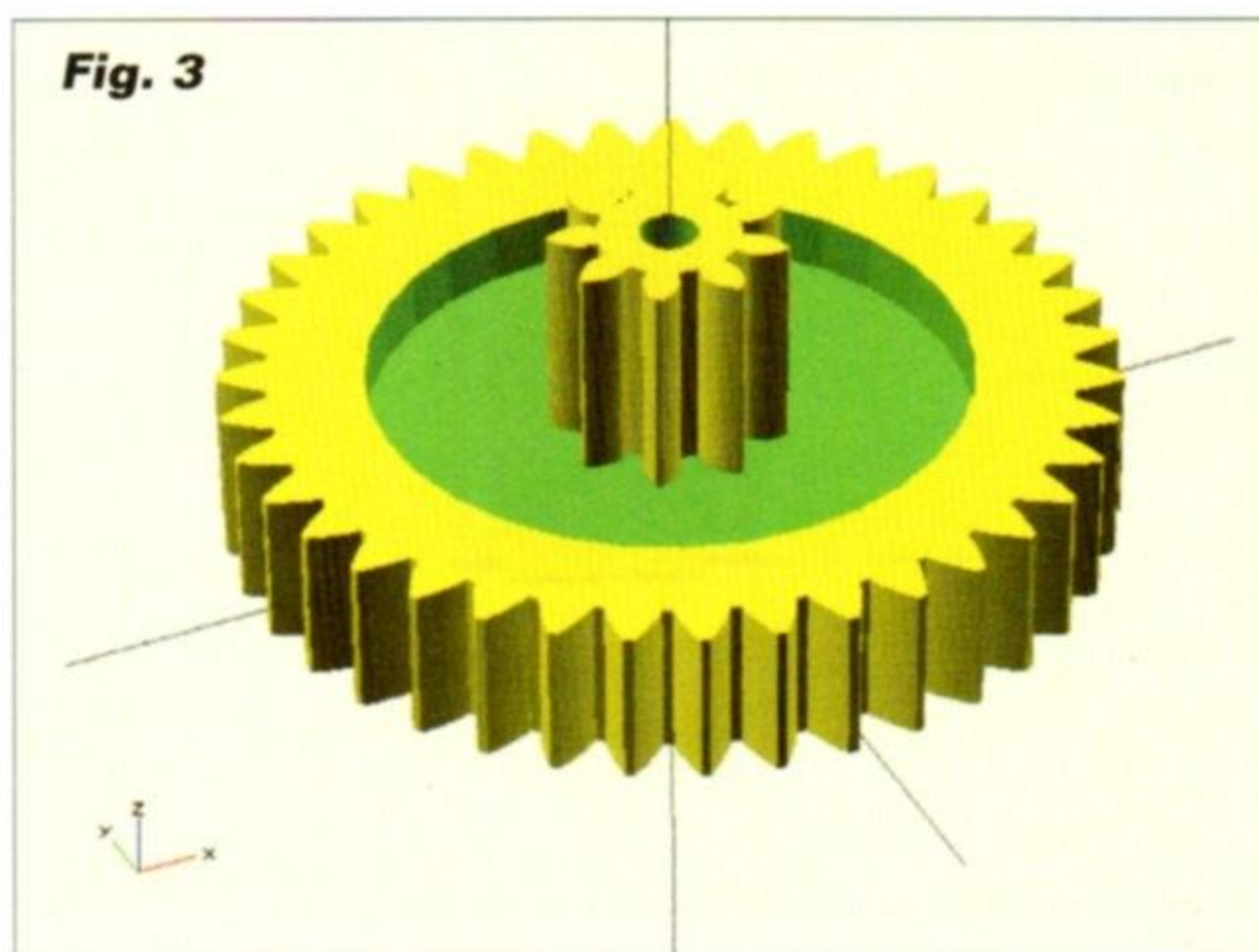
Per utilizzare MCAD nel nostro file `gear.scad` ci avvaliamo dell'istruzione `use <MCAD/involute_gears.scad>`. La differenza con `include` visto prima è che `use` si limita ad importare la definizione di moduli definiti nel file, ignorando eventuali istruzioni che prevedano di disegnare qualche oggetto (**Listato 3**). Questo permette agli autori di librerie, ad esempio, di avere un oggetto di test all'interno del file per verificarne il funzionamento, senza che questo venga incluso in tutti i progetti che richiamano la libreria. Possiamo notare che il comando `use <MCAD/involute_gears.scad>` funziona anche se la directory MCAD non è presente nel nostro progetto: quando si usano `include` ed `use`, OpenSCAD cerca i file richiamati in tre possibili locazioni, nell'ordine:

Listato 3 - gear.scad

```
use <MCAD/involute_gears.scad>
include <lib/sizes.scad>

$fn=32;

union() {
  gear(
    number_of_teeth=gears_number_of_teeth_big,
    circular_pitch = gears_circular_pitch,
    bore_diameter = gears_bore_diameter,
    hub_diameter = gears_hub_diameter,
    thickness = gears_thickness,
    hub_thickness = gears_hub_thickness,
    rim_thickness = gears_rim_thickness
  );
  translate([0, 0, gears_thickness - 0.01]) gear(
    number_of_teeth=gears_number_of_teeth_small,
    circular_pitch = gears_circular_pitch,
    bore_diameter = gears_bore_diameter,
    hub_diameter = gears_hub_diameter,
    thickness = gears_thickness,
    hub_thickness = gears_hub_thickness,
    rim_thickness = gears_rim_thickness * 2 - gears_thickness
  );
}
```

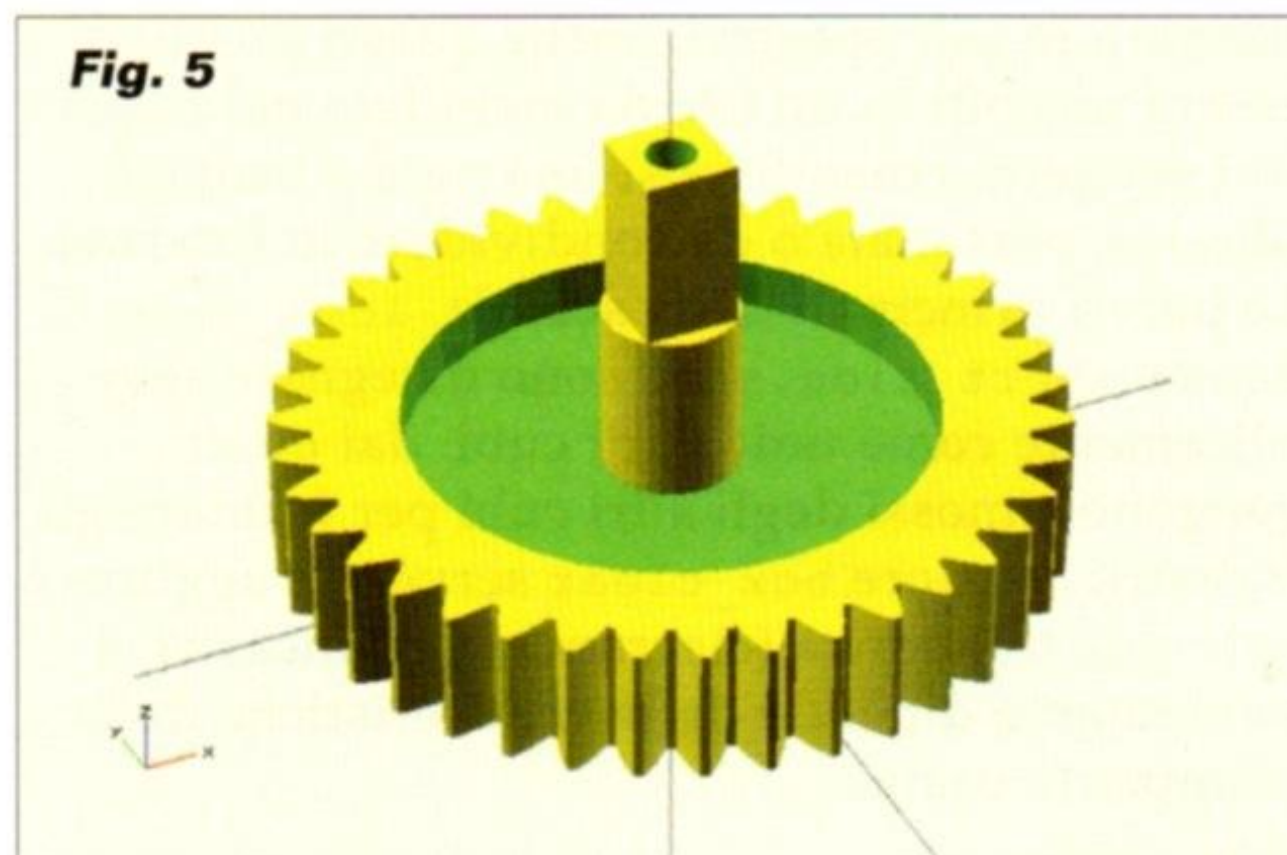



- nella directory del progetto;
- nell'installazione di OpenSCAD;
- nella directory delle librerie dell'utente, visibile dal menu File → Show Library Folder...

In questo caso MCAD si trova nell'installazione di OpenSCAD e può essere richiamata senza dover far nulla.

`involute_gears.scad` definisce un modulo `gear` con numerosi parametri che possiamo usare semplicemente per generare le componenti del nostro ingranaggio.

Per migliorare la precisione con cui OpenSCAD disegna i piccoli dettagli delle componenti dell'ingranaggio, useremo la variabile speciale `$fn=32`; ciò significa che gli archi verranno sempre disegnati in modo che un cerchio completo sia sempre tracciato in 32 segmenti. La parte centrale dell'ingranaggio, `gear.scad`, sarà composta da due ruote dentate con 39 e 9 denti rispettivamente, una sovrapposta all'altra. Come visto nella puntata precedente, il secondo ingranaggio è traslato di poco meno dello spessore del primo, in modo da



definire correttamente le intersezioni (Fig. 3). Purtroppo, a causa di un bug che la libreria `involute_gears.scad` presenta nel calcolo delle dimensioni degli ingranaggi generati e il parametro `circular_pitch` non corrisponde veramente alla distanza tra i due denti; il diametro dell'ingranaggio è calcolato erroneamente come numero dei denti ($\times \text{circular_pitch}/180$). In realtà l'effettiva distanza tra i denti si deve ottenere come $\text{circular_pitch} \times \pi/180$.

Listato 4 - `handle.scad`

```
include <lib/sizes.scad>

$fn=32;

hole_1 = axis_cube_1 + tolerance * 2;

difference() {
  union() {
    hull() {
      cylinder(r=handle_r1, h=handle_h);
      translate([handle_d, 0, 0]) cylinder(r=handle_r2, h=handle_h);
    }
    cylinder(r=hole_1, h=handle_hole_h);
  }
  translate([0, 0, handle_hole_h/2])
    cube([hole_1, hole_1, handle_hole_h + 2], center=true);
  translate([handle_d, 0, -1]) cylinder(r=1.5, h=handle_r1 + 2);
}
```


Listato 5 - handle_knob.scad

```
include <lib/sizes.scad>

$fn=32;

hole_1 = axis_cube_1 + tolerance * 2;

difference() {
    minkowski() {
        cylinder(r=handle_knob_r - 1, h=handle_knob_h - 1);
        difference() {
            sphere(r=1);
            translate([0, 0, -1.1]) cube(2.2, center=true);
        }
    }
    translate([0, 0, -1]) cylinder(r=gears_bore_diameter/2, h=handle_knob_h + 2);
}
```

MODULI

Le altre due parti di ingranaggio, `gear_small.scad` e `gear_big.scad` sono uguali ad una delle due ruote dell'ingranaggio centrale, ma al centro hanno un attacco quadrato al quale si può attaccare la manovella che disegneremo successivamente o ciò che vogliamo far muovere. Dato che l'attacco è uguale nei due modelli, possiamo definirlo in un file a parte da usare come nostra libreria personale: come nell'articolo precedente definiamo un modulo `axis()`, questa volta senza nessun parametro, e lo salviamo nel file `lib/axis.scad`.

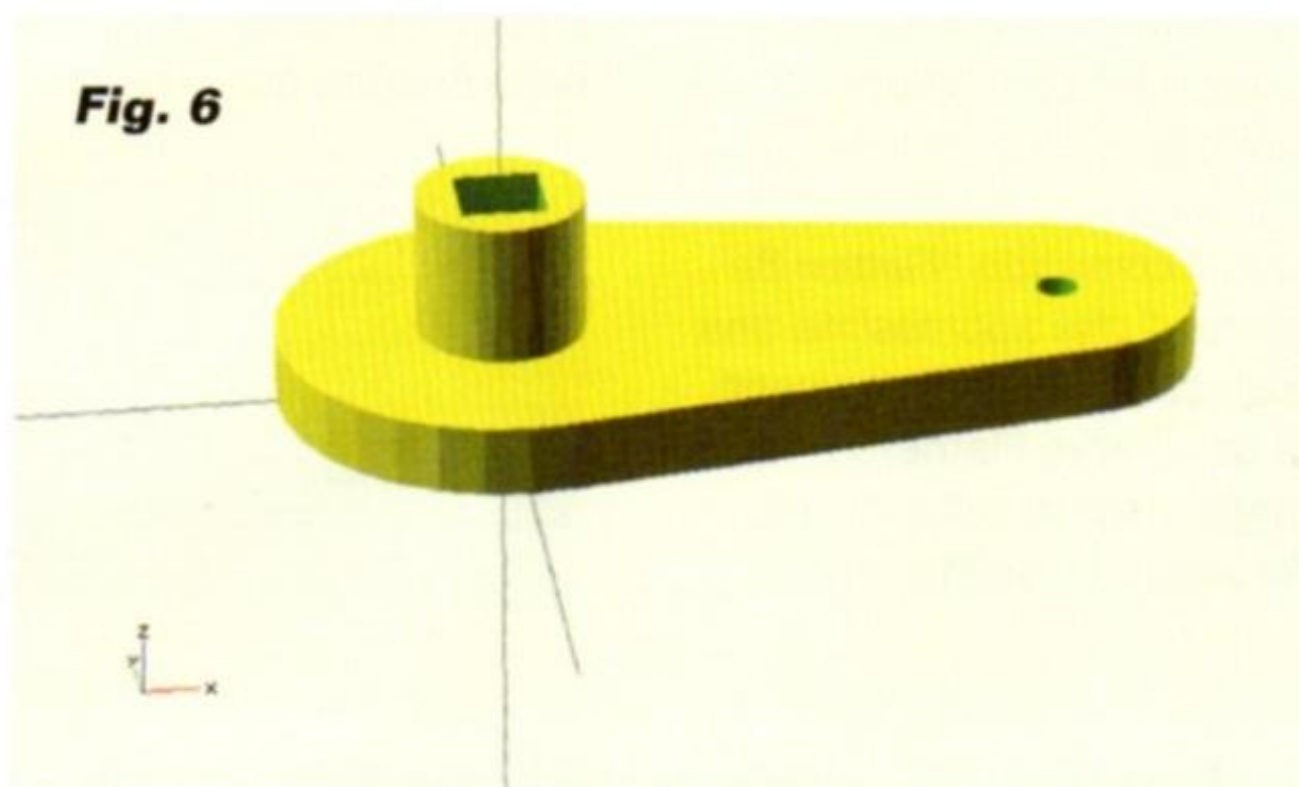


Fig. 6

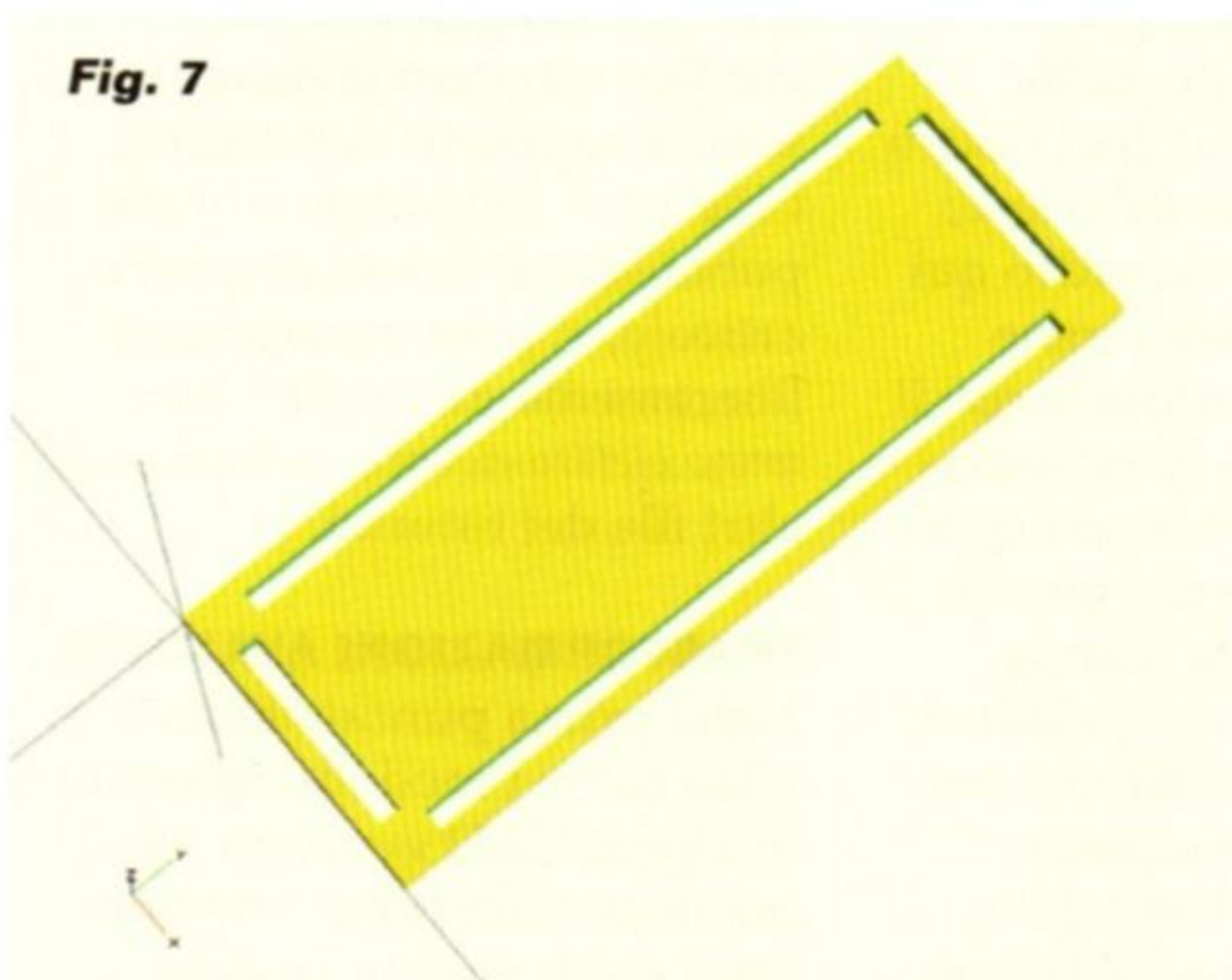


Fig. 7

TRASFORMAZIONI AVANZATE

Per realizzare la tipica forma da manovella, nel file `handle.scad` usiamo l'involucro convesso di due cilindri, spiegato nel seguito di questo articolo. Nelle misure abbiamo specificato anche una tolleranza differente da quella degli incastri della scatola, per potersi adattare meglio alle caratteristiche della stampante o al metodo con cui verrà realizzato il modello fisico, per permettere alla manovella di incastrarsi correttamente nell'attacco quadrato degli ingranaggi (vedere il **Listato 4**).

Per ottenere la manopola (pomello, **Listato 5**) avremmo potuto usare un semplice cilindro, ma ne abbiamo arrotondato l'estremità superiore tramite la somma di Minkowski con una semisfera, anche quella spiegata in dettaglio di seguito, in modo da renderlo più comodo da impugnare.

ESTRUSIONE DI DXF

La base (file `base.scad`) è costruita come le pareti, componendo dei cubi, ma per il lato superiore vogliamo aggiungere delle decorazioni (**Fig. 7**).

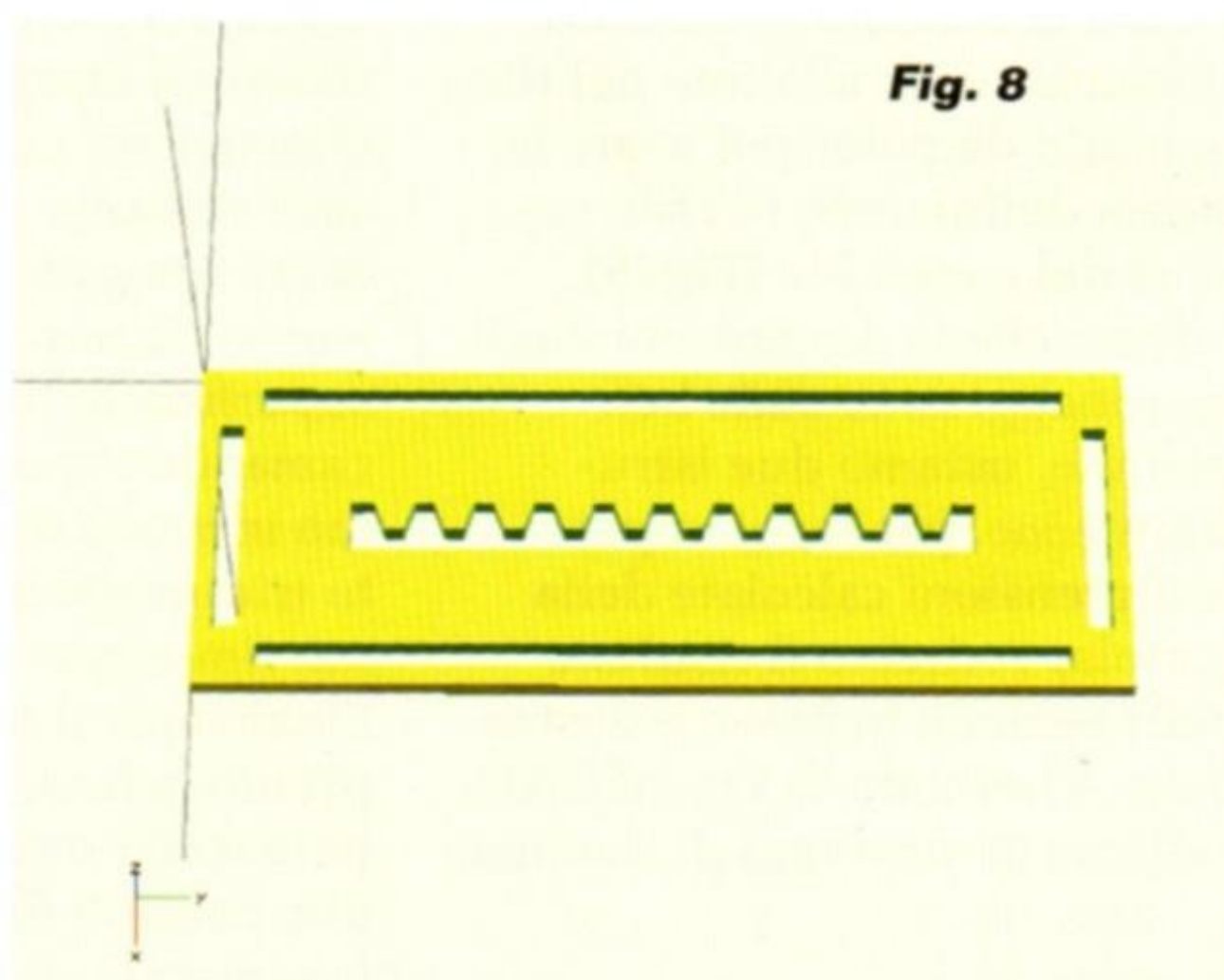
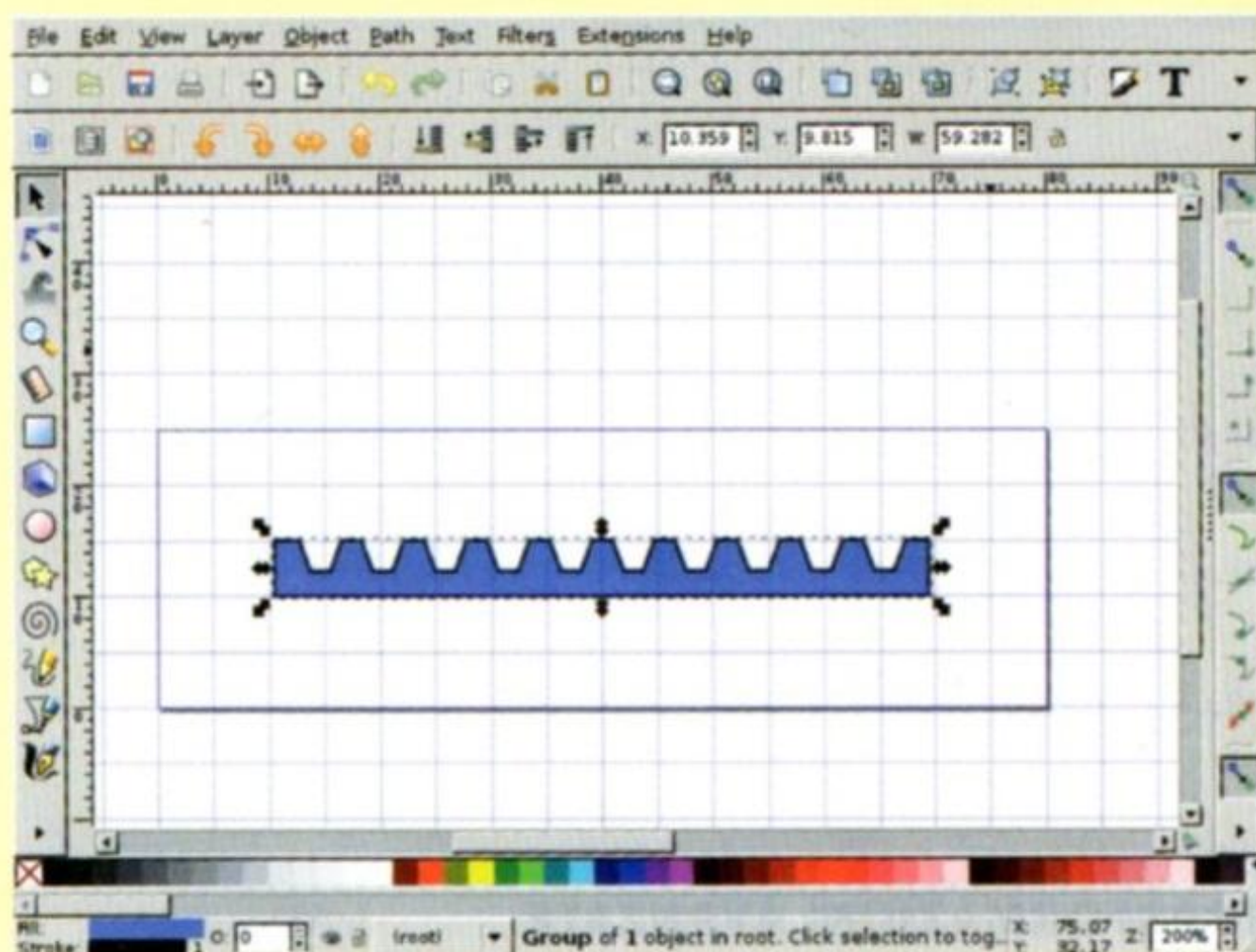


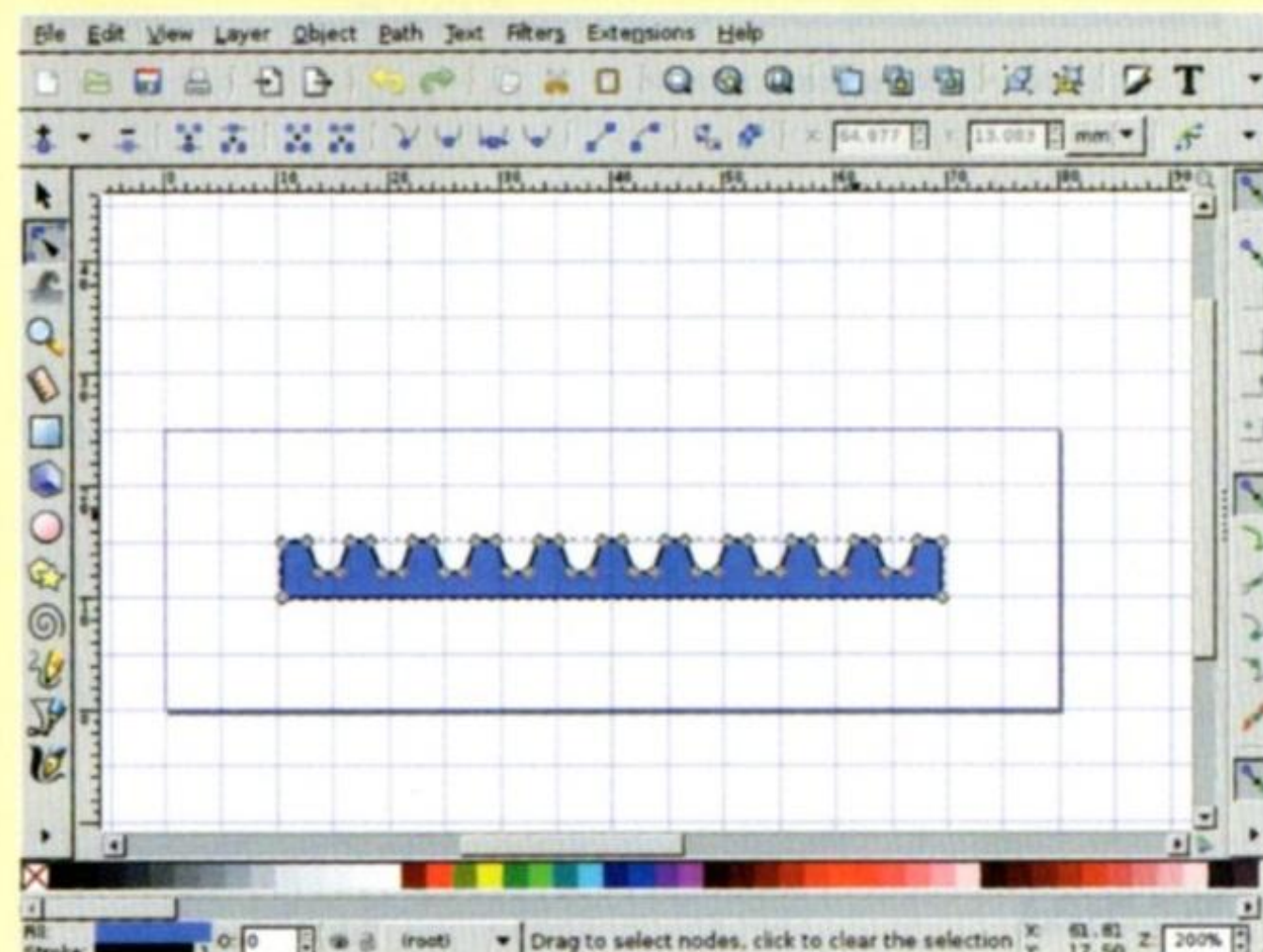
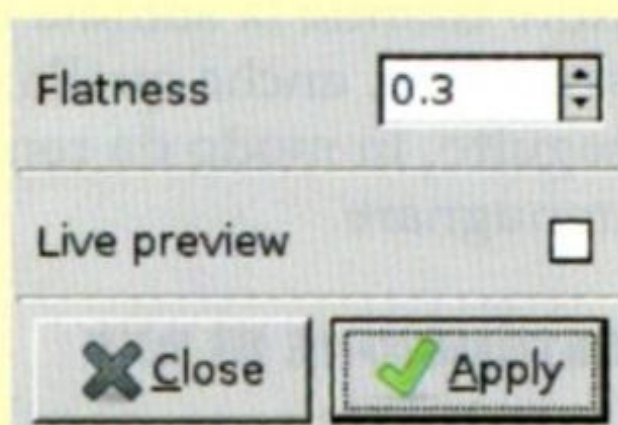
Fig. 8

Convertire file SVG in DXF con Inkscape



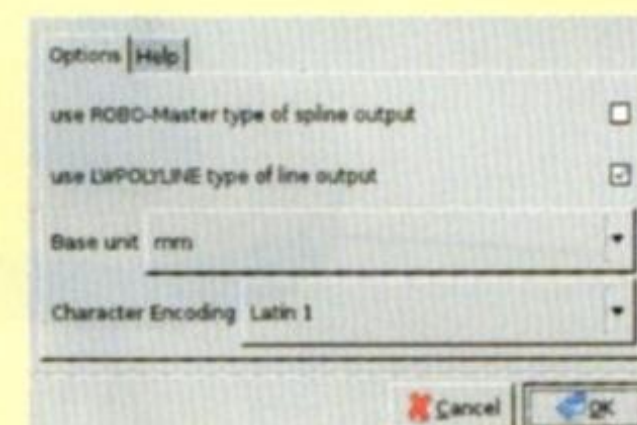
SVG è un formato molto diffuso per il salvataggio di immagini vettoriali, supportato dalla maggior parte dei programmi di disegno. Purtroppo, il supporto per SVG in OpenSCAD è ancora in lavorazione e non è disponibile nelle versioni stabili del programma, che sono in grado di importare forme 2D solo nel formato DXF. Per fortuna, il programma libero Inkscape, che lavora nativamente in SVG, è in grado di esportare file DXF che, se si è posta un po' di attenzione, possono essere letti da OpenSCAD. Innanzitutto, molte delle funzionalità di SVG non possono essere convertite:

ovviamente tutto ciò che riguarda i colori verrà ignorato da OpenSCAD, e non si può fare uso neanche di cose come lo spessore dei tratti né di figure formate da più componenti; ciò che si deve avere è un unico percorso (path) che descrive una superficie chiusa. Eventuali oggetti (circonferenze, quadrati, testo ecc.) andranno convertiti in percorsi, tramite gli strumenti del menu Path, quindi i vari



percorsi andranno combinati tramite unione, differenza eccetera in un unico percorso. Del disegno qui in alto, ad esempio, ciò che verrà esportato sarà la superficie in blu, non il contorno in nero; selezionare l'oggetto ed assegnare un colore di riempimento può essere un buon metodo per verificare ciò che si sta ottenendo. Un'altra funzionalità non supportata dall'import di DXF sono le linee curve: anche qui Inkscape ci viene in aiuto con il comando "Flatten Bezier..." che approssima una riga curva con dei segmenti. Il parametro Flatness andrà aggiustato di volta in volta in base al disegno; valori

più bassi daranno risultati più precisi, ma che comportano maggiore complessità dell'oggetto: in molti casi un valore attorno a 0,3 fornisce un buon compromesso. Finito di elaborare l'immagine, si può salvarla come DXF, avendo cura di selezionare il formato "Desktop Cutting Plotter (AutoCAD DXF R14)" (non "AutoCAD DXF R12") e quindi le opzioni indicate nelle finestre qui in basso.



Allo scopo, innanzitutto definiamo la base come modulo, richiamandolo alla fine del file, in modo da poter poi usare la stessa definizione nel file `top.scad` del coperchio (Fig. 8). Adesso che le dimensioni degli ingranaggi sono state ben definite, usiamo due istruzioni `echo` per far stampare le dimensioni calcolate della scatola; vedremo il risultato nella sezione in basso a destra della schermata di OpenSCAD, assieme ai messaggi di sistema:

```
ECHO: 80
ECHO: 25
```

Con un programma di grafica vettoriale o un CAD, prepariamo un disegno di quelle dimensioni, `top.dxf` (nel nostro caso abbiamo usato Inkscape, come spiegato nel riquadro qui sopra), lo importiamo come superficie bidimensionale con il comando `import` e lo estrudiamo in modo da avere un oggetto tridimensionale che possiamo rimuovere dalla scatola. Finalmente il nostro modello è pronto: adesso possiamo stampare tutti i pezzi, preparare due pezzi di fil di ferro della lunghezza opportuna da usare

come assi ed assemblare il nostro ingranaggio, scegliendo da che lato montare la maniglia in base al rapporto desiderato. I file `.scad`, di cui non trovate pubblicato il listato in questo articolo, si possono scaricare liberamente dal nostro sito www.elettronica.in insieme agli altri file del tutorial.

TRASFORMAZIONI AVANZATE

Nella prima puntata abbiamo visto come i modelli elementari di OpenSCAD vengano manipolati tramite trasformazioni con i quali possono essere

traslati, ruotati o cambiati di dimensioni. OpenSCAD offre anche delle trasformazioni più complesse, tra cui "minkowski" ed "hull".

Usando queste trasforma-

zioni e lavorando su modelli complessi, specie se si usano computer non troppo recenti, può capitare che OpenSCAD non riesca a portare a termine la loro compilazione, ma che

l'interfaccia smetta di rispondere; in questi casi, basta lanciare la compilazione da riga di comando.

SOMMA DI MINKOWSKI

In matematica, la somma di Minkowski di due insiemi $X \subset \mathbb{R}^n$ ed $Y \subset \mathbb{R}^n$ è definita come l'insieme $\{x+y: x \in X, y \in Y\}$, ovvero di ogni coppia di punti presi dai due insiemi di partenza si fa la somma come vettore e il risultato che ne deriva è un punto dell'insieme risultante. Se uno dei due insiemi è centrato nell'origine, come si fa nella maggior parte dei casi, si può visualizzare il risultato finale come l'applicazione di questo solido a tutti i punti dell'altro solido, "base".

In particolare, si può calcolare l'offset di un oggetto, ovvero un oggetto "allargato" di una certa quantità rispetto all'originale; per allargare in tutte e tre le dimensioni si dovrà fare una somma di Minkowski con una piccola sfera (o per semplificare i calcoli un piccolo cubo), se invece si vuole l'offset di una superficie bidimensionale basterà sommarla con un piccolo cerchio.

Ad esempio, nella Fig. 9 abbiamo importato una stella bidimensionale da un file .dxf, ne abbiamo calcolato l'estruzione sommandole un cerchio e quindi abbiamo tolto la stella originale (visualizzata in rosso tramite il modificatore #) per poi fare l'estruzione del solo contorno.

```
linear_extrude(height=1) difference() {
    minkowski() {
        import("star.dxf");
        circle(r=2);
    }
    # import("star.dxf");
}
```

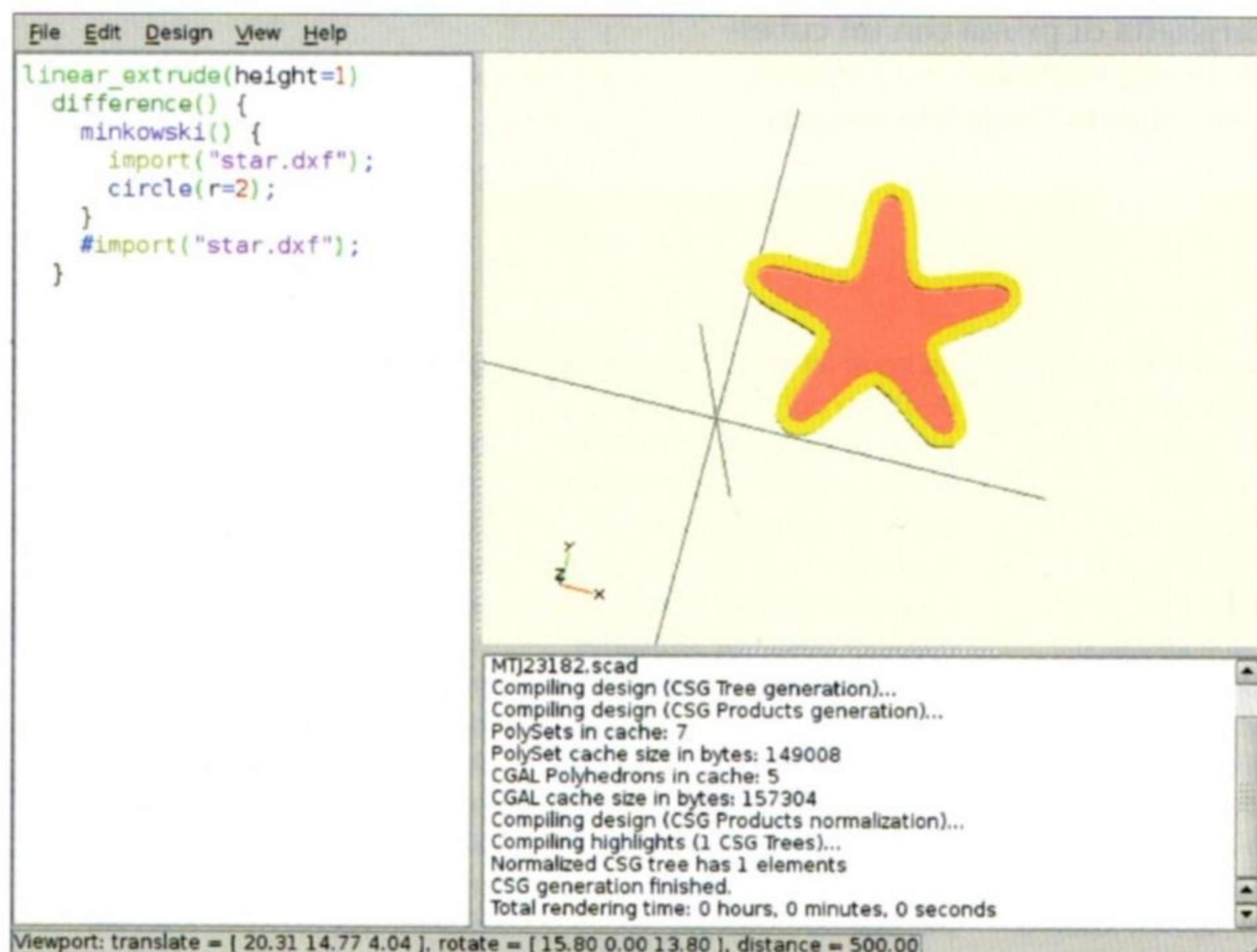


Fig. 9

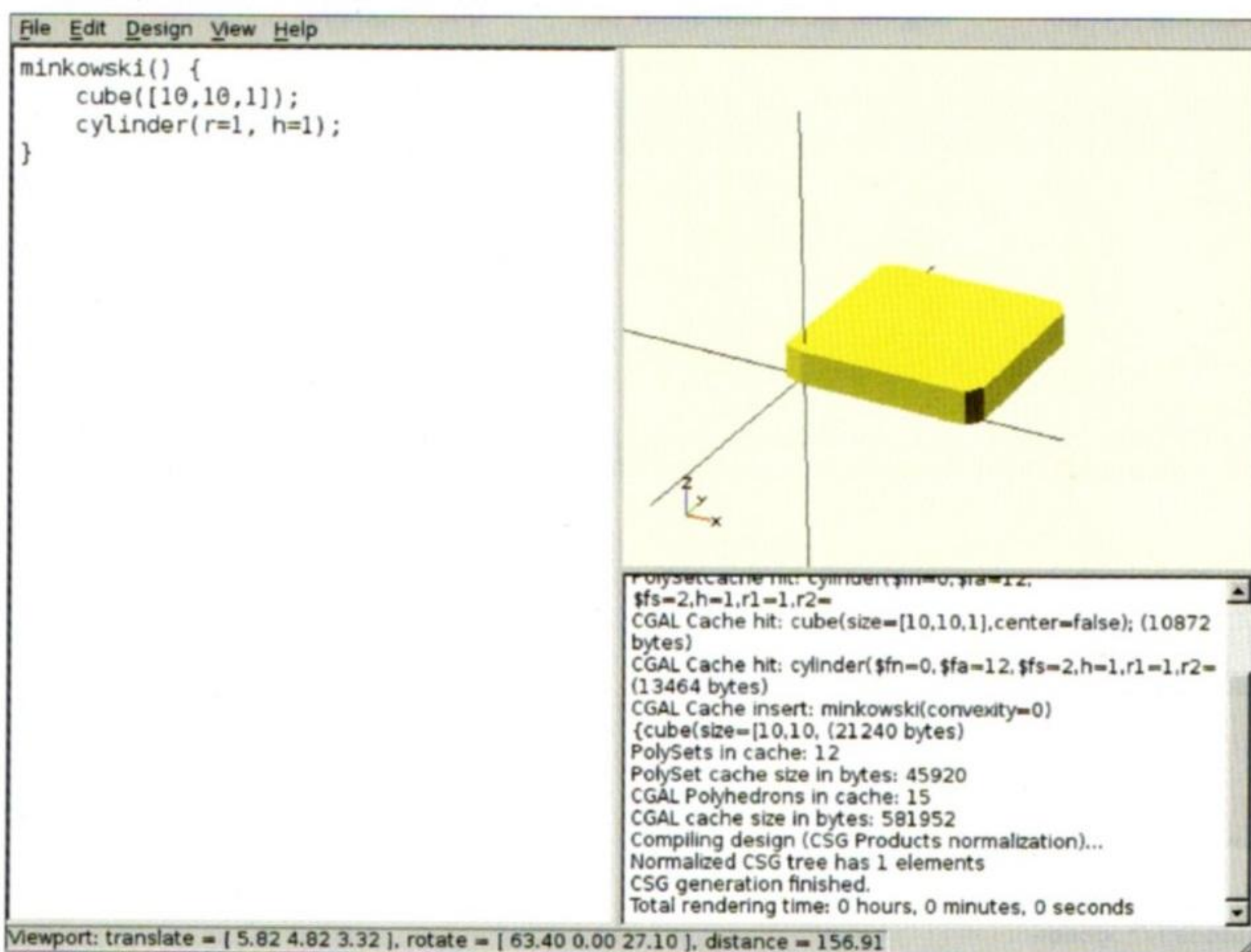


Fig. 10

Un'altra applicazione della somma di Minkowski è per ottenere forme con i bordi arrotondati; ad esempio, facendo la somma di un parallelepipedo ed un cilindro possiamo ottenere una targhetta con gli angoli arrotondati (Fig. 10):

```
minkowski() {
    cube([10, 10, 1]);
```

```
cylinder(r=1, h=1);
}
```

Una cosa cui prestare attenzione è che tale trasformazione ingrandisce l'oggetto originale: ad esempio, confrontando la targhetta di prima con un cubetto 1x1x2, vediamo che l'altezza dell'oggetto finale è la somma

delle due altezze (dato che gli oggetti sommati erano piatti), e che anche le altre dimensioni sono cresciute (Fig. 11):

```
minkowski() {
    cube([10, 10, 1]);
    cylinder(r=1, h=1);
}
# translate([-2, 0, 0]) cube([1,
1, 2]);
```

Makefile

Make è un programma storico usato nei sistemi Unix per automatizzare la compilazione dei programmi, in particolare per quelli scritti in C, ed è disponibile o almeno installabile su tutte le distribuzioni Linux.

Il funzionamento di make viene configurato tramite il file Makefile, in un linguaggio che richiederebbe troppo spazio per essere spiegato, ma l'esempio presente nei listati è sufficientemente generico da poter essere copiato ed inserito in qualunque progetto openscad, avendo

solo l'accortezza di controllare che le righe intente inizino con un carattere tab e non con degli spazi.

Un grosso vantaggio di make è che permette di specificare il nome del file che si vuole ottenere e gestisce una catena di dipendenze che gli permette di effettuare i soli passi di compilazione necessari ad ottenere quel file, evitando anche di rifare il lavoro già fatto nel caso in cui alcuni dei file non fossero stati modificati. Il file allegato può essere configurato con la posizione dei file .scad

sorgenti (di default la stessa directory del file Makefile) e quella dove si vuole che vengano creati i file destinazione (di default la sottodirectory build). Non vengono considerate le sottodirectory: è quindi possibile sfruttarle per file come nel nostro esempio lib/sizes.scad che vengono inclusi negli altri file, ma non corrispondono ad un modello. A quel punto è possibile lanciare make per ottenere tutti gli .stl corrispondenti agli .scad presenti nella directory di origine. Se si usa slic3r per

generare i file Gcode da inviare alla stampante, è anche possibile salvare la configurazione di slic3r che si vuole usare come config.ini nella stessa directory del Makefile e lanciare il comando make gcode per ottenere direttamente tutti i file Gcode pronti per la stampa.

Altri comandi disponibili sono make build/nomemodello.stl e make build/nomemodello.gcode dove nomemodello.scad è un file presente nella directory di origine, per compilare solo quel modello.

Makefile

```
BUILDDIR = build
CONFDIR = .
SRCDIR = .

SLIC3R = slic3r

VPATH = $(SRCDIR):$(BUILDDIR)

STL_TARGETS = $(patsubst $(SRCDIR)/%.scad,$(BUILDDIR)/%.stl,$(wildcard $(SRCDIR)/*.scad))
GCODE_TARGETS = $(patsubst $(SRCDIR)/%.scad,$(BUILDDIR)/%.gcode,$(wildcard $(SRCDIR)/*.scad))

.PHONY: all gcode clean
all: $(STL_TARGETS)

gcode: $(GCODE_TARGETS)

$(BUILDDIR)/%.stl: %.scad $(SRCDIR)/lib/*.scad
    mkdir -p $(BUILDDIR)
    openscad -o $@ $<

$(BUILDDIR)/%.gcode: %.stl
    $(SLIC3R) --load $(CONFDIR)/config.ini $^

clean:
    rm -f $(BUILDDIR)/*.stl $(BUILDDIR)/*.gcode
```


INVILUPPO CONVESSO

L'involuppo convesso, in inglese convex hull di uno o più solidi è definito come il solido convesso più piccolo che li contenga tutti.

Questa trasformazione è resa disponibile in OpenSCAD con il comando `hull` e la si può usare, ad esempio, per descrivere forme un po' più complesse delle solite in base a ciò che vogliamo contengano. Nella Fig. 12 abbiamo fatto un esagono con i bordi arrotondati, specificando dei cilindri (evidenziati in rosso) che ne descrivono gli angoli e chiedendo ad OpenSCAD di calcolare l'"interno". Il codice è il seguente:

```
hull() {
  for (i=[0:5]) {
    rotate([0, 0, i*60]) translate([10, 0, 0]) cylinder(r=4, h=3);
  }
}
```

```
for (i=[0:5]) {
  rotate([0, 0, i*60]) translate([10, 0, 0]) #cylinder(r=4.01, h=3.01);
}
```

Un'altra applicazione pratica di questa trasformazione può essere costruire una base per elementi disposti in posizioni note, ma non facilmente comprese in uno dei solidi elementari, magari aggiungendo poi un offset con `minkowski`.

Ad esempio, nella Fig. 13 si vede una base con buchi per fissaggio in un elenco di posizioni decisamente arbitrarie, così definita:

```
positions = [
  [40, 2, 0],
  [-10, 7, 0],
  [0, 30, 0],
  [20, 10, 0],
  [30, 27, 0]
];

difference() {
  linear_extrude(height=2)
```

```
minkowski() {
  circle(r=3);
  hull() {
    for (i=positions) {
      translate(i)
    }
  }
  for (i=positions) {
    translate(i) cylinder(h=5, center=true);
  }
}
```

USARE OPENS CAD DA LINEA DI COMANDO

Oltre che con l'interfaccia grafica, OpenSCAD può essere usato da linea di comando; oltre ad essere utile nel caso in cui i modelli siano troppo complessi e causino difficoltà all'interfaccia grafica, la linea di comando è un'opzione pratica per automatizzare la compilazione di più modelli (o magari varianti dello stesso), o per delegare la compilazione a computer più potenti di quello sul quale si sta lavorando. Per non far partire l'interfaccia grafica ma far funzionare OpenSCAD in modalità batch, è sufficiente usare l'opzione `-o`:

```
openscad -o destinazione.stl sorgente.scad
```

In questo modo OpenSCAD compilerà il file `sorgente.scad` senza lanciare l'interfaccia grafica e salverà il risultato in `destinazione.stl`, scegliendo il formato in base all'estensione del file.

Un'opzione particolarmente utile per la compilazione di modelli parametrici è `-D`, con la quale si possono sovrascrivere i valori delle costanti usate nel file; ad esempio, se si ha un file che usa le variabili `x` e `y` per descrivere larghezza e lun-

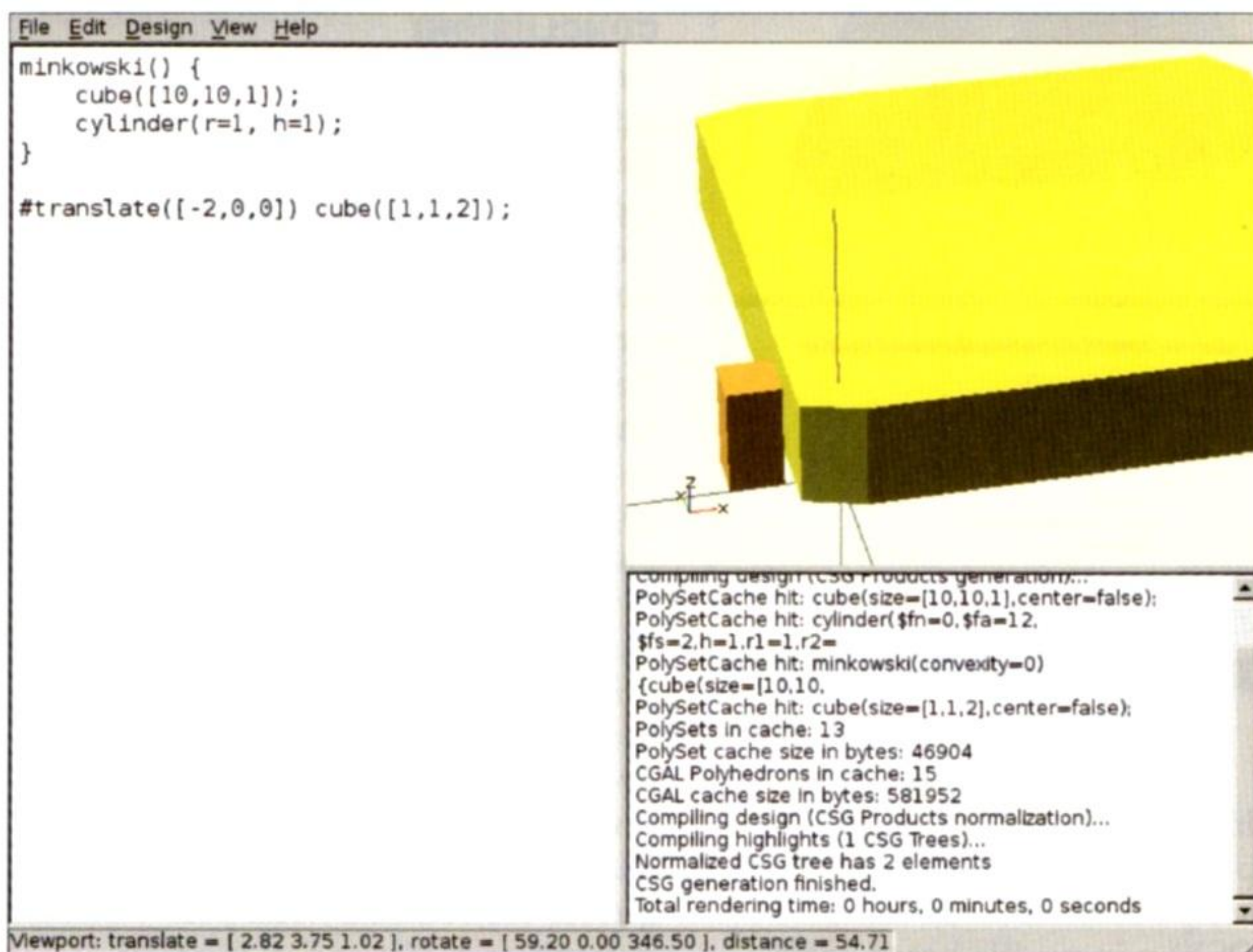


Fig. 11

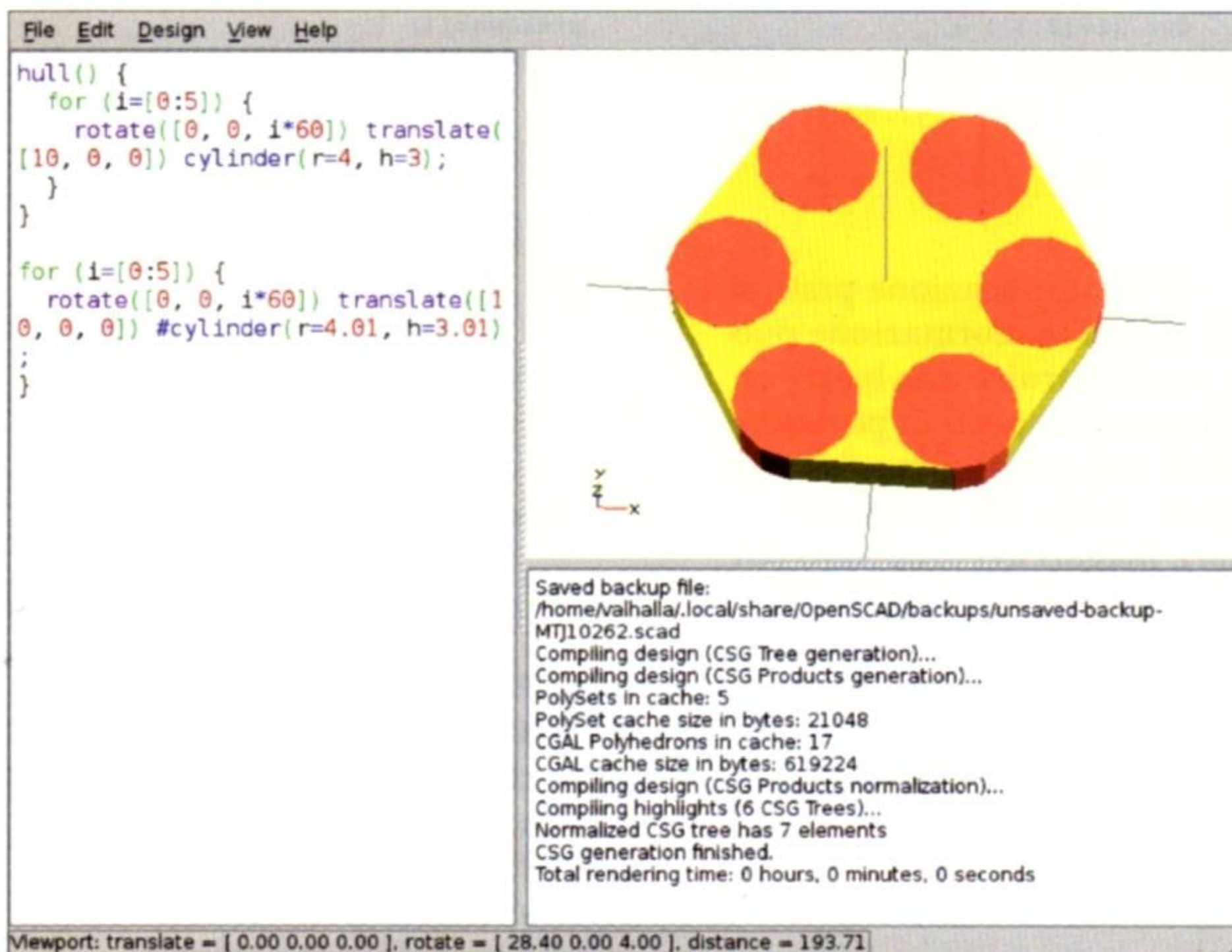


Fig. 12

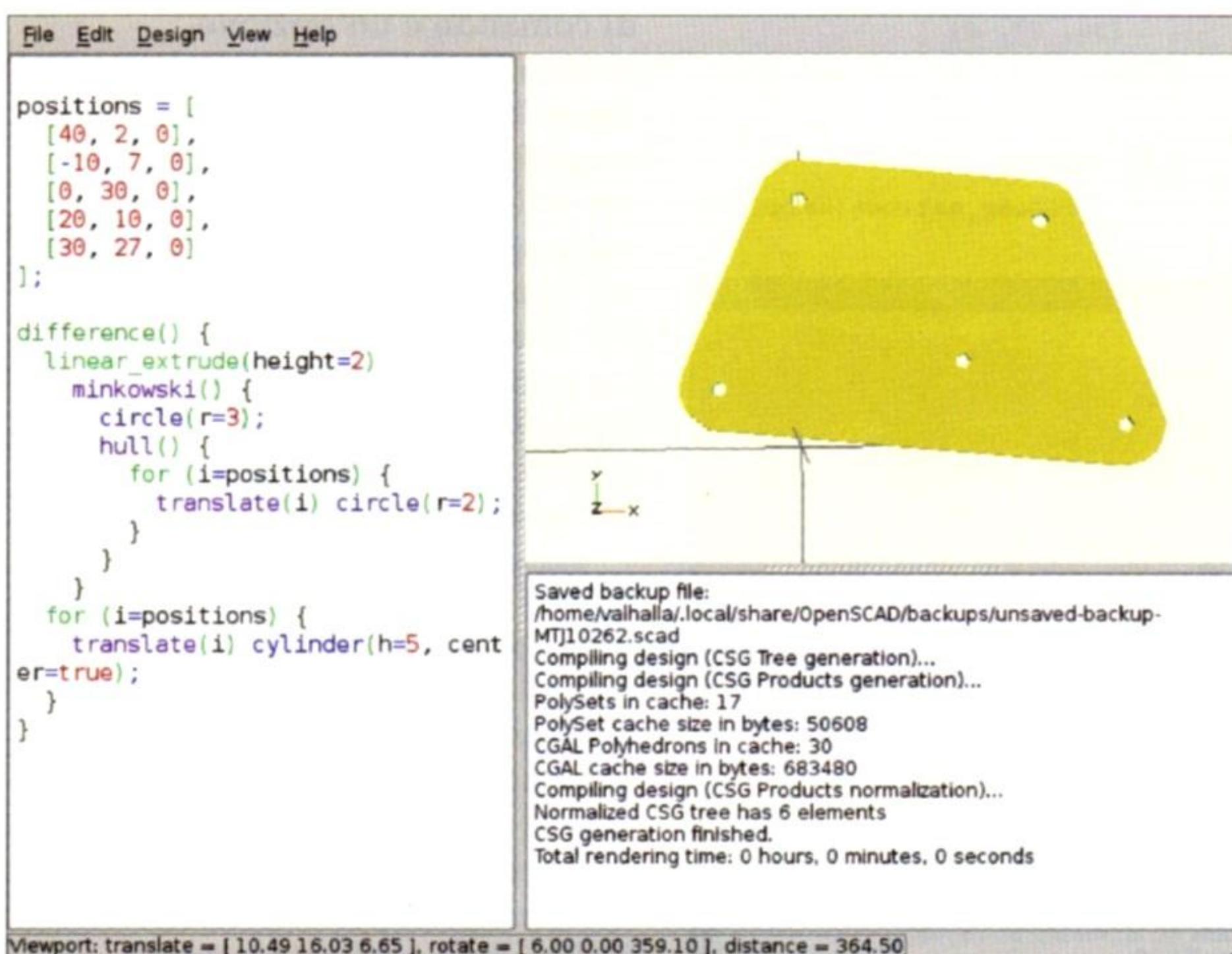


Fig. 13

ghezza di un pezzo lo si può compilare specificando queste misure con:

```

openscad -o destinazione.stl -D
x=60 -D y=20 sorgente.scad

```

In particolare, questa opzione può essere usata all'interno di script di shell (sotto Linux o OSX) o in un file batch sotto Windows per generare più

varianti dello stesso pezzo; ad esempio con uno script di shell:

```

for x in 20 40 60; do
  for y in 30, 50; do
    openscad -o destina-
zione-${x}x${y}.stl -D x=$x -D
y=$y sorgente.scad
  done
done

```

si genereranno i sei file desti-
nazione-20x20.stl, desti-
nazione-20x30.stl ecc. con
il nostro modello nelle varie
misure desiderate. Il valore
specificato con -D può com-
prendere espressioni Open-
SCAD arbitrarie; se si vogliono
usare delle stringhe bisogna
fare attenzione a racchiuderle
tra virgolette e far sì che le
virgolette non vengano inter-
cettate dalla shell. Ad esempio,
se si ha un file OpenSCAD
che importa il contenuto della
variabile file si può usare:

```

openscad -o dest.stl -D
'file="nomefile.dxf"' sorgente.
scad

```

CONCLUSIONI

Nella prima puntata di questo tutorial abbiamo spiegato l'uti-
lizzo dei principali comandi di
OpenSCAD, partendo dal fun-
zionamento del programma per
poi passare al linguaggio con
sintassi, componenti solide e
trasformazioni principali.
Qui abbiamo invece affronta-
to un modello più comples-
so, composto da numerose
parti che ci hanno permesso
di vedere la creazione e l'uso
di librerie, le trasformazioni
più complesse e soprattutto
indicazioni su come gestire un
progetto complesso, diviso in
più parti.

Siamo ora pronti per applicare
queste conoscenze nella pro-
gettazione e disegno di modelli
per le più svariate esigenze. ■



SENSORE DI GAS

Aggiungiamo, con questa piccola breakout board, un efficiente rivelatore di gas a sistemi di allarme generici, tecnologici e antifurto...

di BORIS LANDONI

Poter rilevare i gas presenti nell'aria intorno a noi e la loro concentrazione è spesso utile e talvolta indispensabile...vitale! Pensiamo ad esempio ai locali caldaia e alle cucine, dove si utilizza gas come metano, butano o propano (gas di bombola), GPL o gas di città (dove esiste ancora); in casi del genere, come anche in alcune officine, nelle fabbriche dove si trattano solventi o nei laboratori chimici, è fondamentale rilevare fughe di gas combustibili ed esplosivi, solventi, alcol e simili.

Come pure è importante rilevare la presenza di monossido di carbonio frutto della combustione degli idrocarburi, pericolosissimo per la salute umana perché quando inalato, si lega all'ossigeno trasportato dall'emoglobina impedendone il rilascio nell'organismo e portando a gravissime intossicazioni e -nei casi più gravi- alla morte.

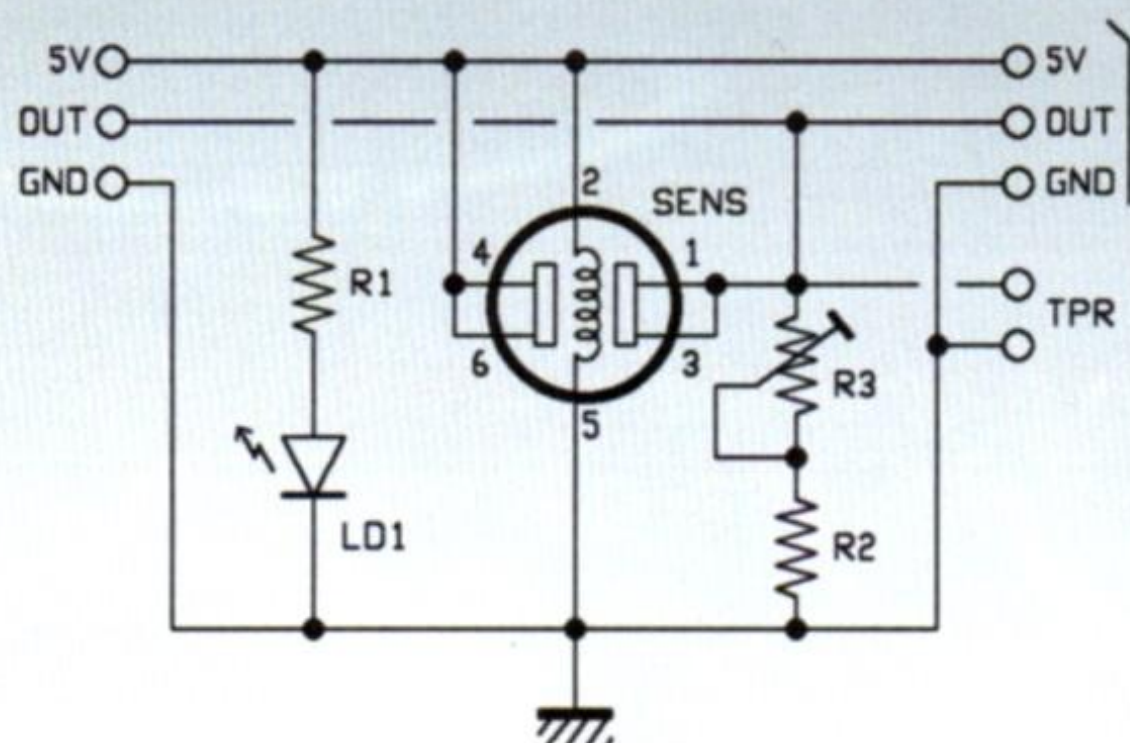
Il rilevamento dei gas meno nocivi è comunque utile a monitorare la situazione ambientale e quella dei luoghi di lavoro, per verificarne la salubrità o stilare statistiche e grafici che conservano lo storico dei valori assunti dagli inquinanti come l'anidride

carbonica o il fumo di sigaretta. Per questa ragione abbiamo deciso di sviluppare una scheda universale sulla quale montare un sensore di gas o fumo dalla caratteristica configurazione a elemento resistivo riscaldato e riscaldatore, quindi tipicamente a quattro o sei terminali (sempre quattro utili) per applicarlo a dispositivi esistenti o interfacciarlo a microcontrollori (con Arduino "va a nozze") o semplicemente indicatori a lancetta.

SCHEMA ELETTRICO

Il circuito è estremamente semplice e quindi compatto e facilmente inseribile in qualsiasi sistema: lo schema mostra l'essenzialità, dato che abbiamo il sensore, un LED (con la sua resistenza serie di limitazione della corrente) per indicare la presenza dell'alimentazione (che è singola a 5 volt in continua) e il trimmer R3, che è stato previsto perché alcuni sensori, di quelli utilizzabili, sono in grado di rilevare vari gas, quindi a seconda della tipologia di gas rilevabile è necessario calibrare la resistenza interna, mediante il trimmer, per ottenere un dato veritiero. Per i sensori dove non

[schema **ELETTRICO**]



è necessaria alcuna calibrazione, si può portare il cursore del trimmer tutto verso la linea OUT in modo da ottenere una resistenza di 0 ohm. Sui terminali TPR (Test Point Resistor) è possibile controllare il valore di resistenza impostato, mediante un tester.

Per capire a cosa serve il trimmer bisogna spiegare come funzionano i sensori utilizzabili in questa scheda, che sono tutti ad elemento sensibile riscaldato: si tratta di norma di un ossido metallico (titanio, stagno ecc.) che quando viene scaldato diventa maggiormente permeabile ai gas presenti nell'aria, i quali una volta penetrati alterano la resistenza elettrica che la membrana avrebbe in condizioni di riposo, vale a dire quando non è "contaminata" dai gas stessi. Nei sensori che utilizziamo, che sono poi i più comuni e usati da decenni, ci sono quattro elettrodi utili, che sono due per il filamento riscaldatore (che di norma è in lega nichel-cromo) e altrettanti per il sensore a membrana; sovente troviamo due coppie di terminali in parallelo per il riscaldatore, dato che esso può assorbire una corrente di valore discreto. Affinché rilevi correttamente secondo le curve indicate, ciascun sensore dev'essere a una determinata temperatura, quindi prima di prenderne per buoni i valori di resistenza forniti (e di riflesso, le percentuali di gas rilevato) bisogna che "vada in temperatura"; il

costruttore di norma indica, per una certa tensione e corrente di alimentazione del riscaldatore, il tempo richiesto a raggiungere la temperatura di regime, eventualmente anche in funzione della temperatura ambiente, dato che sicuramente all'aperto, in estate, l'elemento sensibile si scalda prima che in inverno, almeno nei climi come il nostro.

Siccome l'elemento sensibile è una resistenza variabile, per rilevare la concentrazione di gas, strettamente correlata a quella di resistività, basta disporre l'elemento in un partitore resistivo o, se vogliamo maggiore precisione o un'eventuale compensazione in temperatura, in un ponte di Wheatstone; nella nostra scheda ci accontentiamo della prima soluzione, che fornisce una discreta precisione: l'elemento sensibile ai gas fa partitore con la serie R2/R3. Essendo R3 un trimmer, possiamo regolare la resistenza serie con precisione per adattarla alla sensibilità ai vari gas dei sensori che possono rilevare più gas; ricordiamo che portando il cursore del trimmer verso la linea OUT la resistenza serie si abbassa, mentre cresce ruotando il cursore in direzione contraria. Tradotto in termini di sensibilità, l'aumento della resistenza incrementa la tensione a parità di percentuale di gas rilevata nell'unità di volume d'aria e viceversa.

Il circuito ha un'uscita analogica a livello di tensione, nel range

0÷5V a seconda del valore di concentrazione di gas rilevato. Per tutti i dettagli bisogna verificare il grafico dei singoli sensori, che riportiamo in queste pagine.

La nostra breakout board è compatibile con diversi sensori aventi la medesima pin-out. La compatibilità è garantita per i seguenti:

- MQ-2 (Fumo);
- MQ-3 (Alcol);
- MQ-4 (Metano);
- MQ-5 (gas di città);
- MQ-6 (GPL);
- MQ-7 (Monossido di carbonio).

Analizziamo uno ad uno questi dispositivi in modo da conoscerne le caratteristiche e sapere come interpretare la tensione fornita dalla scheda quando monta uno di essi. Le caratteristiche di ciascun sensore saranno utili anche per scegliere l'alimentazione e l'interfaccia del caso.

MQ-2

Si tratta di un sensore di fumo a semiconduttore ad elevata sensibilità, ampia gamma di rilevazione, risposta rapida, buona stabilità e lunga durata. Oltre che per rilevare il fumo (compreso quello di sigaretta), può anche essere utilizzato per rilevare gas combustibili come GPL, butano, metano, alcol, propano ed anche l'idrogeno, che è estremamente infiammabile.

Le specifiche tecniche dell'MQ-2 sono le seguenti:

- alimentazione a 5 Vcc o ca;
- temperatura di funzionamento da -20°C a +50°C;
- consumo del riscaldatore di circa 800 mW (@5 V);
- Dimensioni (mm) di 19,6 (diametro) x14,5;

Notare che il sensore può funzionare sia in continua che in alternata perché di fatto è composto da due resistenze: una è il riscaldatore e l'altra lo strato sensibile. Le sensibilità sono le seguenti:

- propano e GPL = $200 \div 5000$ ppm;
- butano = $300 \div 5.000$ ppm;
- metano = $5.000 \div 20.000$ ppm;
- idrogeno = $300 \div 5.000$ ppm;
- alcol = $100 \div 2000$ ppm.

Per sensibilità si intende il campo di concentrazioni (esprese in ppm, ossia parti per milione di aria) rilevabile dal sensore.

La Fig. 1 illustra la tipica sensibilità dell'MQ-2 ai vari gas cui è sensibile, alla temperatura ambiente di 20°C e umidità relativa del 65%, in aria con concentrazione di ossigeno del 21 % (che è quella tipica dell'aria che respiriamo); la resistenza di carico è di $5\text{ k}\Omega$. Le curve hanno questo significato: R_0 è la resistenza del sensore a una concentrazione di 1.000 ppm di idrogeno in aria pura; R_s è la resistenza presentata dall'elemento sensibile a varie concentrazioni dei gas.

MQ-3

Questo sensore è in grado di rivelare la concentrazione di alcol nel sangue attraverso il respiro, proprio come fa un comune etilometro in uso durante i controlli stradali. Allo scopo possiede un'elevata sensibilità e un tempo di risposta molto rapido.

Il sensore fornisce in uscita un valore di resistenza basato sulla concentrazione di alcol. Il circuito di contorno è molto semplice: basta una resistenza in serie all'elemento sensibile.

Le caratteristiche tecniche del sensore sono:

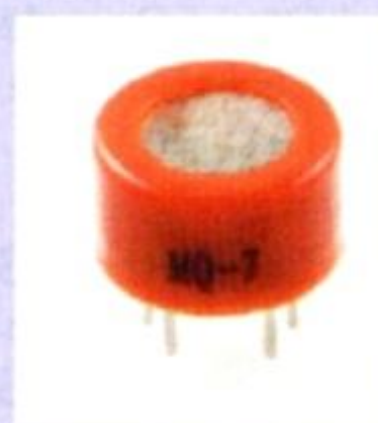
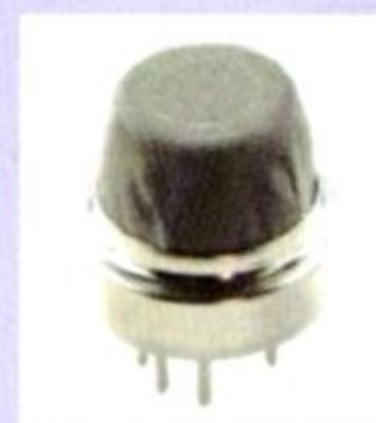
- alimentazione a 5 Vcc o ca;
- Temperatura di funzionamento da -10 a 70°C ;
- Consumo del riscaldatore di circa 750 mW ($@5\text{V}$);
- Dimensioni (mm) = diametro $16,8\text{ mm}$ x altezza $9,3\text{ mm}$.

La Fig. 2 illustra la tipica sensibilità dell'MQ-3 ai vari gas cui è sensibile, alla temperatura ambiente di 20°C e umidità relativa del 65%, in atmosfera con concentrazione di ossigeno del 21 %; la resistenza di carico è di $200\text{ k}\Omega$. Le curve hanno questo significato: R_0 è la resistenza del sensore a una concentrazione di alcol pari a $0,4\text{ mg/l}$ di aria pura; R_s è la resistenza presentata dall'elemento sensibile a varie concentrazioni dei gas che il sensore può rilevare, oltre all'alcol etilico.

MQ-4

Il sensore MQ4 è costruito per rilevare la presenza nell'aria del

I sensori montabili



La breakout board descritta in questo articolo nasce per dotare di sensore di gas un qualsiasi sistema o semplicemente sperimentare una soluzione di rilevamento prima di passare alla produzione. A bordo si possono montare sei diversi sensori, tutti con identica piedinatura ed elemento sensibile di uguale costituzione, pur dosato diversamente per rilevare più certi gas che altri. MQ-3, MQ-4, MQ-5 sono esteriormente identici.

gas naturale (che è composto principalmente da metano) in concentrazioni comprese tra 300 e 10.000 ppm . Il dispositivo è caratterizzato da un'elevata sensibilità e un brevissimo tempo di risposta, come è richiesto da sistemi che richiedono rilevamenti molto rapidi. Come per gli altri sensori già descritti, il circuito richiede una semplice ali-

Fig. 1
Grafico della sensibilità del sensore MQ-2.

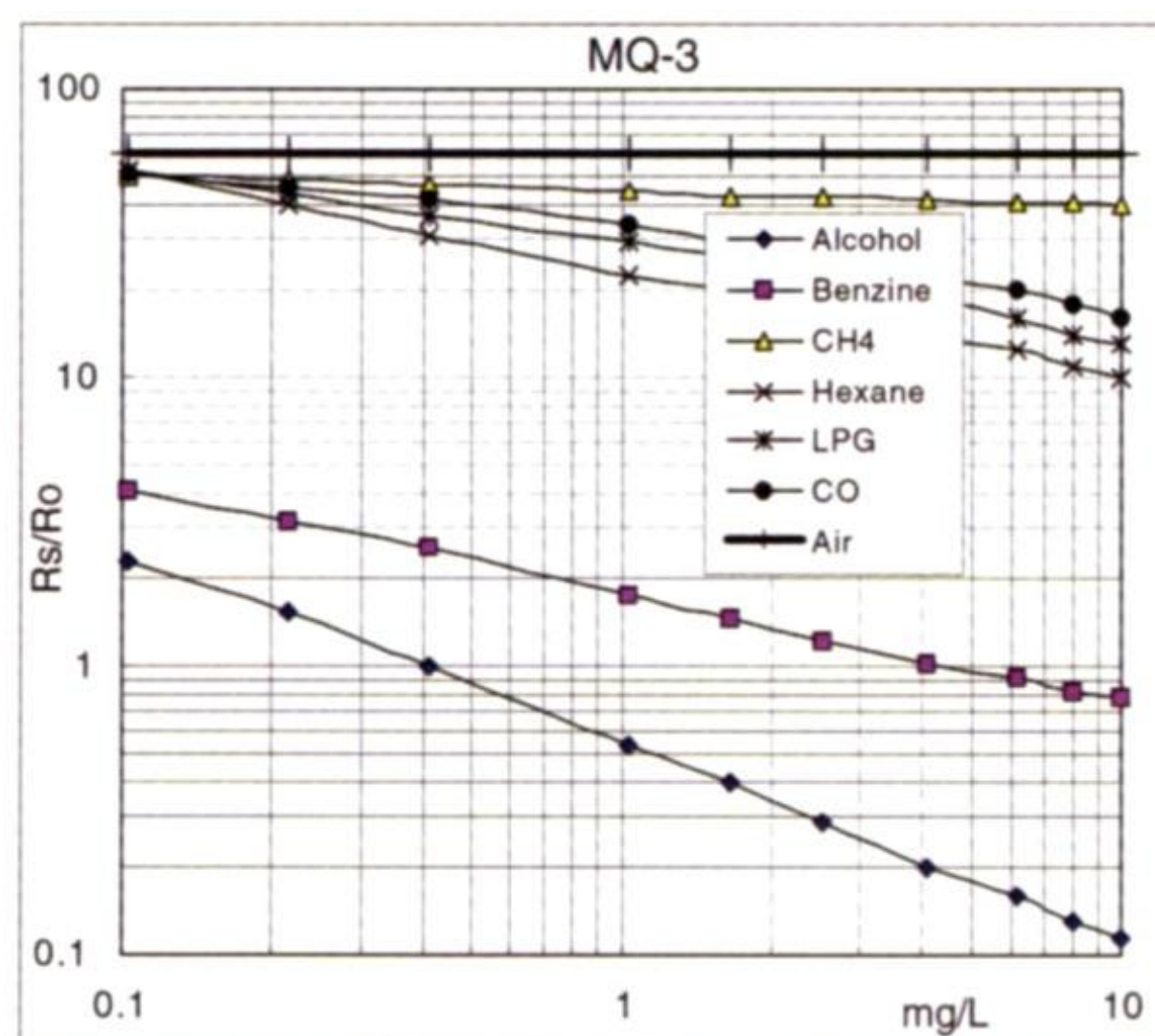
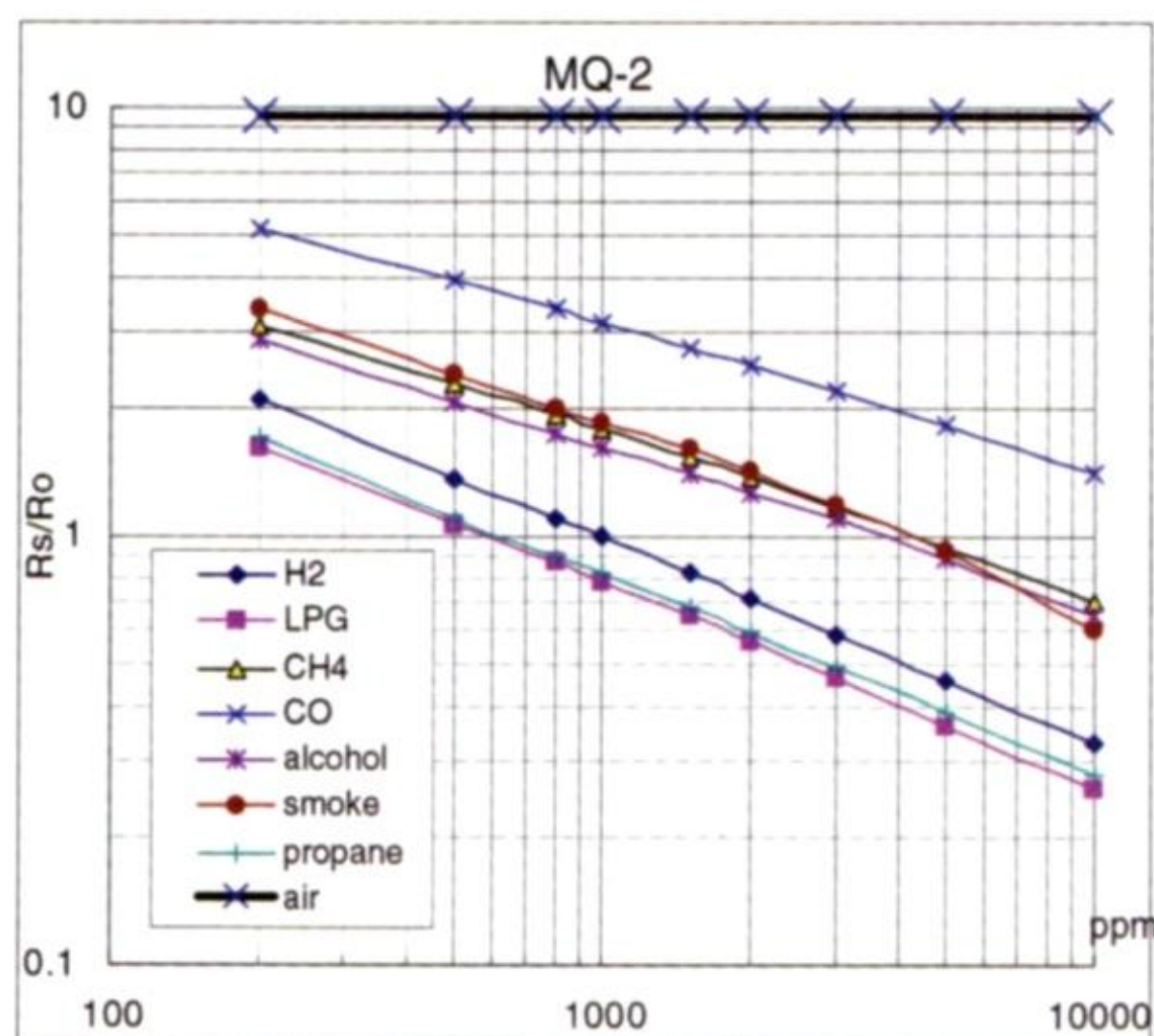


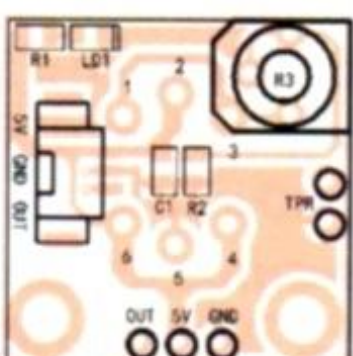
Fig. 2
Curva di sensibilità dell'MQ-3; notate che la sensibilità cala all'aumentare della temperatura e dell'umidità relativa.

Elenco Componenti:

R1: 330 ohm (0805)
R2: 4,7 kohm (0805)
R3: Trimmer 50 kohm
C1: 100 nF ceramico (0805)
LD1: LED verde (0805)
SENS: Sensore MQ-x

Varie:

- Connettore JST 3 vie passo 1.25mm da CS
- Connettore JST 3 vie passo 1.25mm
- Circuito stampato S1209



trazioni comprese tra 200 e 10.000 ppm, rimanendo quasi insensibile ai disturbi provocati dai fumi da cucina, fumo di sigaretta e alcol. Dispone di una elevata sensibilità e tempi di risposta molto rapidi. Le sue specifiche tecniche sono:

- alimentazione di 5 Vcc o ca;
- temperatura di funzionamento da -10°C a +50°C;
- consumo del riscaldatore inferiore a 800 mW (@5V);
- dimensioni (mm) pari a 19,6 (diametro) x 14,5 (altezza);

mentazione a 5V e il collegamento in serie ad una resistenza.

Le specifiche tecniche dell'MQ-4 sono:

- alimentazione a 5 Vcc o ca;
- temperatura di funzionamento da -20°C a +50°C;
- consumo del riscaldatore pari a circa 950 mW (@5 V);
- dimensioni (mm) pari a 19,6 (diametro) x 14,5 (altezza);

L'elemento sensibile è un microtubo di materiale ceramico (allumina, Al₂O₃) su cui è deposto un sottile strato sensibile di biossido di stagno (SnO₂). Il sensore ha

un'elevata selettività nei confronti del GPL (ma anche di iso-butano e propano) che può rilevare anche se disturbato dalla presenza di fumo di sigaretta e alcol.

Nel nostro circuito, alimentato a 5 volt fornisce una tensione di uscita di 2,5V ÷ 4 V a una concentrazione di metano (CH₄) di 5.000 ppm. La concentrazione rilevabile di CNG (ossia gas naturale compresso) è fra 300 e 10.000 ppm.

MQ-5

Questo sensore è in grado di rilevare la presenza nell'aria di GPL, gas naturale, gas di città in concen-

L'elemento sensibile dell'MQ-5 è un microtubo di materiale ceramico (allumina, Al₂O₃) su cui è deposto un sottile strato sensibile di biossido di stagno (SnO₂).

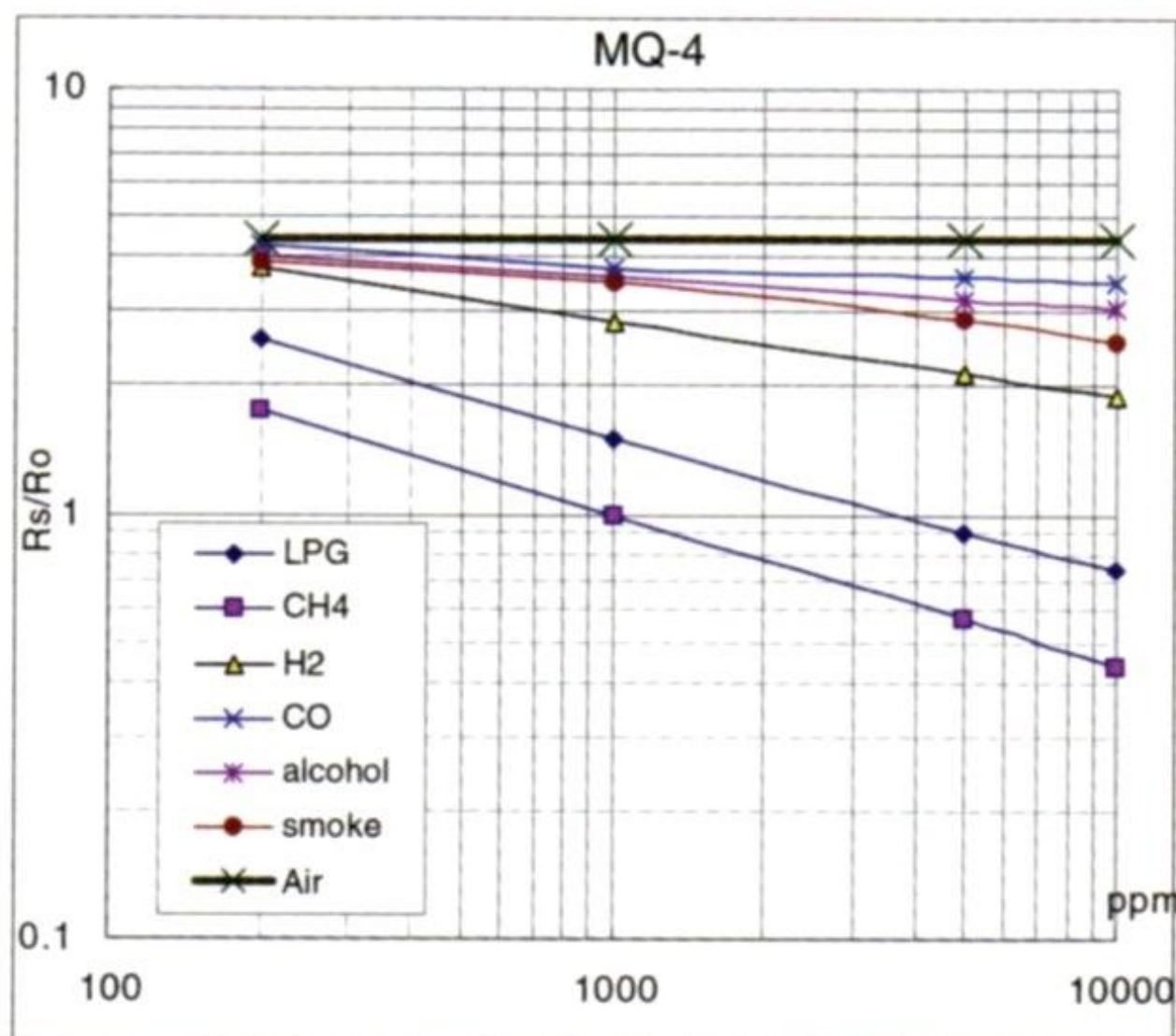
Il sensore ha un'elevata selettività nei confronti del GPL (ma anche di iso-butano e propano) che può rilevare anche se disturbato da fumo di sigaretta, esalazioni di cucina e alcol.

MQ-6

Questo sensore è pensato principalmente per rilevare la presenza nell'aria di GPL (Gas di Petrolio Liquefatto) in concentrazioni comprese tra 300 e 10.000 ppm, però

Fig. 3

Curva di sensibilità dell'MQ-4; la sensibilità cala all'aumentare della temperatura e della umidità relativa.



MQ-5

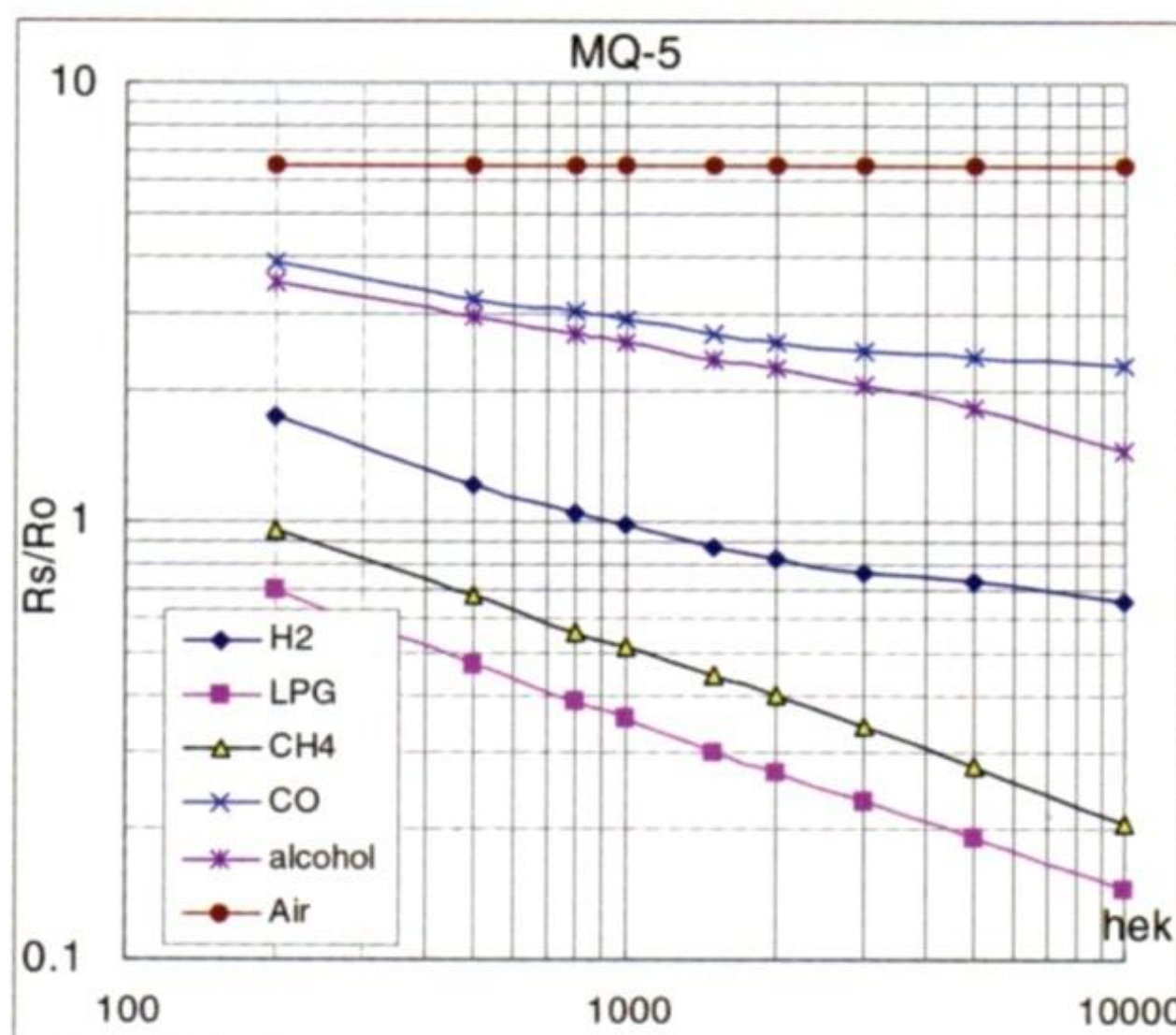


Fig. 4

Curva di sensibilità dell'MQ-5; la sensibilità cala all'aumentare della temperatura e della umidità relativa.

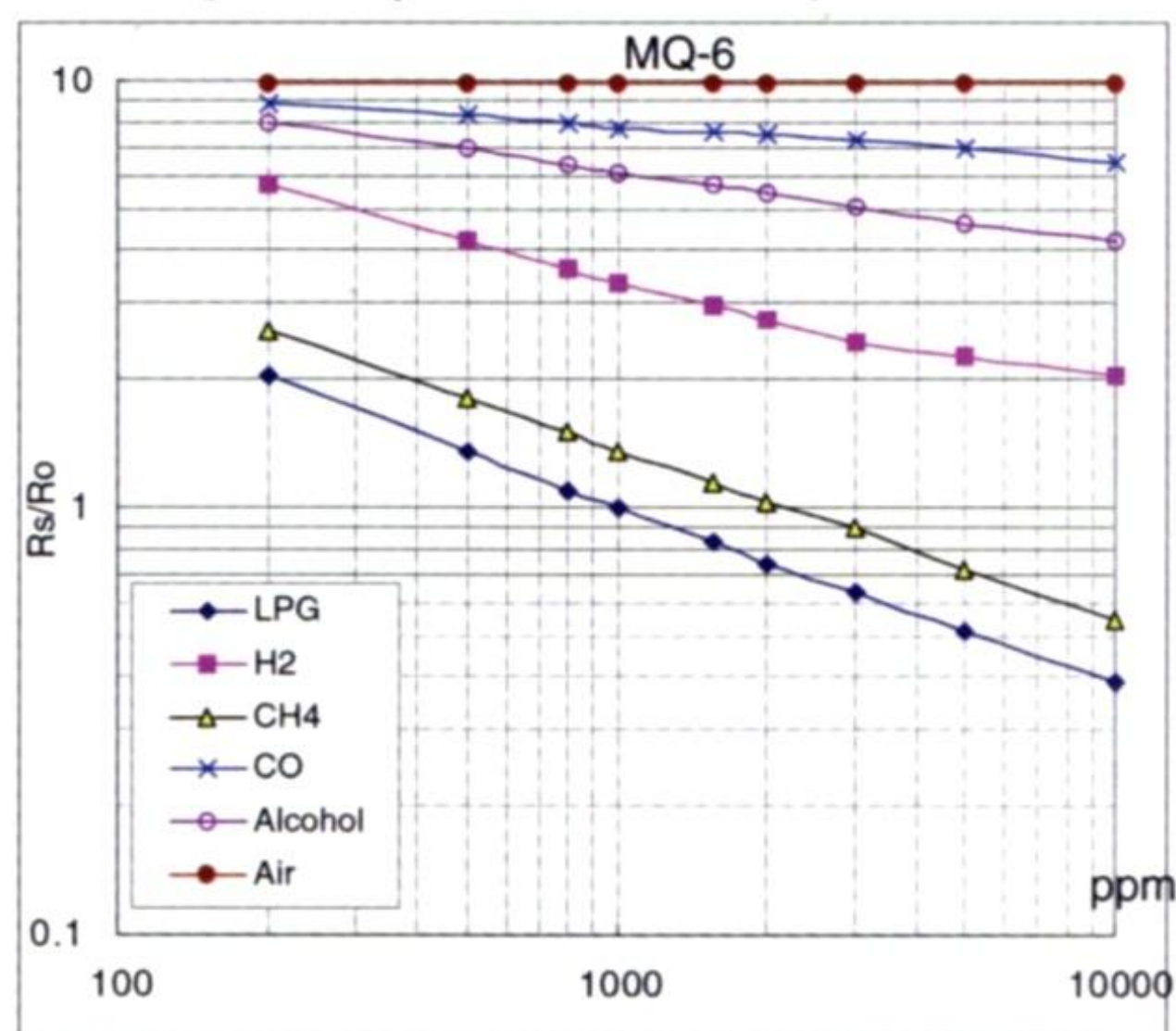


Fig. 5
Caratteristica della sensibilità del sensore MQ-6 ai gas che può rilevare.

può rilevare anche CO (monossido di carbonio), idrogeno, metano, propano e iso-butano. Il circuito applicativo è quello degli altri sensori.

Le specifiche tecniche sono:

- alimentazione a 5 Vcc o ca;
- temperatura di funzionamento da -10°C a +50°C;
- consumo del riscaldatore di circa 950 mW (@5V);
- dimensioni (mm) pari a 19,6 (diametro) x 14,5 (altezza).

L'elemento sensibile è un microtubo di materiale ceramico (allumina, Al_2O_3) su cui è depositato un sottile strato sensibile di biossido di stagno (SnO_2). L'elemento sensibile assicura elevata sensibilità e rapida risposta al gas.

Il sensore ha un'elevata selettività nei confronti del GPL (ma anche di iso-butano e propano) che può rilevare anche se disturbato dalla presenza di fumo di sigaretta, esalazioni di cucina e alcol. La Fig. 5 illustra la tipica caratteristica della sensibilità del sensore MQ-6 per i gas cui questo dispositivo è sensibile; la curva è riferita a una temperatura ambiente di 20 °C e a un'umidità relativa del 65%, oltre che una concentrazione di ossigeno, nell'aria, del 21%. La resistenza di carico usata per il test che ha

restituito la curva caratteristica è di 20 kΩ. Nella curva, R_o è la resistenza dell'elemento sensibile a una concentrazione di riferimento di 1.000 ppm di GPL in aria pura. Le curve di R_s fornite sono riferite ognuna a uno dei gas (R_s è la resistenza del sensore a varie concentrazioni di gas).

MQ-7

Si tratta di un sensore dedicato esclusivamente a rilevare la presenza di monossido di carbonio (CO) in concentrazioni comprese tra 20 e 2.000 ppm. Caratterizzato dall'elevata sensibilità e da brevissimi tempi di risposta, si interfaccia in maniera molto semplice e richiede un'alimentazione di 5 volt. Come gli altri, permette di rilevare la concentrazione di gas sulla base della caduta di tensione ai capi della resistenza di carico. Per leggere il segnale mediante un microcontrollore e quindi in formato digitale, basta collegare l'uscita della breakout board a un convertitore A/D.

Le specifiche tecniche del sensore MQ-7 sono le seguenti:

- alimentazione a 5 Vcc o ca;
- consumo del riscaldatore intorno ai 900 mW (@5V);
- dimensioni (mm) 16,8 (diametro) x 9,3 (altezza);

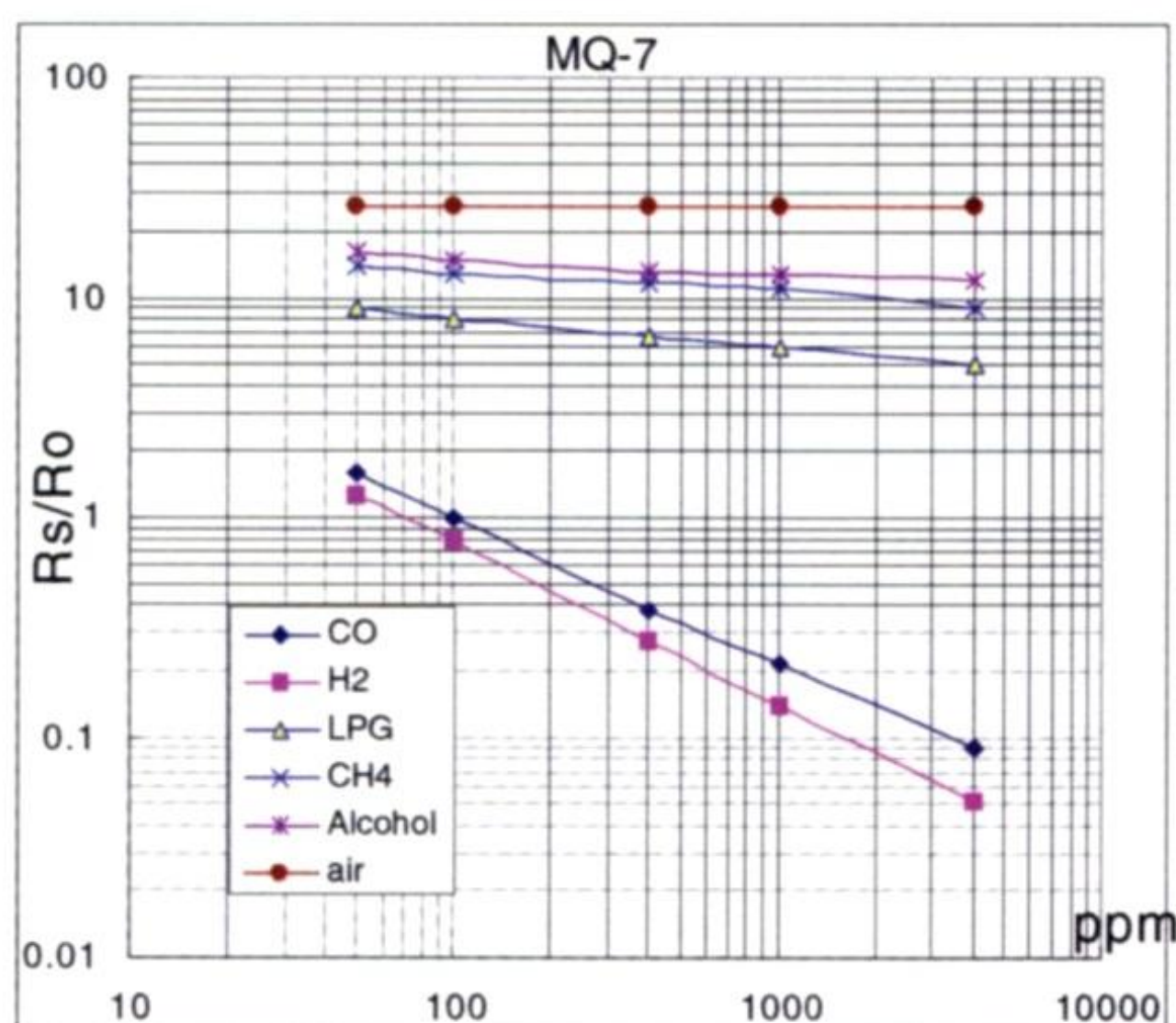


Fig. 6 - Tipica risposta dell'MQ-7 ai vari gas cui è sensibile, alla temperatura ambiente di 20 °C e umidità relativa del 65%, in atmosfera con concentrazione di ossigeno del 21%; la resistenza di carico è di 10 kΩ. Le curve hanno questo significato: R_o è la resistenza del sensore a una concentrazione di 100 ppm di CO in aria pulita; R_s è la resistenza presentata dall'elemento sensibile a varie concentrazioni dei gas.

L'elemento sensibile è, anche in questo caso, un microtubo di materiale ceramico (allumina, Al_2O_3) su cui è depositato un sottile strato sensibile di biossido di stagno (SnO_2).

La Fig. 6 riporta la curva di sensibilità dell'MQ-7.

CONNESSIONI

Siccome tutti i sensori utilizzati condividono la stessa struttura e piedinatura, la scheda di prototipazione qui proposta permette di ospitare quello che desiderate senza apportare modifiche: l'unica cosa da variare è la posizione del cursore del trimmer, in modo da ottenere la resistenza indicata per lo specifico sensore montato. I test-point TPR presenti nel circuito permettono di tarare con precisione (la tolleranza dipende dalla qualità dello strumento utilizzato...) la resistenza di carico

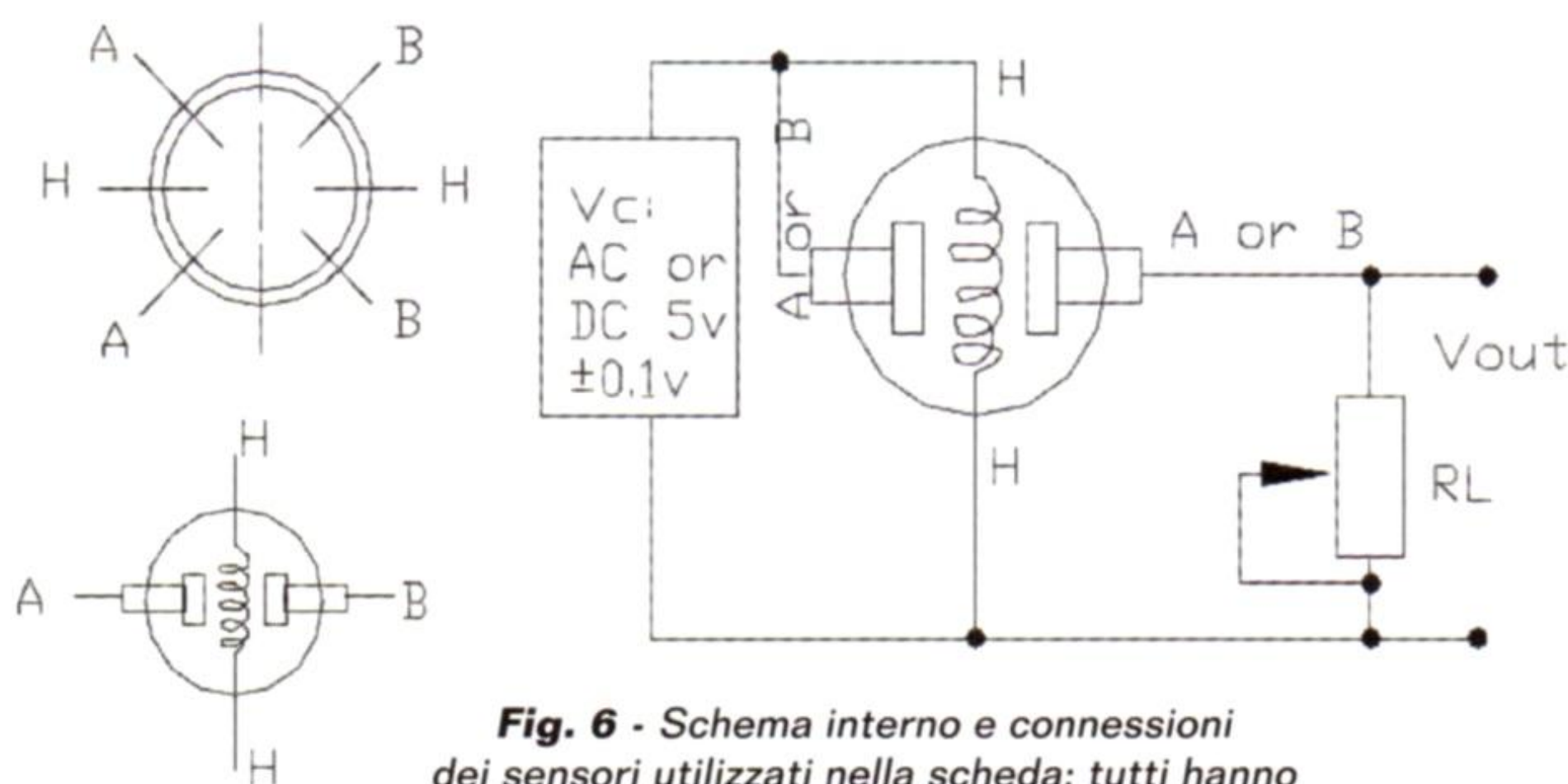


Fig. 6 - Schema interno e connessioni dei sensori utilizzati nella scheda: tutti hanno la stessa configurazione e piedinatura.

servendosi di un multimetro impostato sulle misure di resistenza, ovvero di un ohmetro, per chi lo ha a disposizione.

Essendo nata come breakout board, la scheda qui descritta dispone di tutte le connessioni riportate su una fila di piazzole che permettono di montare un pin-strip a passo 2,54 mm che agevola il montaggio in altre unità, ovvero un connettore miniatura a tre poli. Le piazzole per i test-point TPR sono a parte.

REALIZZAZIONE PRATICA

Passiamo adesso dalla teoria alla pratica e vediamo la costruzione della breakout board, che richiede un circuito stampato ottenibile per fotoincisione seguendo le due tracce (il c.s. è a doppia faccia) dei lati rame scaricabili sotto forma di file immagine dal nostro sito www.elettronica.in.it insieme ai file del progetto. Ottenuta la basetta e procurati i componenti, tutti per montaggio superficiale, potete iniziare a disporli ognuno con i terminali in corrispondenza delle rispettive piazzole (che avrete avuto cura di spalmare con pasta fluxante allo scopo di agevolare le saldature) stagnandoli poi a partire dalle resistenze fisse e dal condensatore, per proseguire con il LED e terminare con il

connettore miniatura e il trimmer. L'orientamento del LED è visibile nel piano di montaggio pubblicato in queste pagine. Se volete inserire la breakout board in un altro circuito montate anche i pin-strip in corrispondenza delle piazzole 5V, GND e OUT.

Ricordate che il montaggio di componentistica SMD richiede un po' di manualità e una buona vista da vicino: se non l'avete, procuratevi una lente d'ingrandimento, che vi aiuterà a verificare che i componenti più piccoli siano saldati correttamente e che non vi siano baffi di stagno a cortocircuitare terminali dell'integrato o pad di componenti vicini. Alle doti "fisiche" dovete aggiungere del filo di lega saldante da 0,5 mm di diametro, una pinzetta per posizionare i componenti e un saldatore a punta finissima da non più di 20 watt.

Inseriti tutti i componenti, non resta che montare il sensore, rammentando che durante l'utilizzo dovrà stare rivolto verso il basso in modo da rilevare la presenza del gas; nel caso vogliate realizzare un etilometro, il sensore (quindi anche la basetta) va posto in fondo a un tubo rivolto all'imboccatura con la zona sensibile (quella con la retina metallica).

Se dovete rilevare dei gas più

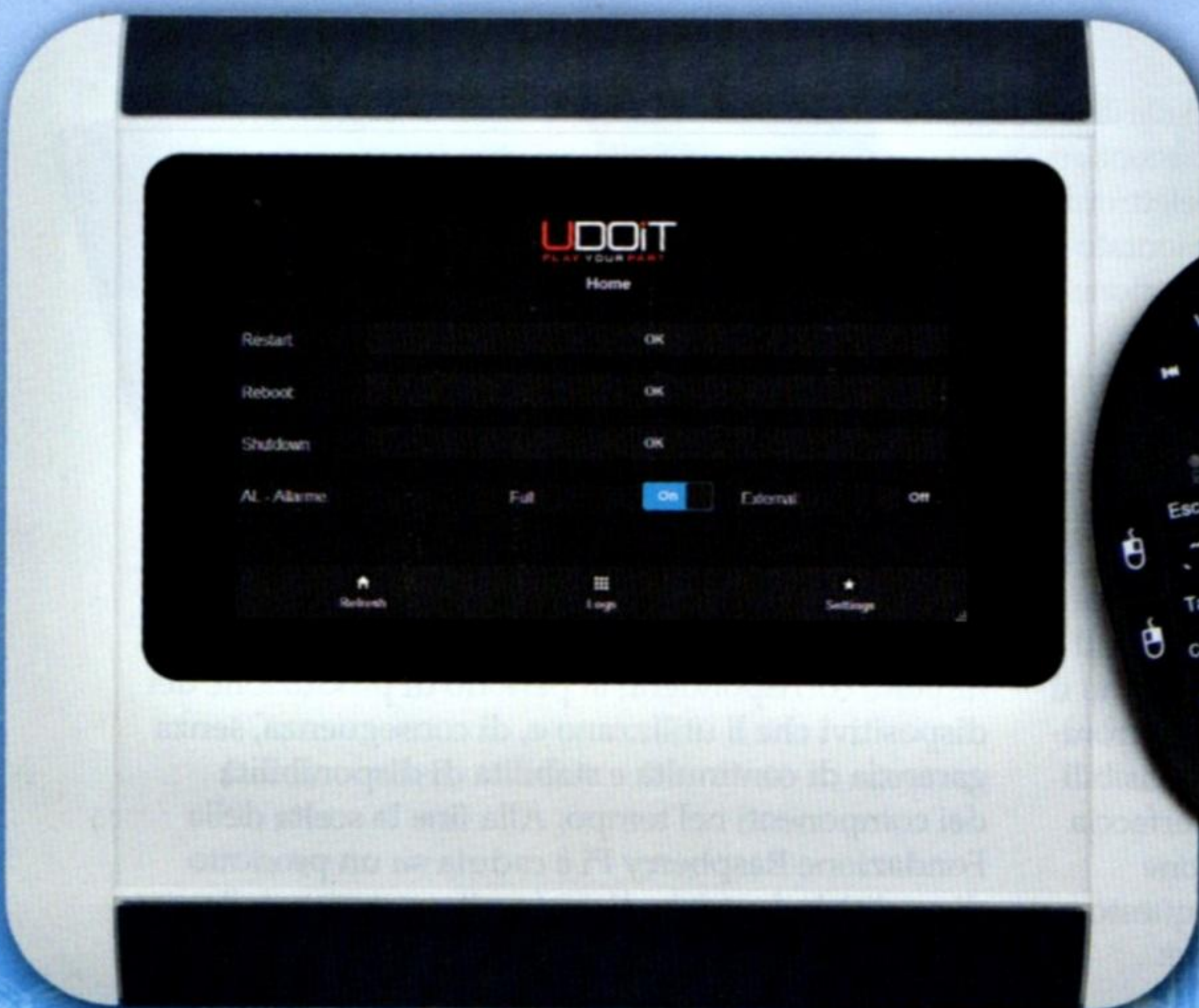
leggeri dell'aria, quindi idrogeno, metano, monossido di carbonio ecc. il sensore va posto in cima al luogo dove effettuare il rilevamento; dovete fare il contrario se intendete rilevare GPL, propano e butano, che essendo più pesanti dell'aria tendono a precipitare (il sensore va quindi posto in basso, in prossimità del pavimento). Scelto il sensore, sulla base delle specifiche tecniche indicate per ciascuno nei precedenti paragrafi, tarate il trimmer; durante la misura accertatevi di non toccare con le dita i puntali (le zone metalliche) altrimenti la resistenza del corpo altera la misura. La soluzione ideale consiste nel collegare i puntali ai test-point tramite cavetti terminanti con piccoli "coccodrilli". Le piazzole per i test-point TPR sono a parte e possono essere riportate mediante pin-strip sul circuito in cui si installa la breakout board.



per il MATERIALE

Tutto il materiale necessario per realizzare questo progetto è acquistabile presso Futura Elettronica. La Breakout compatibile con i sensori della serie MQ (Cod. MQBREAKOUT) è in vendita a Euro 3,50. Sono inoltre disponibili i sensori della serie MQ: sensore di fumo (cod. MQ2) a Euro 3,00; sensore di alcool (cod. MQ3) a Euro 4,50; sensore di gas metano (cod. MQ4) a Euro 3,00; sensore di GPL e gas naturale (cod. MQ5) a Euro 3,00; sensore di GPL (cod. MQ6) a Euro 3,50; sensore di monossido di carbonio (cod. MQ7) a Euro 4,50.

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



DOMOTICA OPEN SOURCE con i moduli UDOiT

Dotiamo il gateway di UDOiT, basato su Raspberry Pi, di un display touch screen e di un contenitore adatto a fargli fare bella figura in qualunque ambiente.

di MARCO MAGAGNIN

Un sistema di controllo domotico, anche se dispone di radiocomando e connettività web per la gestione, è opportuno che abbia anche un'interfaccia di gestione diretta, per esempio sotto forma di touch screen montato sulla centralina. Per confezionare in questo modo il gateway di UDOiT, basato su Raspberry Pi, abbiamo bisogno di due elementi principali: un display LCD dotato di schermo touch screen ed un bel contenitore nel quale racchiudere il tutto. Per il display vi sono diverse soluzioni possibili ma, dopo circa due anni dall'annuncio, la fondazione Raspberry Pi ha finalmente rilasciato la sua soluzione touch screen ufficiale, che ha avuto un tale successo di vendite da richiedere diverse settimane di attesa per le nuove disponibilità. Questo display risolve il problema delle

soluzioni precedenti, che per lo più richiedono modifiche al kernel del sistema operativo per poter essere utilizzati. Questa necessità si ripercuote sulla possibilità di mantenere aggiornata la distribuzione del sistema operativo Raspbian con i nuovi rilasci, che si susseguono abbastanza frequentemente. Per utilizzare il display ufficiale, al contrario, è sufficiente aggiornare il sistema operativo Raspbian con i soliti comandi:

```
apt-get update
apt-get upgrade
apt-get dist-upgrade
```

Abbiamo avuto la possibilità di utilizzare uno dei nuovi display disponibili ed anche il prototipo del conte-

nitore per display e Raspberry Pi progettato dalla ditta TEKNO ENCLOSURE, azienda nota tra gli appassionati per la produzione di contenitori per progetti elettronici. Nella foto di apertura vediamo il display montato nel contenitore con visibile l'applicazione di gestione che abbiamo presentato in questi articoli.

Cogliamo l'occasione per sottolineare, come nel caso delle configurazioni presentate nel numero 200 di Dicembre 2015/Gennaio 2016 della rivista, che tutti gli argomenti presentati in questo articolo possono essere utilizzati in generale per integrare qualsiasi applicazione, e non unicamente il sistema di gestione di automazione domestica UDOiT. Torniamo ad occuparci del display. Aprendo la scatola troviamo il display con i suoi connettori ed una scheda elettronica di conversione del formato di uscita video visibili in Fig. 1. Il display è caratterizzato da un'interfaccia DPI (Display Parallel Interface). La Fondazione Raspberry Pi ha comunicato che la scelta di questo display doveva rispondere a diverse esigenze:

- Riproduzione dei colori in alta qualità;
- Qualità dei singoli Pixel (a volte è possibile riconoscere i confini dei singoli pixel);
- Rapporto di contrasto elevato;
- Elevato angolo di visibilità laterale;
- Convenienza;
- Disponibilità nel tempo.

L'ultima caratteristica è la più complessa da ottenere in quanto, generalmente questi tipi di display sono



Fig. 1

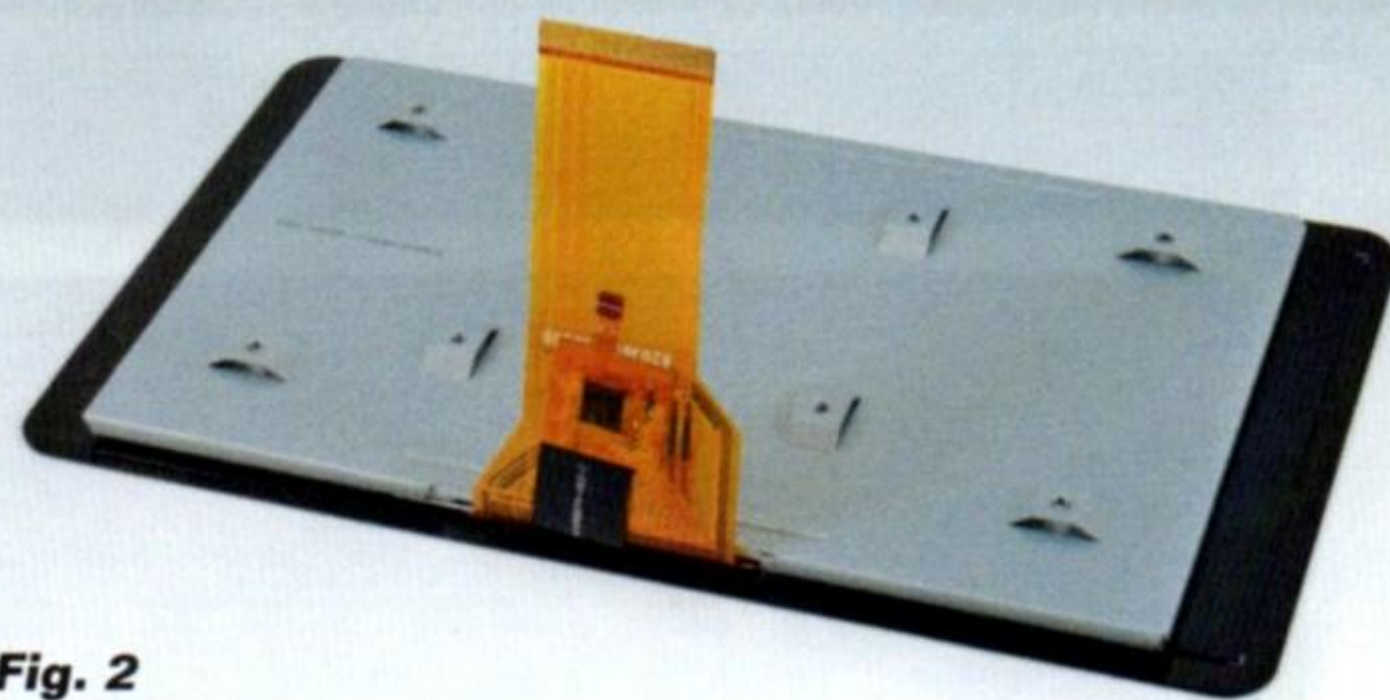


Fig. 2

utilizzati in tablet e dispositivi mobili, che richiedono specifiche differenti a seconda del produttore e del modello e che vengono prodotti per periodi limitati, corrispondenti al periodo di produzione dei dispositivi che li utilizzano e, di conseguenza, senza garanzia di continuità e stabilità di disponibilità dei componenti nel tempo. Alla fine la scelta della Fondazione Raspberry Pi è caduta su un prodotto di qualità industriale di un fornitore storico della Fondazione stessa, la Inelco Hunter con base in Gran Bretagna, che offre le seguenti caratteristiche:

- Display video con risoluzione RGB 800×480 a 60 fps;
- Profondità di colore a 24-bit;
- Schermo touchscreen FT5406 capacitivo a 10 punti di contatto contemporanei;
- Angolo di visione laterale di 70 gradi;
- Telaio di sostegno in metallo con foratura adatta al montaggio di Raspberry Pi;
- Interfaccia video DPI (Display Parallel Interface).

L'ultima caratteristica spiega il motivo della adozione della scheda di conversione dell'uscita video. Riassumiamo le diverse tipologie di uscita del segnale video possibili per il microcontrollore Raspberry Pi:

- HDMI – è lo standard video, allo stato attuale, più comunemente adottato, e che permette, nel caso di Raspberry Pi, di collegarlo ad un monitor od un apparecchio TV fino alla definizione 4K con una deriva di segnale relativamente bassa e conseguenti ridotte emissioni spurie. Sempre nel caso di Raspberry Pi sono implementate le specifiche CEC, che permettono la comunicazione bidirezionale tra Raspberry Pi e TV ed EDID che permettono il riconoscimento automatico delle modalità video supportate del monitor o TV. Il problema con una soluzione display in HDMI è data dal costo elevato dell'hardware di conversione tra HDMI ed interfaccia nativa del display;
- DPI (Display Parallel Interface) – è un'interfaccia parallela a 24 bit con clock ed altri segnali di sincronizzazione per un totale di 28 conduttori con una frequenza di scansione di circa 70MHz.

Fig. 3

Questa interfaccia è stata scartata nell'utilizzo con tablet e smartphone a causa delle interferenze elettromagnetiche generate dall'alto numero di conduttori e dal consumo elevato;

- DSI (Display serial interface) – è un'interfaccia seriale ad alta velocità basata su un certo numero di linee dati ad 1Gbit. La tensione di funzionamento delle linee dati è di 200mV, che permette di mantenere molto bassi i livelli di disturbo elettromagnetico e il consumo. Sfortunatamente i display nativi DSI sono prodotti solo per impieghi specifici, per lo più destinati ad equipaggiare tablet e smartphone e sono soggetti alla disponibilità commerciale correlata al periodo di vita del prodotto di destinazione, con scarsa garanzia di continuità;
- DBI (Display Bus Interface) – è una vecchia tecnologia che normalmente impiega hardware e memorie interne aggiuntive per la bufferizzazione dei fotogrammi allo scopo di evitare il fenomeno delle immagini "strappate". Queste dotazioni hardware aggiuntive rendono i display DBI eccessivamente costosi.

A fronte di queste considerazioni e delle caratteristiche delle uscite video presenti su Raspberry Pi una delle scelte possibili poteva essere la adozione di un display video con ingresso DPI, che dal punto di vista economico rappresenta la scelta ottimale, collegato direttamente ai pin di GPIO di Raspberry Pi. Questo avrebbe occupato pressoché tutti i pin di GPIO disponibili su Raspberry Pi senza lasciare concreta possibilità di collegare altri dispositivi. Per questo motivo è stata introdotta la scheda di conversione da DSI a DPI, in modo da mantenere la scelta di adottare un display DPI economico ed utilizzare l'uscita video DSI, il cui connettore è presente su Raspberry Pi fin dalla sua prima apparizione.

MONTIAMO IL DISPLAY

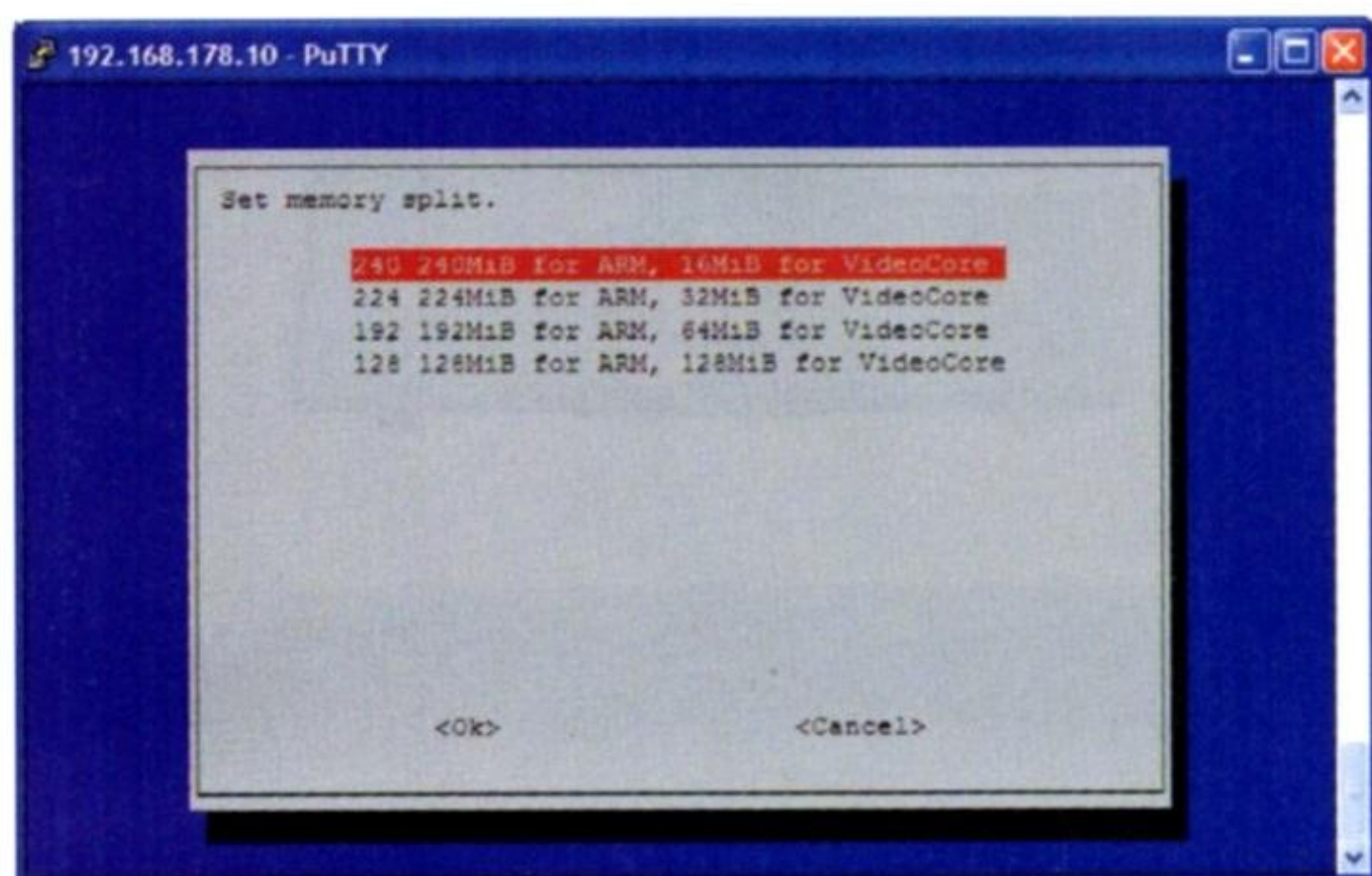
Abbiamo visto che la soluzione display progettata dalla Fondazione Raspberry Pi è costituita da un display LCD e da una scheda di conversione del bus di comunicazione video. Nella parte inferiore del display sono presenti due connettori in formato flat cable. Uno da 28 vie collegato al display vero e proprio ed uno più piccolo che fa capo al sensore touch. Il verso di montaggio è visibile nella foto di Fig. 2. Il cavo a 28 vie va collegato al connettore nella parte inferiore della scheda di conversione. Per inserire il cavo flat è necessario aprire il connettore "tirando" con molta delicatezza le due piccole linguette laterali, meglio aiutandosi con un piccolo cacciavite in plastica, in modo da liberare la "pinza" che trattiene il



cavo. Si inserisce il cavo, sempre con molta delicatezza ed attenzione, curando che sia ben perpendicolare al connettore. Per l'allineamento ci si può aiutare con il limite superiore del supporto dei terminali dei cavi. Una volta ben posizionato si richiude il connettore "spingendo", sempre con delicatezza, sulle "alette" esterne della chiusura. Fatta questa operazione si posiziona la scheda sul supporto in metallo del display, in modo da far corrispondere i fori di fissaggio sulla scheda con quelli filettati sul supporto dello schermo. Fissate la scheda con i distanziali forniti nella confezione del display come visibile in Fig. 3. Nella foto è mostrato il display già montato nel contenitore e fissato con le due barrette trasversali di fissaggio. Potete eseguire il montaggio del display indipendentemente dal contenitore ed inserirlo poi successivamente.

Per usi diversi dal gateway per il sistema di gestione UDOiT è possibile utilizzare contenitori differenti disponibili in commercio. Inserite il cavo più piccolo nel suo alloggiamento nella parte superiore della scheda, con lo stesso metodo dello sblocco del fermo con le alette, posizionamento del cavo e successivo bloccaggio. Ora è il turno del cavo piatto da 15 vie, anch'esso incluso nella confezione, che collega l'uscita DSI di Raspberry Pi al corrispondente ingresso della scheda di conversione. Innanzitutto il connettore DSI è quello sopra all'alloggiamento della

Fig. 4



SD Card, da non confondere con l'altro, identico, destinato alla Camera Pi. Con lo stesso procedimento, il cavo a 15 vie deve essere inserito nel connettore della scheda di conversione con i contatti rivolti verso l'alto. I contatti sono riconoscibili per il loro colore "argento". A questo punto vi consigliamo di inserire la SD Card nel suo alloggiamento, prima di collegare il terminale opposto del cavo al connettore di Raspberry Pi, con i contatti argentati rivolti verso i connettori USB di Raspberry Pi (Fig. 4). Per quanto riguarda l'alimentazione esistono tre possibilità:

1. Collegare l'alimentazione esterna al connettore micro USB della scheda di conversione e collegare un cavo USB A/micro USB tra il connettore USB standard della scheda di conversione ed il connettore di alimentazione micro USB di Raspberry Pi;
2. Collegare i terminali di alimentazione presenti sul connettore a pettine della scheda di conversione ai terminali di alimentazione 5V e GND del connettore GPIO di Raspberry Pi, utilizzando i cavetti con i terminali femmina/femmina compresi nella confezione;
3. Alimentare separatamente con due fonti di alimentazione e due cavi micro USB distinti la scheda di conversione e Raspberry Pi.

Vi consigliamo l'ultima possibilità, utilizzando un alimentatore con due uscite a 5V ed almeno 1A ciascuna come il modello 8220-PSSEUSB14W o 8220-PSSEUSB14B, reperibili presso Futura Elettronica. Questa soluzione permette di dare alimentazione in simultanea sia alla scheda di conversione che a Raspberry Pi. In caso alimentaste il tutto con due alimentatori separati date tensione prima alla scheda di conversione e poi a Raspberry Pi. In caso contrario il display può non venire riconosciuto (schermo nero). Prima di dare tensione alla "centrale" di controllo del sistema UDOiT collegate il "dongle" UDOiT che funge da master della rete a radiofrequenza ad un connettore USB di Raspberry Pi. Collegate un cavo di rete oppure il dongle WiFi, se avete già configurato

Fig. 5

Raspberry Pi come client WiFi o come Access Point. Per configurare la rete fate riferimento agli articoli precedenti. Se, come abbiamo fatto noi, volete collegare una tastiera esterna senza fili, collegate l'apposito dongle ad un connettore USB libero. Finalmente... date tensione. Se la vostra distribuzione Raspbian è aggiornata vedrete lo "splash screen" sul display e poi tutti i messaggi a terminale che documentano l'avanzamento del processo di boot. Al termine verrà presentato il prompt di login. Se sul display non viene evidenziato nulla è necessario aggiornare la distribuzione Raspbian. Collegatevi con PuTTY come utente root, e digitate i comandi:

```
apt-get update
apt-get upgrade
apt-get dist-upgrade
```

e concludete con un

```
reboot
```

A questo punto tutto dovrebbe funzionare correttamente. Il display viene riconosciuto automaticamente, e considerato come il display predefinito, al posto del display HDMI standard. Se, con il display installato, si desidera comunque reindirizzare l'uscita video verso l'uscita HDMI è possibile farlo modificando il file "config.txt" nella cartella /boot aggiungendo in fondo al file la configurazione:

```
display_default_lcd=0
```

In questo caso ricordatevi di assegnare almeno 128 MB di memoria alla scheda video con lo strumento raspi-config. Se invece utilizzate il solo display touch potete azzerare (0MB) o minimizzare (16MB)

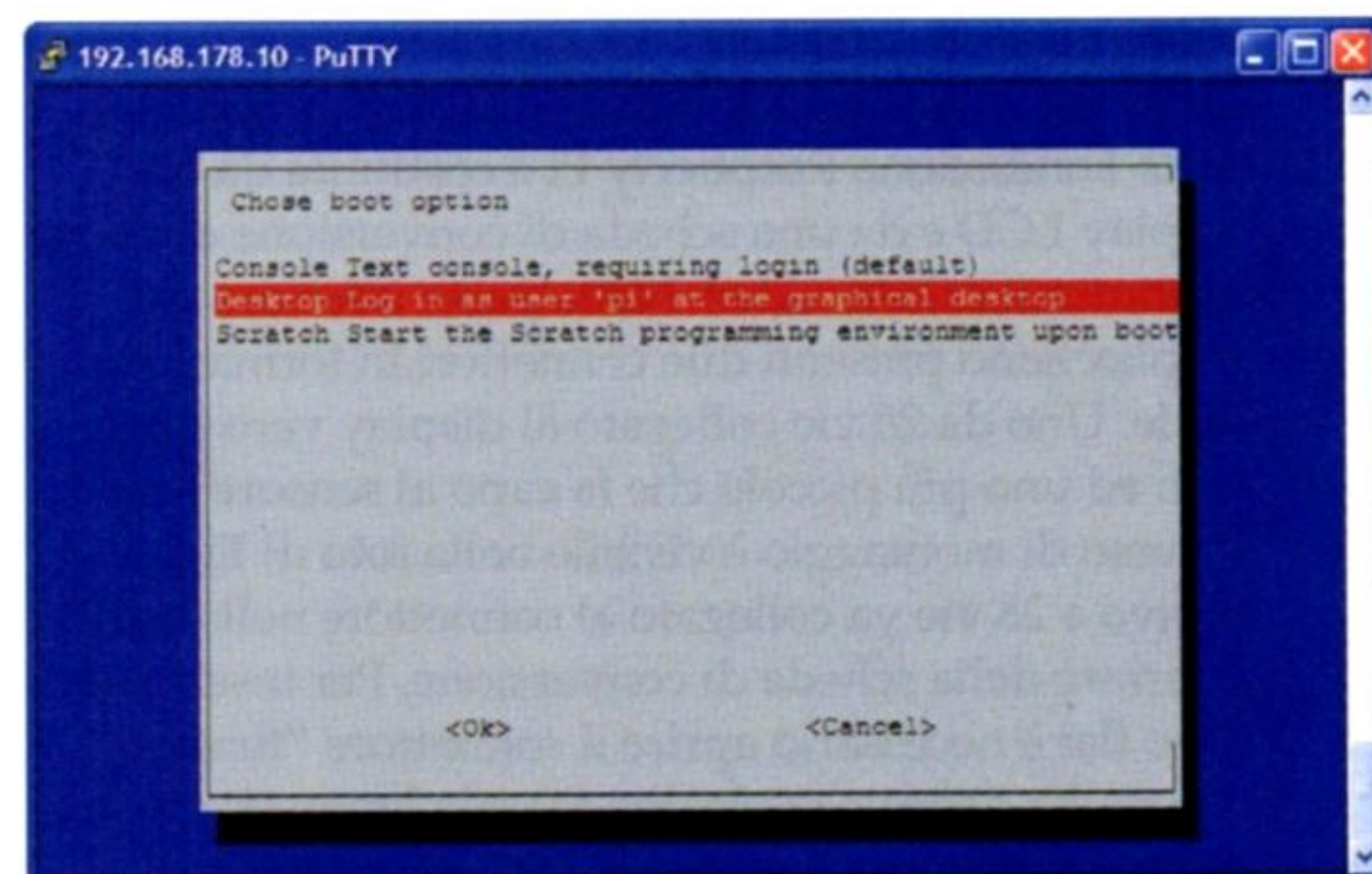


Fig. 6

Raspbian per Raspberry Pi 2 è ora basata su Debian 8 "Jessie"

Gli sviluppatori di Raspbian hanno appena aggiornato la distribuzione, che ora è basata su Debian 8 "Jessie", l'ultima versione stabile disponibile al momento. Gli sviluppatori di Debian scelgono i nomi da assegnare alle distribuzioni dai personaggi di "Toy Story" e Jessie è la bambola cow-girl in Toy Story 2. Dunque, che c'è di nuovo? Molti dei cambiamenti sono invisibili agli utenti normali, finalizzati a migliorare le prestazioni e la gestione dei processi. Questo introduce alcune differenze da valutare per chi utilizza Raspberry Pi come sistema di sviluppo "embedded". Gli aggiornamenti più visibili riguardano il fatto che ora l'impostazione predefinita prevede la partenza automatica del desktop, ovviamente ciò può essere cambiato nel nuovo strumento di configurazione che sostituisce raspi-config (Fig. A). Altre modifiche visibili sono l'adozione del browser web Epiphany, la suite di strumenti office LibreOffice, il client di posta Claws Mail, ed una serie di strumenti per lo sviluppo Java. LibreOffice è una suite di strumenti office con funzionalità professionali e compatibile con i file di Microsoft Office. Include un word processor, un foglio di calcolo, oltre a strumenti per la grafica di presentazione, disegno vettoriale e gestione database. L'utilizzo di questi strumenti

non presenta difficoltà per chi utilizza già strumenti office e sono stati ottimizzati per funzionare bene su Raspberry Pi 2.

Per chi utilizza il GPIO ora non vi è più la necessità di accedervi come utente "root". Nella voce di menu dedicata alla programmazione sono disponibili due nuovi ambienti per lo sviluppo di applicazioni Java (Fig. B): BlueJ e Greenfoot (sviluppati dall'Università del Kent e da Oracle). In definitiva, a nostro parere, la nuova distribuzione costituisce un avvicinamento alla versione di Debian per desktop, con una strizzatina d'occhio a chi lo vuole utilizzare come mini PC desktop. Nulla vieta comunque di continuare ad utilizzare Raspberry Pi come ambiente di sviluppo "embedded". Sul sito di Raspberry sono scaricabili entrambe le versioni di Raspbian: Wheezy e Jessie.

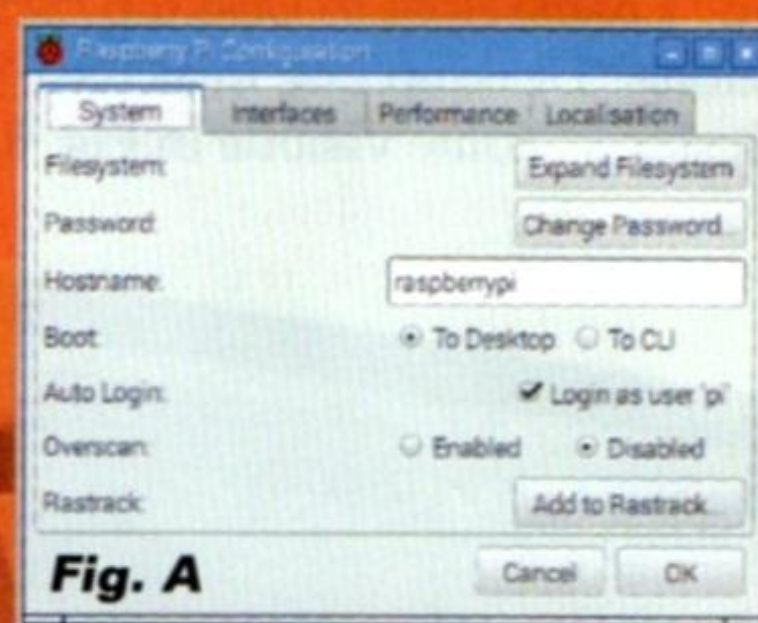


Fig. A

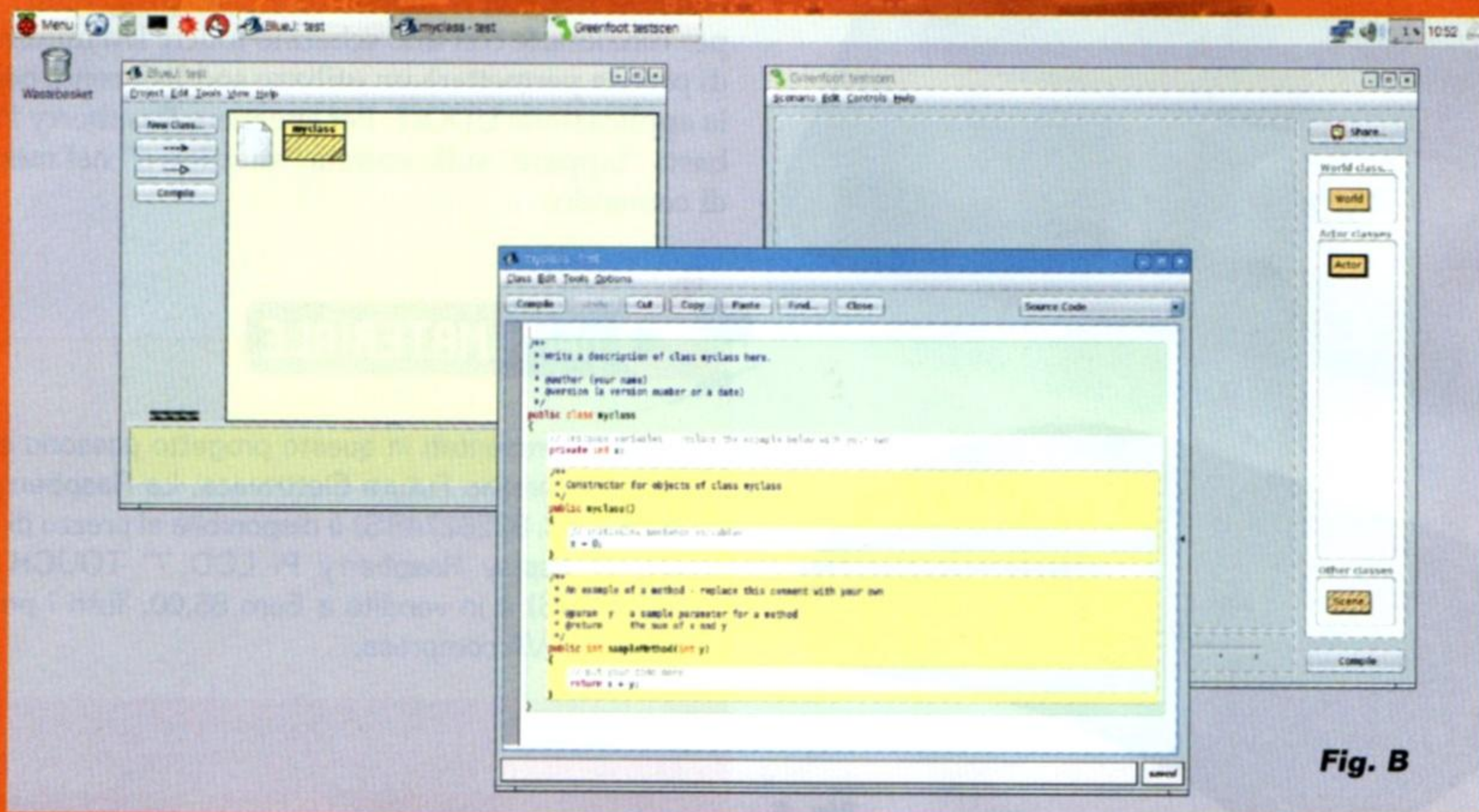


Fig. B


```
/etc/xdg/lxsession/LXDE-pi/autostart - RaspberryPi - E...
# Auto run the desktop
#@lxpanel --profile LXDE-pi
#@pcmanfm --desktop --profile LXDE-pi
#@xscreensaver -no-splash

# Auto run the browser
@xset s off
@xset -dpms
@xset s noblank
@midori -e Fullscreen -a http://192.168.178.10

Riga: 1/11 Codifica: 1252 (ANSI - Latino)
```

la quantità di memoria assegnata al video, sempre utilizzando lo strumento raspi-config, ed andando alla voce di menu "Advanced Options>>Memory Split" a tutto vantaggio delle prestazioni del sistema come visibile in Fig. 5.

BROWSER WEB AL BOOT

Con un'ultima configurazione potete fare in modo che il browser web parta in automatico al boot a pieno schermo e con la pagina di comando aperta come pagina predefinita. Per questo è necessario riaprire lo strumento raspi-config e nella voce di menu "Enable Boot to Desktop/Scratch" scegliere l'opzione "Desktop Log in as user 'pi' at the graphical desktop" come visibile in Fig. 6.



Fig. 8

Fig. 7

Poi, utilizzando lo strumento WinSCP, aprire il file /etc/xdg/lxsession/LXDE-pi/autostart e configuratelo come segue (Fig. 7):

```
# Auto run the browser
xset s off
xset -dpms
xset s noblank
midori -e Fullscreen -a http://<indirizzo-IP-Raspberry>
```

Dopo un reboot vedrete partire Raspberry Pi ed al termine apparire la pagina home dell'applicazione web di gestione dell'applicazione UDOiT come visibile in Fig. 8.

Se si desidera uscire dall'applicazione e tornare al prompt di login, dalla tastiera senza fili basta premere la combinazione di tasti:

CTRL + ALT + F1

oppure

CTRL + ALT + F2

Un'ultima considerazione sullo schermo touch del display PI, uno schermo capacitivo a 10 punti, ovvero in grado di riconoscere contemporaneamente dieci punti di contatto. Avendo solo dieci dita direi che è più che sufficiente. Il driver di questo schermo gestisce sia la simulazione di un mouse standard che gli eventi associati ai contatti. È in grado di interfacciarsi al server X come un mouse, anche se il server X stesso non è stato progettato per funzionare con uno schermo touch. Un minimo di pratica permetterà un utilizzo soddisfacente per la applicazione UDOiT. Per spegnere Raspberry Pi basta "tappare" sulla entrata "shutdown" nel menu di comando.



per il MATERIALE

I prodotti presentati in questo progetto possono essere acquistati presso Futura Elettronica. La Raspberry Pi 2 Tipo B (cod. 8326274RS) è disponibile al prezzo di 44,00 Euro, il display Raspberry Pi LCD 7" TOUCH (cod. 8997466RS) è in vendita a Euro 85,00. Tutti 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

Galileo 11 e 12 regolarmente in orbita

Anche Galileo 11 e 12 sono entrati regolarmente in orbita dopo il lancio del 17 dicembre 2015 avvenuto dalla base ESA dalla Guyana Francese da parte di un vettore Soyuz. Con questo lancio il numero dei satelliti della costellazione europea Galileo è raddoppiato negli ultimi nove mesi, potendo contare ad oggi ben 12 satelliti in orbita.

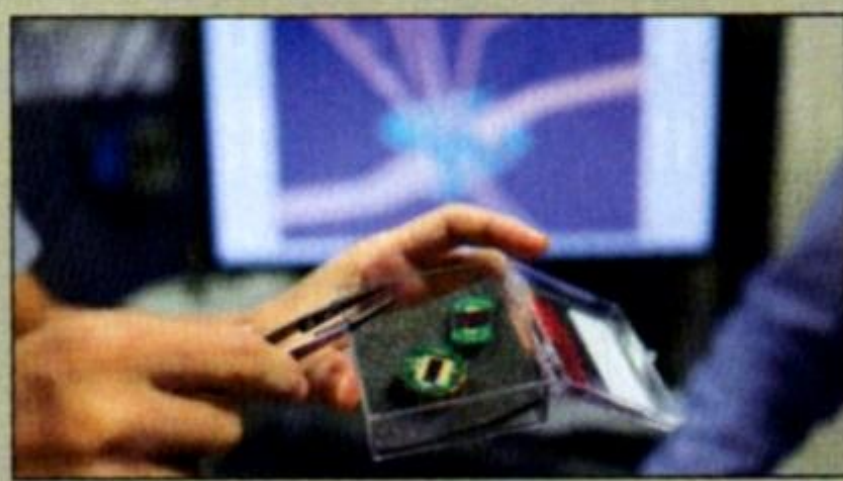
Questo sesto lancio di Galileo è stata un'operazione da manuale con tutte le fasi portate a termine nei tempi previsti, con lo stadio superiore Fregat che ha posizionato nell'orbita a 23.500 km di altezza i due satelliti, 3 ore e 48 minuti dopo il liftoff.

Il prossimo anno la campagna di lanci (che a regime dovrà portare a quota 30 satelliti la costellazione) si fermerà per lasciare spazio ad un'attività di test, in vista dell'avvio ufficiale dell'operatività con la fase denominata "Initial Services", che è l'obiettivo posto dalla Commissione Europea per la fine del 2016.

I lanci riprenderanno nella seconda metà del 2016 con un vettore Ariane 5 in grado di portare in orbita con ciascun lancio ben 4 satelliti invece di due.

www.esa.int

Col grafene sensori magnetoresistivi 200 volte più sensibili rispetto al silicio



Tra i più usati sia in ambito domestico che industriale, i sensori magnetoresistivi (MR sensor) sono in grado di rilevare variazioni di livello nei campi magnetici dovute, nella maggior parte dei casi, allo spostamento, all'interno del campo stesso, di materiali ferromagnetici; risulta così possibile misurare, ad esempio, la velocità di rotazione tramite ruote dentate. Tradizionalmente i sensori magnetoresistivi sono fabbricati utilizzando il silicio. È però possibile impiegare altri materiali, come hanno dimostrato i ricercatori della National University of Singapore (NUS) che hanno realizzato dei sensori magnetoresistivi facendo ricorso al grafene e al nitrato di boro, riuscendo

ad ottenere una sensibilità di circa 200 volte superiore rispetto al silicio. Nei loro studi – presentati recentemente sulla rivista *Nature Communications* – i ricercatori della NUS hanno utilizzato il nitrato di boro come substrato per sottili fogli di grafene; il chip risultante realizza un'interfaccia che consente agli elettroni di passare attraverso il materiale molto rapidamente. Il chip possiede un'elevata sensibilità sia ai campi magnetici di alta che di bassa intensità, e la sua resistenza è insensibile alla temperatura, così come alla tensione di alimentazione la cui scelta consente di impostare il punto di lavoro ottimale. Altra caratteristica molto interessante dei prototipi al grafene realizzati alla NUS riguarda la massima temperatura di funzionamento che, con la tradizionale tecnologia al silicio, non supera i 127° C, mentre nel caso del grafene è possibile andare oltre. Non solo, all'aumentare della temperatura la sensibilità migliora, tanto che a 127°C è ben otto volte superiore rispetto a quella ambiente.

<http://phys.org>

Il laser che raffredda l'acqua

Il laser non si limita a scaldare o a bruciare, ma può anche raffreddare ad esempio l'acqua, come dimostrato in un esperimento nel quale per la prima volta dei fasci concentrati di luce infrarossa hanno portato un bicchiere d'acqua dalla temperatura ambiente a poco più di 2 gradi.

L'esperimento è stato pubblicato sulla rivista dell'Accademia delle Scienze degli Stati Uniti (Pnas). Il test è stato condotto nell'università di Washington. Si intravedono già interessanti applicazioni nei processi industriali che richiedono grande precisione, come nella fabbricazione dei microprocessori o in biologia, nello studio sempre più in profondità delle cellule. Infatti il vantaggio base offerto dalla refrigerazione laser rispetto alle altre tecnologie criogeniche, è senza dubbio la precisione, intesa in termini spaziali.

Annabell, il cervello artificiale che ha imparato a parlare

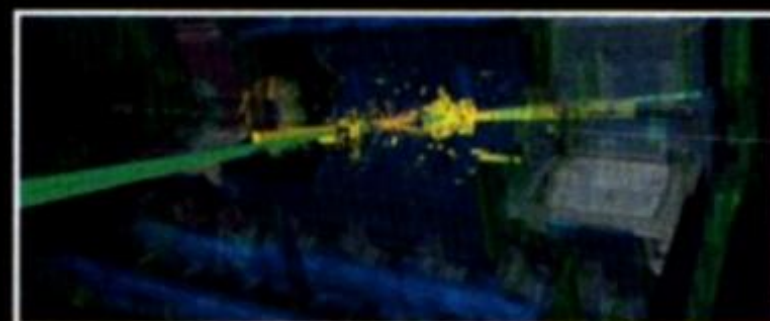


Un gruppo di ricercatori dell'Università di Sassari e dell'Università britannica di Plymouth ha sviluppato per la prima volta un modello cognitivo, costituito da 2 milioni di neuroni artificiali connessi tra loro, capace di imparare a dialogare usando il linguaggio umano partendo da una condizione di "tabula rasa", solo attraverso la comunicazione con un interlocutore umano. Il modello chiamato Annabell (Artificial Neural Network with Adaptive Behavior Exploited for Language Learning) è stato descritto in un articolo pubblicato sulla rivista scientifica internazionale PLOS ONE e presentato alla conferenza internazionale BICA 2015. La ricerca getta luce sui processi neurali che sono alla base dello sviluppo del

linguaggio. Annabell non ha conoscenze linguistiche precodificate, impara a parlare esclusivamente attraverso la comunicazione con un interlocutore umano attraverso due meccanismi fondamentali, presenti anche nel cervello biologico: la plasticità sinaptica e il gating neurale. Questo meccanismo è fondamentale per l'apprendimento e per la memoria a lungo termine. Il gating neurale si basa sulla proprietà di certi neuroni, detti bistabili, di comportarsi come dei cancelli (in inglese gate) che possono essere aperti o chiusi da un segnale di controllo proveniente da altri neuroni.

www.uniss.it

Cern: nuovo record di energia per il Large Hadron Collider



All'interno del Large Hadron Collider (Lhc) del Cern di Ginevra sono state ottenute collisioni di particelle ad energie da record, mai raggiunte finora da nessuna macchina.

L'acceleratore di particelle ha raggiunto l'energia record di 13.000 miliardi di elettronvolt (TeV), completando la fase di collisioni fra protoni ed al suo interno sono cominciati a scorrere i primi fasci stabili di ioni di piombo.

I quattro esperimenti dell'acceleratore (Cms, Atlas, Alice e per la prima volta Lhcb) raccoglieranno per un mese i dati sulle nuove collisioni.

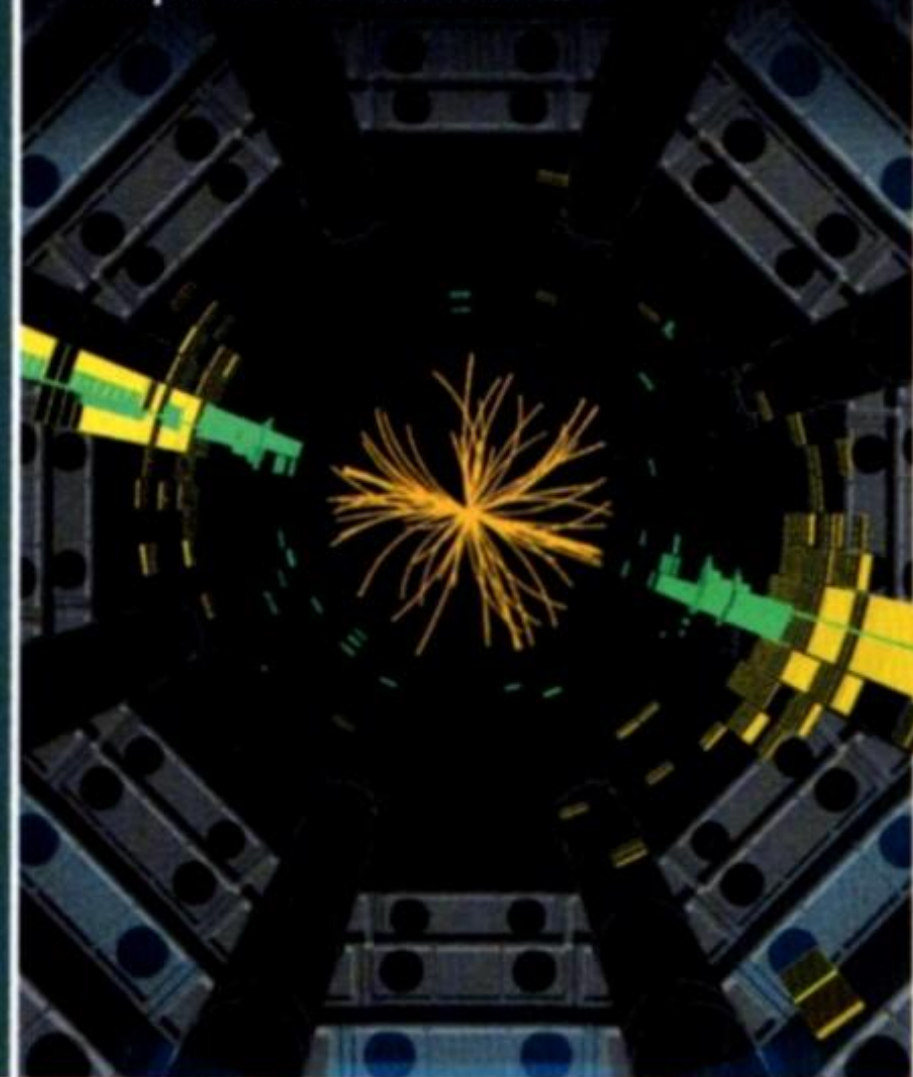
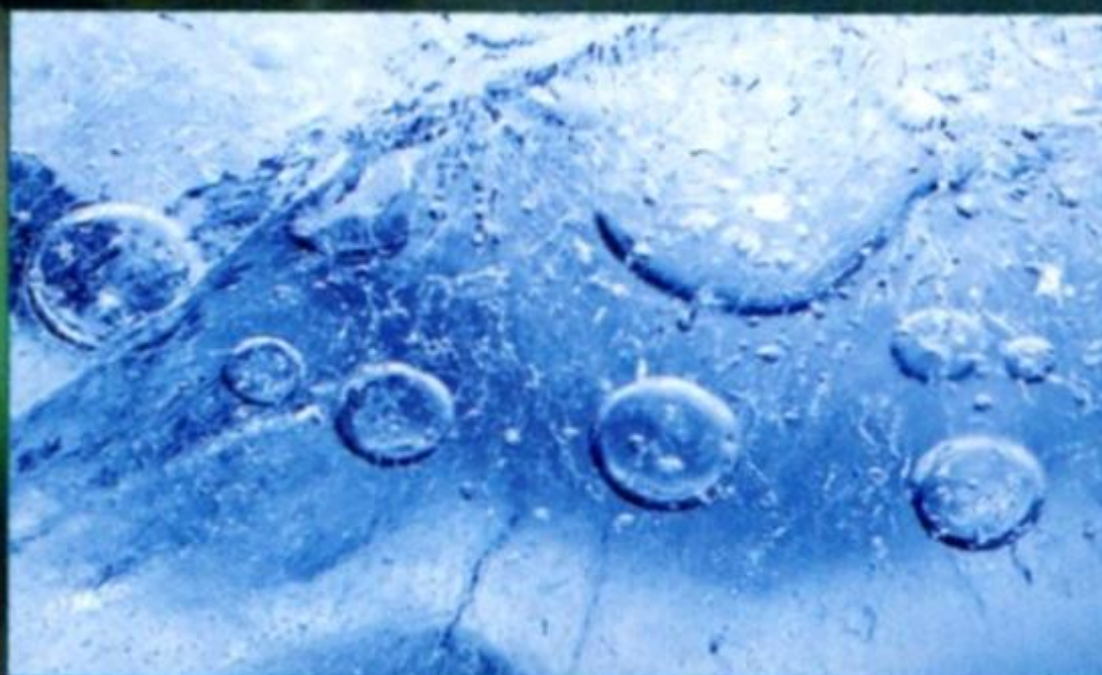
"È tradizione far collidere gli ioni per circa un mese ogni anno come parte del programma di ricerca di Lhc, ma quest'anno raggiungiamo una nuova energia che consente di esplorare la materia a uno stadio molto precoce del nostro universo", spiega il direttore generale del Cern, Rolf Heuer.

Le collisioni di ioni consentiranno ai fisici del Cern di studiare plasma di quark e gluoni: un insieme di particelle esistito pochi milionesimi di secondo dopo il Big Bang.

<http://home.web.cern.ch>

Le potenzialità di raffreddamento in determinate condizioni dei raggi laser erano note dal 1995, in seguito ad un esperimento condotto nei Laboratori Usa di Los Alamos, ma solamente ora è stata dimostrata tale possibilità anche nei liquidi. La teoria dietro a tale risultato non è nuova, risale al 1929, quando in un articolo pubblicato su Zeitschrift für Physik il fisico Peter Pringsheim suggerì che la cosiddetta fotoluminescenza anti-Stokes poteva ridurre la temperatura del materiale illuminato.

www.washington.edu



L'auto senza pilota made in Cina

Le self-driving car iniziano a diffondersi anche a livello globale e, dopo Google, anche Baidu, il motore di ricerca cinese, ha presentato il proprio progetto legato alla guida autonoma. La vettura testata è stata realizzata con un sistema di videocamere e sensori per mappare l'ambiente circostante ed evitare gli ostacoli mentre il veicolo è stato ottenuto modificando una BMW Serie 3.

I primi test realizzati su strada si sono svolti senza problemi tra le strade di Pechino per 30 chilometri, raggiungendo una velocità massima di 100 chilometri all'ora.

L'automobile senza pilota di Baidu si è districata in maniera agile tra strade secondarie e autostrade. Il progetto è stato avviato nel 2013, ma il debutto sul mercato non

è ancora stato programmato.

L'azienda cinese ha l'obiettivo di introdurre la tecnologia soprattutto nel trasporto pubblico, lanciando gli autobus senza conducente.

www.technologyreview.com



Marte è più vicina: Falcon 9 rientra dallo spazio e atterra con successo

Dopo due tentativi falliti, questa volta il razzo Falcon 9 è riuscito nell'impresa di rientrare da dove era partito, atterrando con precisione e senza danni, dopo aver messo in orbita 11 minisatelliti Orbcomm. Space X raggiunge così il più importante obiettivo che Elon Musk, il suo visionario fondatore, si era posto: quello di ridurre drasticamente il costo dei lanci spaziali mediante il recupero e il riutilizzo del razzo vettore, l'elemento che incide maggiormente sui costi di lancio. Con l'obiettivo finale di riuscire a realizzare una colonia su Marte. In pochi anni la società di Elon Musk (è stata fondata nel 2002) ha cambiato radicalmente l'approccio alle tecnologie spaziali introducendo i concetti di modularità e riutilizzo che hanno portato alla realizzazione di un motore, il Merlin, utilizzato in dieci elementi per fornire la spinta al vettore Falcon, consentendogli il rientro a terra con un atterraggio morbido e controllato.

Ed è proprio nella serata di lunedì 21 dicembre 2015 che tutto ciò è avvenuto per la prima volta, con la partenza da Cape Canaveral per la messa in orbita di 11 satelliti Orbcomm e il rientro del primo stadio del vettore Falcon 9 dieci minuti dopo la partenza. L'atterraggio è avvenuto in un'apposita piazzola nell'area di Cape Canaveral: l'enorme vettore (quasi 50 metri di altezza) si è fermato in posizione verticale appoggiandosi su quattro "zampe" che al momento del lancio erano nascoste nel corpo del vettore; attualmente sono in corso le operazioni di controllo e recupero dei motori e della struttura. Questo traguardo dimostra anche la fattibilità di un atterraggio controllato su Marte da parte di un razzo da utilizzare in seguito per il volo di ritorno sulla Terra (ricordiamo che la massa di Marte è pari al 10% di quella terrestre).

www.spacex.com

Xenon 1T: trappola per la materia oscura

Presso i Laboratori Nazionali del Gran Sasso (LNGS) dell'Istituto Nazionale di Fisica Nucleare (INFN) inizia la caccia alla materia oscura con XENON1T, un esperimento che coinvolge 21 gruppi di ricerca di tutto il mondo. "Illuminare l'oscurità" è lo slogan scelto dai ricercatori di XENON1T, che ha l'ambizioso obiettivo di fare luce su uno dei misteri della fisica contemporanea: di che cosa sia fatta la materia oscura che costituisce circa un quarto dell'universo.

E l'esperimento che hanno realizzato è una vera e propria trappola per la materia oscura.

Per catturare le particelle di materia oscura, i fisici di XENON1T hanno bisogno di condizioni particolari. A partire dal luogo in cui ospitare l'esperimento: le viscere del Gran Sasso, che schermano dai raggi cosmici, preservando così quello che i fisici chiamano "silenzio cosmico". Ma la schermatura della sola montagna non è sufficiente. Anche l'esperimento dev'essere realizzato in modo idoneo a rivelare i rarissimi eventi di interazione della materia oscura con quella ordinaria. I fisici di XENON1T hanno scelto di utilizzare un gas nobile ultrapuro, lo xenon per l'appunto, raffreddato a una temperatura molto bassa, -95 °C, per mantenerlo allo stato liquido. Il rivelatore (la Camera a Proiezione di Tempo, TPC), cuore dell'esperimento, è in grado di dare un segnale quando le particelle interagiscono al suo interno. Il rivelatore è immerso in un criostato, un thermos, in acciaio inossidabile a bassa radioattività, contenente circa 3500 kg di xenon liquido.

www.infn.it





>



4

SVILUPPARE APP android CON Processing

dell'ing.
MIRCO SEGATELLO

Impariamo a scambiare dati tra uno smartphone e Arduino, utilizzando la connessione Bluetooth. Quarta puntata.

In questa quarta lezione vi spiegheremo come far comunicare uno smartphone Android con una board Arduino e lo faremo utilizzando il modulo Bluetooth interno al telefono. Per poter mettere i due dispositivi in comunicazione sarà necessario dotare Arduino di un opportuno modulo Bluetooth correttamente configurato. Per facilitare la realizzazione di quest'applicazione utilizzeremo i due moduli Bluetooth più diffusi in commercio, ovvero il modulo RN-42 della Roving Network ed il BTBee della Seedstudio (codice WLS125E1P) basato sul chip CSR Bluecore04. Entrambi i moduli sono ugualmente validi e offrono le funzioni fondamentali per operare nella nostra applicazione; la loro ampia diffusione commerciale permette di trovare delle soluzioni hardware molto comode come degli

shield compatibili Arduino. Ad esempio, in commercio è facilmente reperibile il modulo HC-05, anch'esso facilmente interfacciabile con Arduino, del quale sul web si trovano parecchi esempi di utilizzo.

I moduli Bluetooth si dividono in tre classi a seconda della potenza di trasmissione, dalla quale dipende anche la massima distanza di lavoro; a riguardo consultate la **Tabella 1**, dove vedete riassunte le caratteristiche essenziali dei moduli Bluetooth presi in considerazione. In una tipica applicazione, una periferica (nel nostro caso Arduino, che si comporta da slave) deve risultare rilevabile dagli altri dispositivi (*discoverable device*) ovvero "visibile" e deve essere in attesa di venire contattata. Il dispositivo master (lo smartphone) invia una richiesta di ricerca a

Tabella 1 - Classi Bluetooth.

Classe	Potenza (mW)	Potenza (dBm)	Distanza (m)
Classe 1	100	20	~ 100
Classe 2	2,5	4	~ 10
Classe 3	1	0	~ 1

tutte le periferiche presenti nel suo campo di portata; tutte le periferiche contattate rispondono comunicando il loro indirizzo.

La periferica master avvia la sincronizzazione con il dispositivo scelto secondo una tecnica, detta *paging* e, tramite il protocollo SDP (Service Discovery Protocol), la master scopre tutti i servizi resi disponibili dalla slave. A conclusione della fase di scoperta dei servizi, la periferica master è in grado di creare un canale di comunicazione con il punto di accesso, utilizzando il protocollo L2CAP.

Alcuni dispositivi integrano un meccanismo di sicurezza, detto accoppiamento (pai-

Fig. 1 - Modulo Bluetooth RN-42.



Fig. 2- Modulo BluetoothBee.



ring, in inglese...), che permette di limitare l'accesso ai soli utilizzatori autorizzati, per garantire un certo livello di protezione della comunicazione. Lo slave invia quindi una richiesta di *pairing* alla periferica master con la conseguente richiesta di inserimento del codice PIN (acronimo di Personal Information Number) da parte dell'utente; se il PIN è corretto la periferica viene associata e la comunicazione può avvenire.

La procedura di ricerca ed associazione dei dispositivi può richiedere molto tempo; per questo il nostro smartphone memorizza in una lista tutti i dispositivi (e i relativi parametri di funzionamento compreso il PIN, se esiste) che sono stati associati, velocizzando la connessione con un dispositivo già in elenco.

Per facilitare l'utilizzo dei dispositivi Bluetooth è stata definita una serie di profili, che identificano una serie di possibili applicazioni. I principali profili sono i seguenti:

- **Advanced Audio Distribution Profile (A2DP):** profilo di trasmissione audio avanzata;
- **Audio Video Remote Control Profile (AVRCP):** profilo di telecomando multimediale;
- **Cordless Telephony Profile (CTP):** profilo di telefonia senza fili;
- **File Transfer Profile (FTP):** profilo di trasferimento di file;
- **Generic Access Profile (GAP):** profilo d'accesso generico;
- **Hands-Free Profile (HFP):** vivavoce;
- **Human Interface Device Profile (HID):** profilo d'interfaccia uomo-terminale;
- **Headset Profile (HSP):** profilo auricolare;
- **Personal Area Networking Profile (PAN):** profilo di rete personale;
- **Service Discovery Application Profile (SDAP):** profilo di scoperta delle applicazioni disponibili;
- **Synchronization Profile (SP):** profilo di sincronizzazione con un gestore delle informazioni personali detto PIM (Personal Information Manager);

Tabella 2 - Caratteristiche dei moduli Bluetooth commerciali.

Modulo	Caratteristiche
RN-42	Produttore Roving Networks Protocollo Bluetooth: specifiche Bluetooth v2.0 compatibile Potenza di uscita: Classe 2 Compatibile con tutti i prodotti Bluetooth che supportano SPP Include i supporti per i protocolli BCSP, DUN, LAN, GAP, SDP, RFCOMM e L2CAP UART completamente configurabile Velocità di trasferimento dati UART: fino a 3 Mbps Velocità trasmissione dati: da 721 kbps a 2 Mbps Basso consumo in modalità sleep Funzionamento a 3,3 V cc Certificato CE, FCC e RoHS
RN-41	Come modulo RN-42 ma in classe 1
CSR Bluecore04	Enhanced Data Rate (EDR) compatibile per entrambe le modalità di modulazione 2Mbps e 3Mbps Soluzione single-chip che integra radio, baseband, microcontrollore Potenza di uscita: Classe 2 Interfaccia per flash memory esterna (8Mbits) Bluetooth v2.0+sistema EDR (v2.1 aggiornabile) Bluetooth full speed funzionante con Piconet Compatibile con supporto Scatternet Compatibile con SCO (eSCO) Porte USB e dual UARTs Compatibile con standard 802.11
HC-05	Protocollo Bluetooth: specifiche Bluetooth v2.0 + EDR Modulazione 3Mbps Potenza di uscita: Classe 2 Frequenza: banda ISM a 2,4 GHz Modulazione: GFSK (Gaussian Frequency Shift Keying) Potenza di emissione: 4dBm, Class 2 Sensibilità: -84dBm a 0,1% BER Velocità: - Asincrono: 2.1Mbps(Max) / 160 kbps - Sincrono: 1Mbps/1Mbps Sicurezza: autenticazione e crittografia Interfaccia: UART con baud rate programmabile Alimentazione: 3,3 Vcc - 50 mA

- Serial Port Profile (SPP): profilo porta seriale (utilizzato per la comunicazione dati, ad esempio quando si sfrutta un dispositivo Bluetooth come interfaccia con un modem GSM).

Per la nostra applicazione è necessario che il modulo BT sia in modalità rilevabile all'accensione e predisposto per una comunicazione secondo il profilo SPP, ovvero adatto allo scambio di dati come in una comunicazione seriale cablata.

Per il nostro primo esempio utilizzeremo lo shield per Arduino basato sul modulo RN-42 della Roving Networks, opportunamente programmato allo scopo; l'RN-42 si può acquistare presso la Futura Elettronica (www.futurashop.it, contraddistinto dal codice prodotto 7300-RN42) in versione con adattatore socket (7100-FT1018M) oppure montato nello shield compatibile Arduino (codice 7100-FT1032M); il modulo Bluetooth su socket Xbee (BlueToothBee) si può acquistare anch'esso da Futura Elettronica (codice 7300-BLUETBEE).

Prima di utilizzare il modulo è opportuno che vengano ripristinati i valori di fabbrica tramite il dip-switch presente sullo shield serigrafato *default* (tutti i dip devono essere posizionati su OFF). Portate il dip-switch su ON e alimentate lo shield, quindi portatelo sullo stato OFF, poi su ON e di nuovo OFF attendendo almeno un secondo tra una transizione e l'altra; così lo shield è predisposto per far comunicare il modulo RN-42 con Arduino tramite la seriale hardware, la quale pertanto non può essere usata per far comunicare Arduino con il PC. Per tale motivo, se proverete a programmare Arduino con lo shield installato vi comparirà un messaggio di errore. Per consentire un facile *debug* del modulo dal PC, preferiamo mettere in comunicazione il modulo RN-42 con il modulo FDTI e poterlo così gestire da PC; allo scopo staccate lo shield da Arduino e connettetelo tramite dei jumper come descritto nella **Tabella 3**. Su Arduino caricate un semplicissimo sketch che imposta i pin 0 e 1 come ingressi, ovvero pone ad alta impedenza i due pin del modulo UART.

Passiamo ora alla fase di programmazione del modulo: avviate l'IDE di Arduino ed

Tabella 3- Connessioni tra Arduino e BT shield

Arduino	BT shield RN-42
RX - pin 0	TX - pin 1
TX - pin 1	RX - pin 0
+3,3V	+3,3V
GND	GND

Tabella 4- Principali comandi di configurazione del modulo RN-42.

Comando	Funzione
\$\$\$	Entra in modalità comando
SF,1<CR>	software reset
SM,0<CR>	slave mode (già di default)
SN,Arduino<CR>	imposta il nome in Arduino
SP,1234<CR>	pin code a 1234 (già di default)
SU,115<CR>	baud rate a 115000 (già di default)
S~,0<CR>	Imposta profilo SPP (già di default)
+++<CR>	echo ON (già di default)
--<CR>	Esce dalla modalità comando
D<CR>	Riporta le impostazioni base
E<CR>	Riporta le impostazioni avanzate
M<CR>	Riporta lo stato del segnale
GB<CR>	Riporta l'indirizzo del modulo
GK<CR>	Riporta lo stato della connessione
V<CR>	Riporta la versione del firmware

* La scritta <CR> indica che la stringa ha come ultimo carattere il carattere speciale "ritorno carrello".

aprite SerialMonitor settando la comunicazione a una velocità di 115 kbps e la modalità di invio stringa con "Nessun fine riga". Inviare la stringa \$\$\$ (tre caratteri dollaro) per far entrare il modulo BT in modalità comando, quindi impostate la modalità di invio stringa con "Ritorno Carrello" ed inviate la stringa "SN,Arduino" per modificare il nome del modulo. Tutti i rimanenti parametri devono essere lasciati ai valori predefiniti. Ora inviate il carattere D per ottenere i valori di configurazione; in definitiva, avrete il modulo impostato come visibile in **Fig. 5**. Per comodità riportiamo i principali comandi nella **Tabella 4**.

È importante interpretare lo stato dei LED per sapere lo stato della connessione, il quale viene così segnalato:

- LED 2 giallo fisso + LED1 rosso lampeggio lento = non connesso;
- LED 2 giallo fisso + LED1 rosso lampeggio veloce = dispositivo rilevato e disponibile (PAIRED);
- LED 1 rosso acceso fisso = connesso.

Per verificare se il nostro modulo funziona correttamente possiamo fare una prova

Fig. 3 - BT shield connesso ad Arduino.

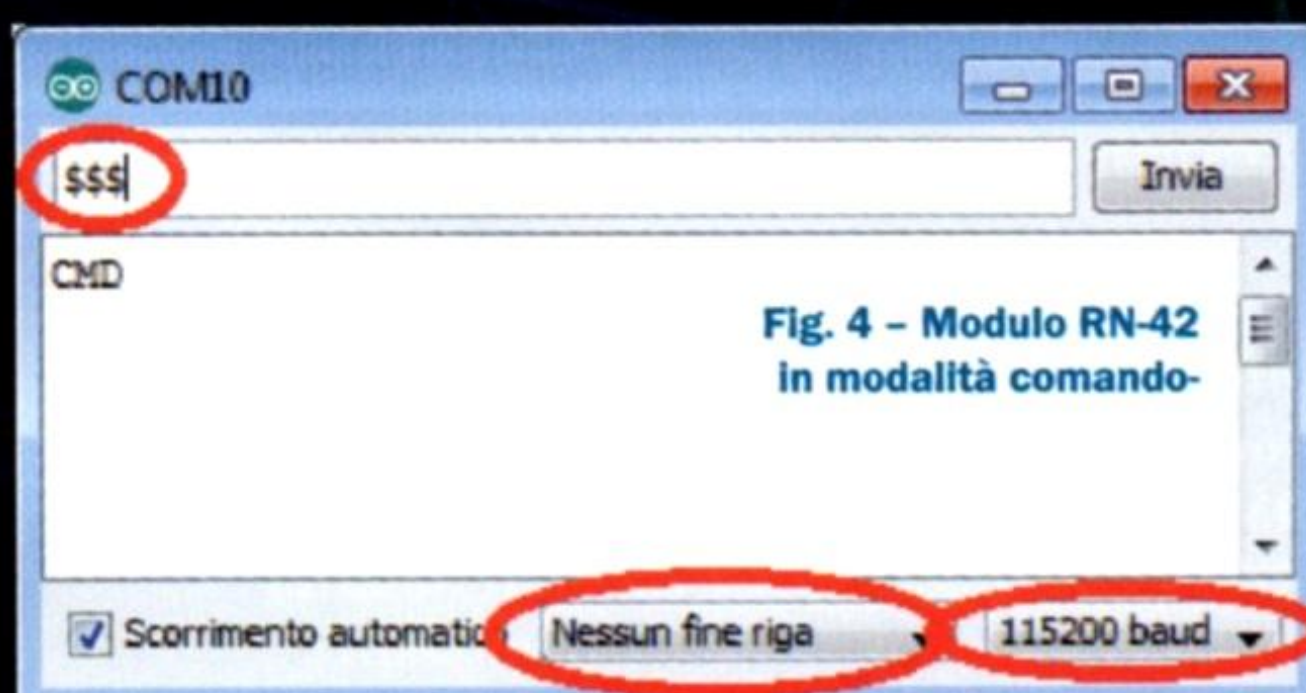
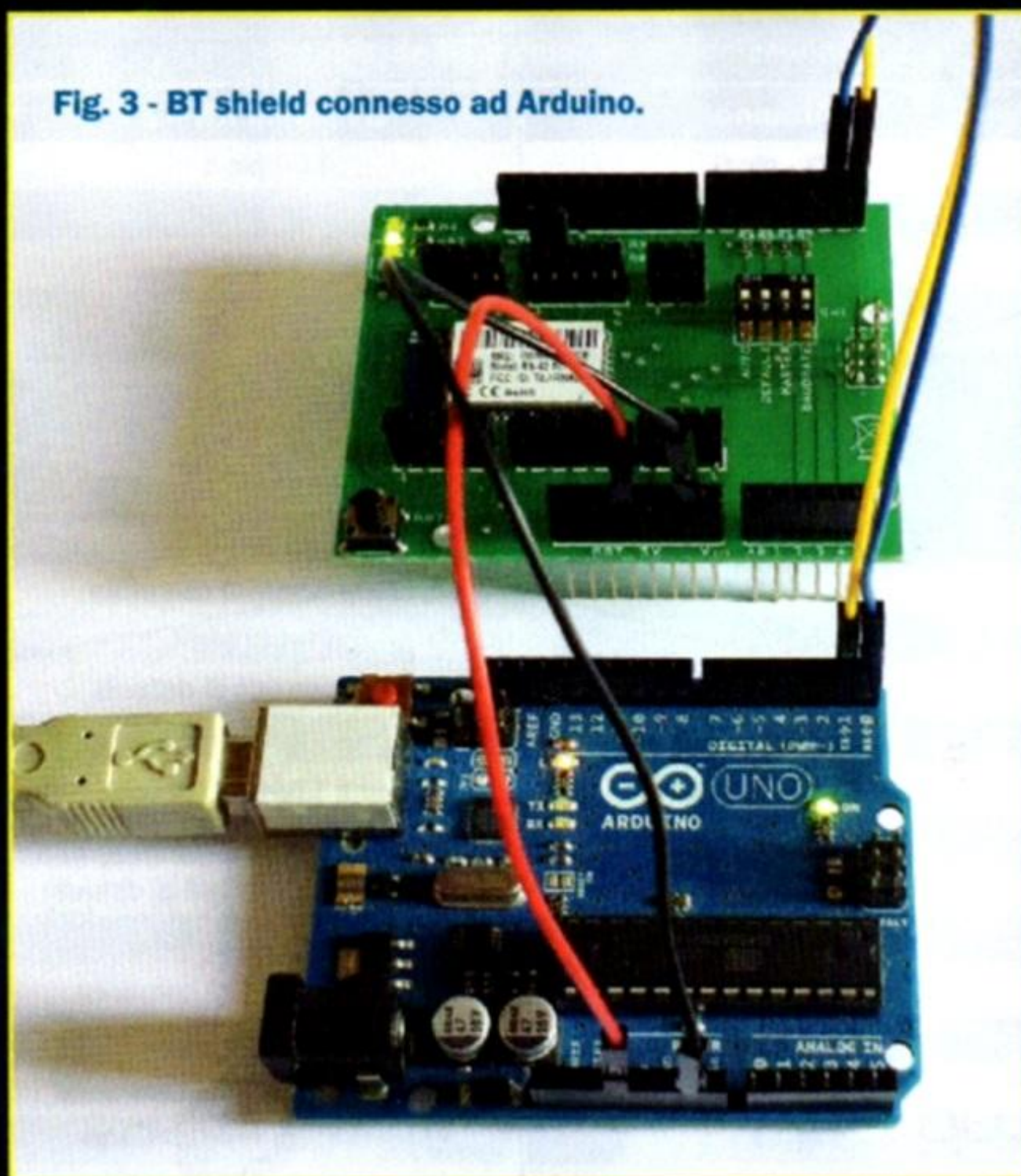


Fig. 4 - Modulo RN-42 In modalità comando-

installando sul nostro smartphone l'applicazione di nome *bluetooth SPP pro* (o app simili) che permette di testare la nostra connessione. Andate nelle impostazioni Bluetooth dello smartphone, avviate la ricerca dei dispositivi ed associatevi con Arduino, quindi avviate l'applicazione Bluetooth SPP pro ed impostate la modalità "CMD line mode". Ora Arduino è connesso

al PC e Serial Monitor è in esecuzione con impostata la modalità di invio stringhe con "sia NL che CR (<NL><CR>"); adesso potete interagire inviando e ricevendo stringhe da e verso il PC tramite lo smartphone. Analogamente potete utilizzare il modulo BTBee della Seeedstudio; tutti i dettagli per la sua programmazione, compreso lo sketch per Arduino (*Bluetooth_demo_code.zip*), sono disponibili all'indirizzo: http://www.seeedstudio.com/wiki/Grove_-_Serial_Bluetooth. Il modulo comunica, per impostazione predefinita, a una velocità di 38.400 Baud ed il PINcode vale "0000".

Sono disponibili due LED di segnalazione utili per monitorare lo stato di funzionamento della connessione. Nello stato sconnesso si vedrà il solo LED verde lampeggiare 2 volte al secondo; in attesa di collegamento (pairing mode) ci saranno sia il LED verde che il LED rosso lampeggiare 1 volta al secondo e nello stato connesso lampeggerà il solo LED verde una volta ogni due secondi.

Nel **Listato 1** riportiamo la parte dello sketch per Arduino che si occupa di settare i parametri interni del modulo; queste righe dovranno essere eseguite ad ogni accensione del modulo. In entrambi i casi disponiamo di un modulo configurato e funzionante i cui dati ricevuti e spediti sono gestiti da Serial-Monitor di Arduino.

Possiamo ora occuparci dell'applicazione per Android: le API di Android forniscono tutti i metodi per l'accesso alla periferica Bluetooth e in rete si trovano diversi esempi facilmente adattabili alla propria esigenza; per rendere fruibili gli esempi anche ai meno esperti di programmazione, ci appoggeremo alla libreria Ketai, la quale fornisce tutti gli strumenti per imparare rapidamente ad accedere alla periferica. Tra gli esempi forniti assieme alla

Listato 1

```
blueToothSerial.begin(38400); //Set BluetoothBee BaudRate to default baud rate 38400
blueToothSerial.print("\r\n+STWMOD=0\r\n"); //set the bluetooth work in slave mode
blueToothSerial.print("\r\n+STNA=Arduino\r\n"); //set the bluetooth name as "Arduino"
blueToothSerial.print("\r\n+STPIN=0000\r\n"); //Set SLAVE pincode "0000"
blueToothSerial.print("\r\n+STOAU=1\r\n"); // Permit Paired device to connect me
blueToothSerial.print("\r\n+STAUTO=0\r\n"); // Auto-connection off
delay(2000); // This delay is required.
blueToothSerial.print("\r\n+INQ=1\r\n"); //make the slave bluetooth inquirable
Serial.println("The slave bluetooth is inquirable!");
delay(2000); // This delay is required.
```


libreria troverete lo sketch di nome BluetoothCursors.pde, che dimostra l'utilizzo dei comandi per la gestione di una semplice comunicazione tramite Bluetooth. Da questo esempio abbiamo ricavato il listato dell'ES10 nel quale abbiamo implementato l'invio e la ricezione di semplici stringhe (**Esercizio 10**). La struttura del programma è intuitiva e consiste nell'implementare una macchina a stati a seconda dello stato della connessione BT. Nello stato iniziale status=0 il Bluetooth dello smartphone, che funge da master, cerca di scoprire i dispositivi a portata. Se tra i dispositivi viene trovato Arduino si passa allo stato 2 altrimenti si passa allo stato 1 e l'utente viene informato che non è stato trovato. Nello stato 2 si avvia la connessione con il modulo Arduino e se va a buon fine si passa allo stato 3, in cui la comunicazione è stabilita. Avremmo potuto (come nell'esempio Ketai) rilevare tutti i dispositivi e lasciare all'utente l'onere di selezionare a quale connettersi, ma, nel nostro caso, sarebbe stato del tutto inutile connettersi ad un dispositi-



Fig. 5 - Settaggio modulo RN-42.

Listato 2

```
/* ES10 invio ricezione stringhe via BT
Basato su libreria KETAI
*/

import android.content.Intent;
import android.os.Bundle;
import ketai.net.bluetooth.*;
import ketai.ui.*;
import ketai.net.*;

PFont fontMy;
KetaiBluetooth bt;
int status = 0;
String send, received = "Received: ";
char input;

// The following code is required to enable bluetooth at startup.
void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    bt = new KetaiBluetooth(this);
}

void onActivityResult(int requestCode, int resultCode, Intent data) {
    bt.onActivityResult(requestCode, resultCode, data);
}

void setup() {
    size(displayWidth, displayHeight);
    frameRate(10); //velocità con cui viene ripetuto void draw()
    orientation(PORTRAIT); //verticale
    background(0); //uso display nero
    fontMy = createFont("SansSerif", 60);
    textFont(fontMy);
    //start BT connections
    bt.start();
    // avvio scoperta dispositivi (discover devices)
    bt.discoverDevices();
}

void draw() {

    fill(80,90,100); //riempimento rettangolo
    stroke(255); //colore del bordo bianco
    rect(0, 0, width, 200);
    fill(200); //colore del testo

    if (status==0) //sto cercando Arduino (is discovering)
    {
        text("Discovering devices...", 5, 100);
        ArrayList<String> names;
        names = bt.getDiscoveredDeviceNames();
        for (int i=0; i < names.size(); i++)
        {
            String device=names.get(i);
            println("[ "+i+" ] " + device);
            if (device.equals("Arduino")) //in elenco c'è Arduino
            {
                bt.connectToDeviceByName("Arduino"); //si connette al
                dispositivo selezionato
                status=2;
            }
        }
        if (!bt.isDiscovering()) //ho cercato ma non ho trovato
        {
            status=1;
        }
    }
    if (status==1)
    {
        text("Arduino no Discover", 5, 100);
    }
    if (status==2)
    {
        text("Try to connect Arduino...", 5, 100);
        if (bt.connectToDeviceByName("Arduino"))
        {
            fill(90,80,70); //riempimento rettangolo
            stroke(255); //colore del bordo bianco
            rect(0, 200, width, 200);
            fill(200); //colore del testo
            textAlign(CENTER);
            text("Send", width/2, 300);
            status=3;
        }
    }
    if (status==3)
    {
        textAlign(LEFT);
        text("Connected to Arduino", 5, 100);
        //Visualizza i dati ricevuti
        text(received, 0, 600, width, height/2);
    }
}
```

(Continua)

Listato 2 (segue)

```

void mousePressed() {
  if (mouseY > 200 && mouseY < 400 && status==3)
  {
    send= "ciao"; //invia una stringa
    println("send:" + send);
    byte[] data = send.getBytes(); //converte la stringa in un array di byte
    //Invia la stringa
    bt.broadcast(data);
  }
}

//Funzione richiamata quando arrivano delle stringhe da Arduino
void onBluetoothDataEvent(String who, byte[] data) {
  if (status==3)
  {
    //le stringhe vengono accodate (conversione da array di byte a stringa)
    received += new String(data);
    //clean if string to long
    if (received.length() > 175)
      received = "Message to long!";
  }
}

```

Listato 3

```

/* ES11 controllo via Android di un LED
uso BTshield con RN-42
*/

int LED = 13;
byte dataIn;

void setup()
{
  pinMode(LED, OUTPUT);
  // mette in alta impedenza i pin TX e RX.
  pinMode(0, INPUT);
  pinMode(1, INPUT);
  Serial.begin(115000);
}

void loop()
{
  if (Serial.available())
  {
    dataIn = Serial.read();
    if (dataIn == 'x')
    {
      digitalWrite(LED, HIGH);
      Serial.println("LED ON");
    }
    if (dataIn == 'y')
    {
      digitalWrite(LED, LOW);
      Serial.println("LED OFF");
    }
  }
}

```

Listato 4

```

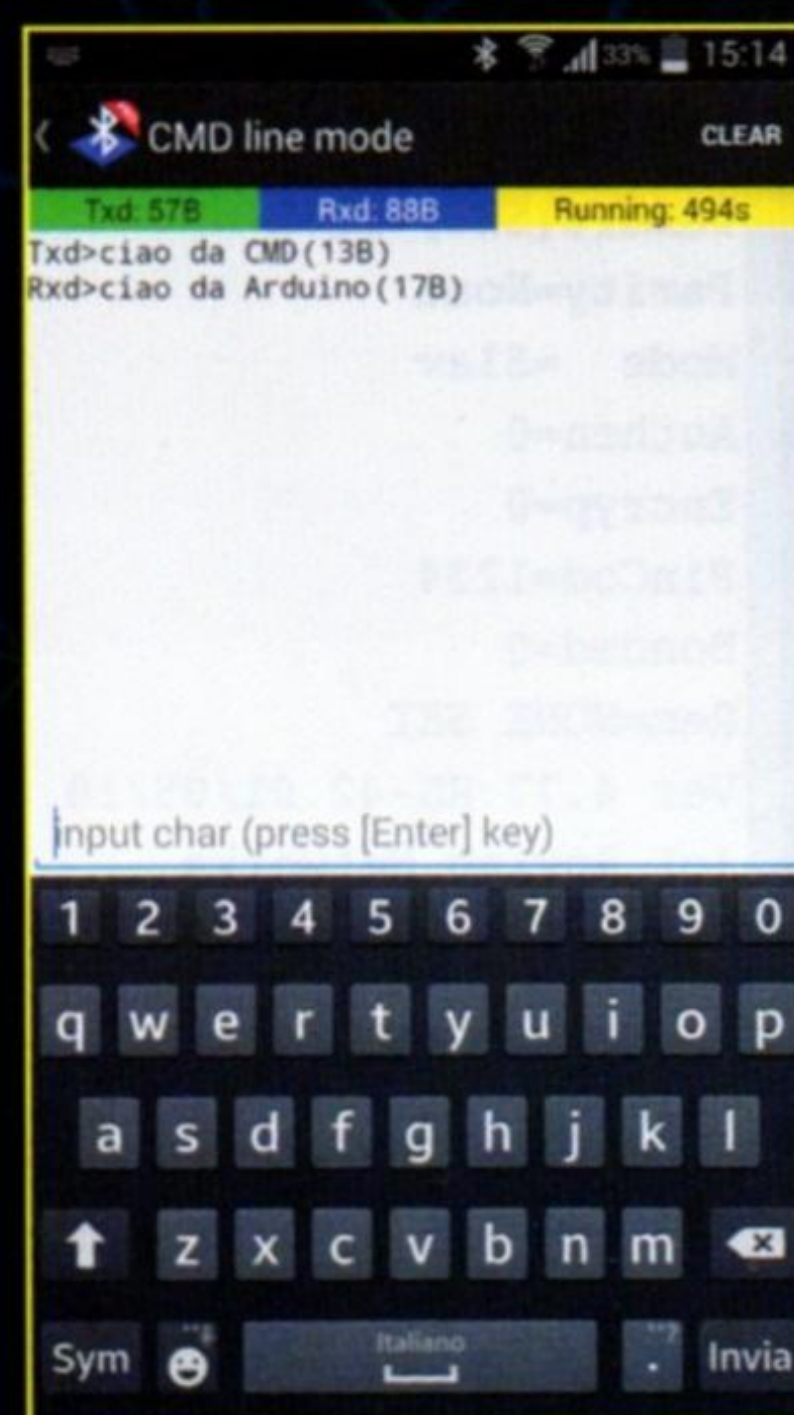
void mousePressed() {
  if (mouseY > 200 && mouseY < 400 && status==3)
  {
    input = 'x'; //invia un char
    byte[] data = { (byte)input }; //invia un byte
    //Send the data
    bt.broadcast(data);
  }
  if (mouseY > 400 && mouseY < 600 && status==3)
  {
    input = 'y'; //invia un char
    byte[] data = { (byte)input }; //invia un byte
    //Send the data
    bt.broadcast(data);
  }
}

```



Fig. 6 - Scambio dati tra BT e smartphone.

tivo che non sia la nostra scheda Arduino. Avviata la comunicazione comparirà un pulsante che, se toccato, permetterà di inviare la semplice stringa "ciao"; nel contempo qualsiasi stringa ricevuta sarà visualizzata nel display. Con questo semplice esempio si possono intuire le enormi potenzialità di sviluppo di applicazioni molto evolute e facilmente integrabili nel campo della robotica. Per dimostrare l'integrazione con il mondo Arduino proviamo a sviluppare un'applicazione che consenta di accendere e spe-

Fig. 7
Applicazione
Bluetooth SPP
pro in modalità
CMD.

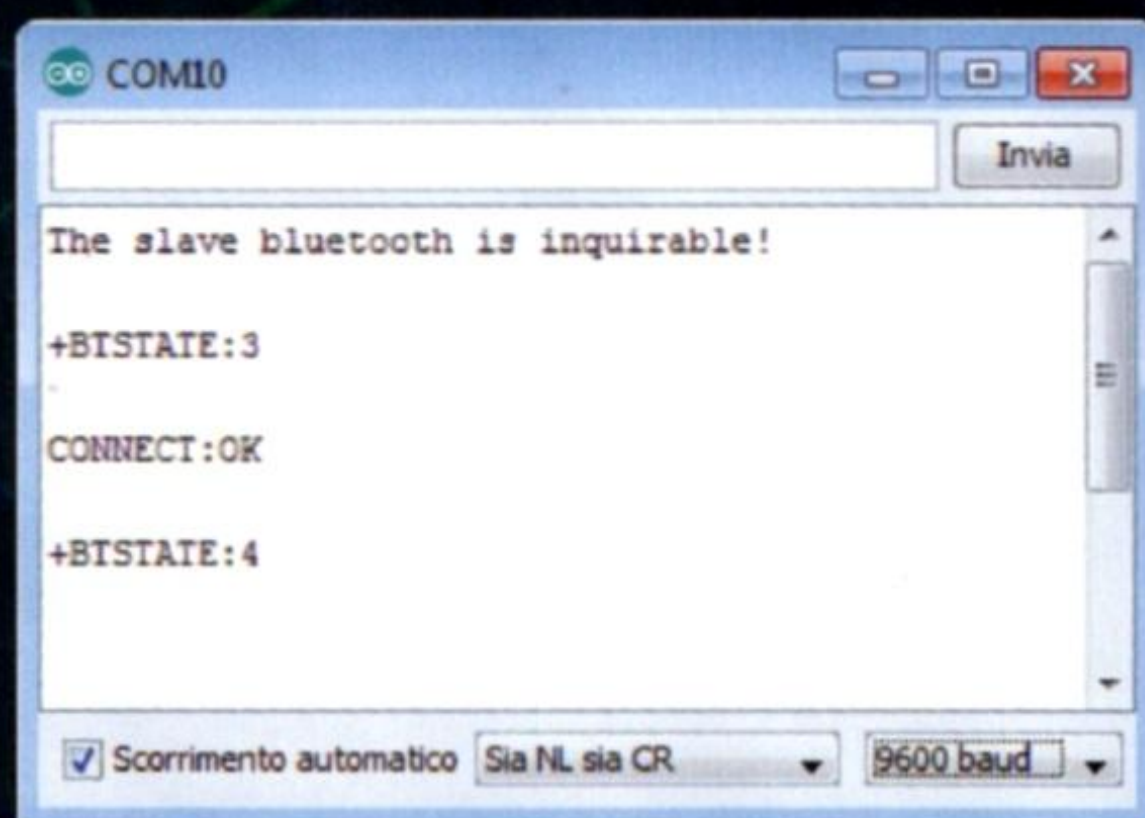


Fig. 8 - Utilizzo del modulo BTBee.

gnere un LED. Per prima cosa dovremmo programmare Arduino affinché rimanga in ascolto sulla seriale hardware, in attesa dei caratteri che giungeranno dal modulo BT; a seconda del carattere ricevuto il LED verrà acceso o spento.

Il **Listato 3** (da caricare su Arduino) è molto semplice, composto solo da una manciata di righe, e prevede anche una risposta, a conferma dell'avvenuta esecuzione del comando. Nello sketch per Processing dovremmo prevedere la presenza di due pulsanti i quali, premuti, determinano l'invio dei caratteri, rispettivamente, x e y, per accendere e spegnere il LED (**Listato 4**); l'esempio è volutamente semplificato con una grafica essenziale, per consentirvi di comprendere più facilmente il codice. Tra gli esempi allegati al corso trovate anche l'esempio 12, nel quale è stato aggiunto un joystick virtuale, tramite il quale poter inviare due coordinate utili per comandare il movimento di un robot. In questo esempio, a seconda della posizione del dito sul display, vengono rilevate le coordinate X e Y ed inviate ad Arduino nel formato "D CoordinataX CoordinataY" scalate con valore 0÷255 (Byte), facilmente utilizzabili con il comando `analogWrite` per gestire i due motori di avanzamento di un robot o la luminosità di qualche lampadina. Per consentire ad Arduino di interpretare i dati che arrivano dallo smartphone e contemporaneamente avere un debug su SerialMonitor, la comunicazione tra lo shield BT ed Arduino avviene tramite una seriale virtuale (pin 10 e 11) lasciando libera la seriale hardware di comunicare con il PC. Viste le limitate capacità della seriale software, la velocità del modulo BT dovrà essere ridotta al valore di 38.400. I collegamenti da effettuare sono i seguenti:

- 3V3 shield -> 3V3 Arduino;
- GND shield -> GND Arduino;
- pin 0 shield -> pin 11 Arduino;
- pin 1 shield -> pin 10 Arduino.

Tutti i moduli BT con profilo SPP sono compatibili con l'applicazione che abbiamo proposto, quindi anche i moduli HC-05 oppure HC-06 sono compatibili, come anche il modulo DAGU distribuito dalla ditta Futura Elettronica (codice 2586-DG010), basato sempre su un modulo HC-06. Gli ultimi moduli elencati sono del tipo plug-and-play e non necessitano di alcuna configurazione; l'unica personalizzazione prevista riguarda il nome del dispositivo e la velocità di comunicazione. Il nome del dispositivo è del tutto arbitrario, mentre la velocità di comunicazione impostata (o predefinita) è quella da utilizzare nell'applicazione per Arduino; il dispositivo Android si adeguerà al valore impostato sul modulo. L'applicazione non è compatibile

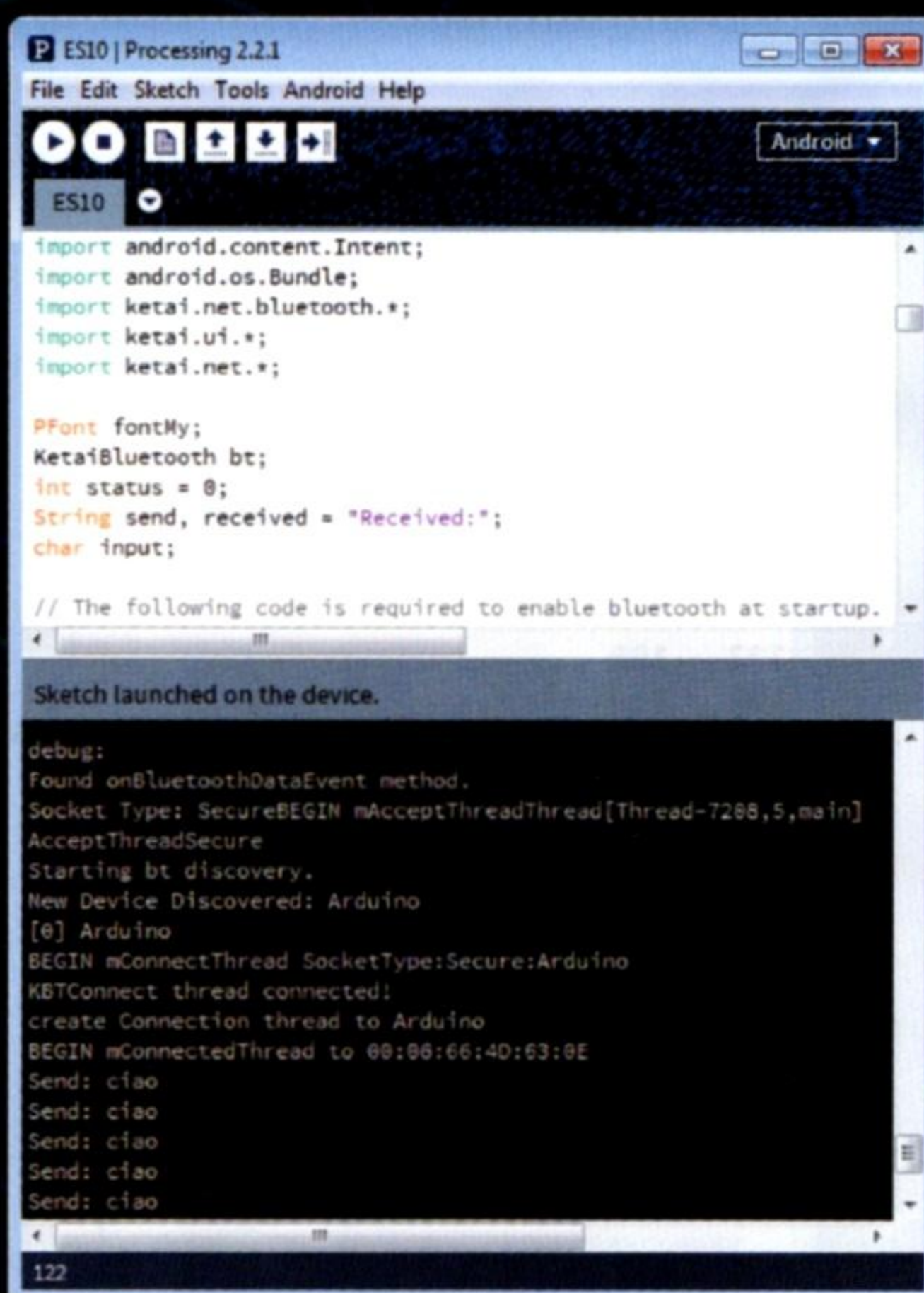


Fig. 9 - Schermata di Processing con avviato lo sketch n° 10.

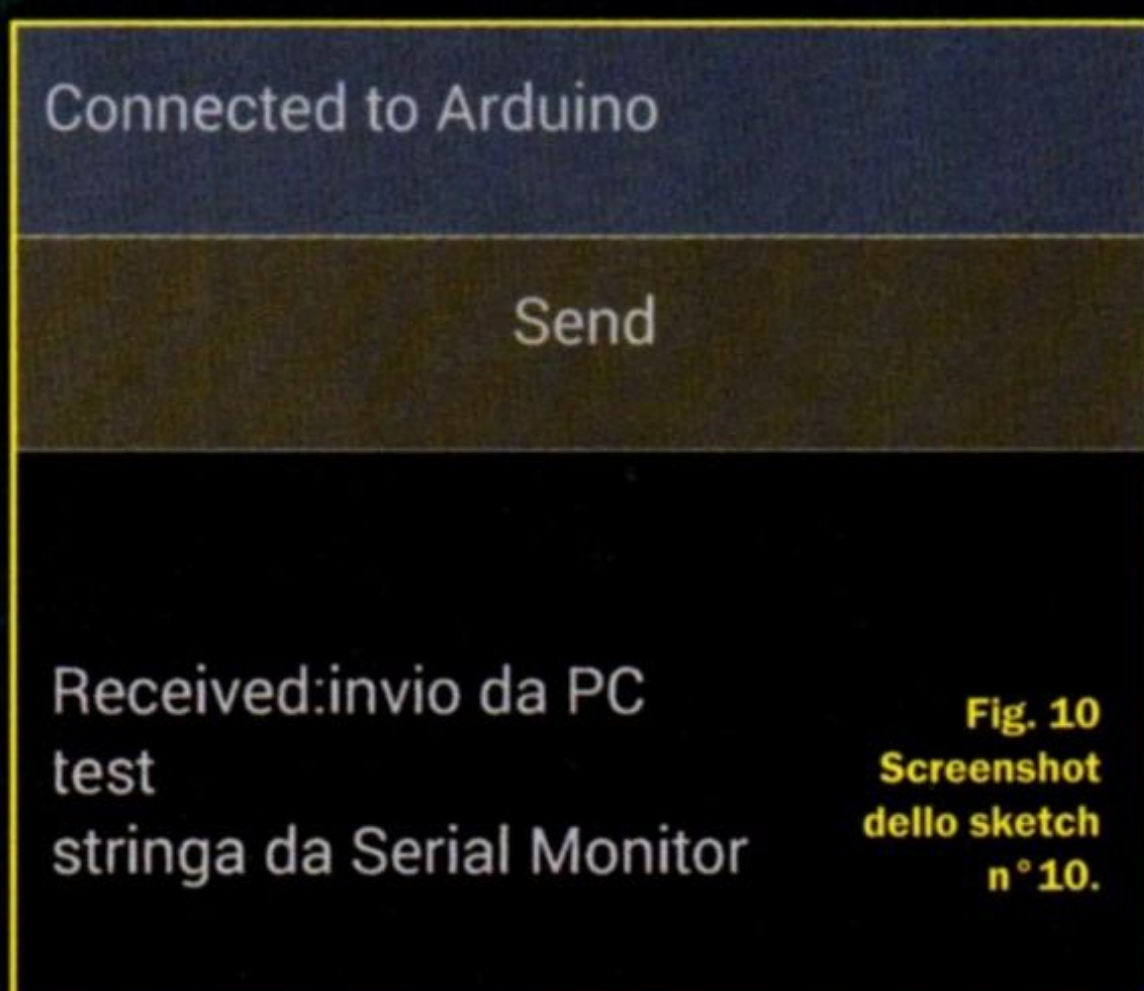


Fig. 10
Screenshot
dello sketch
n° 10.

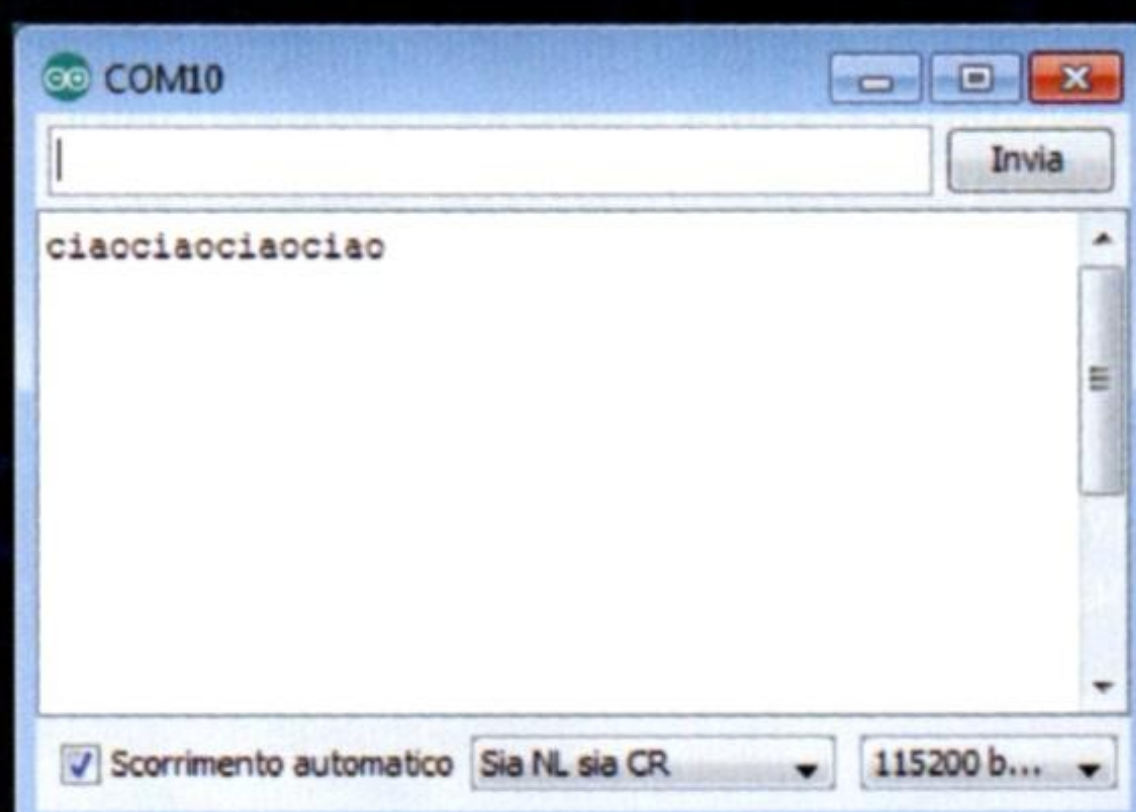


Fig. 11 - Schermata di SerialMonitor per lo sketch ° 10.

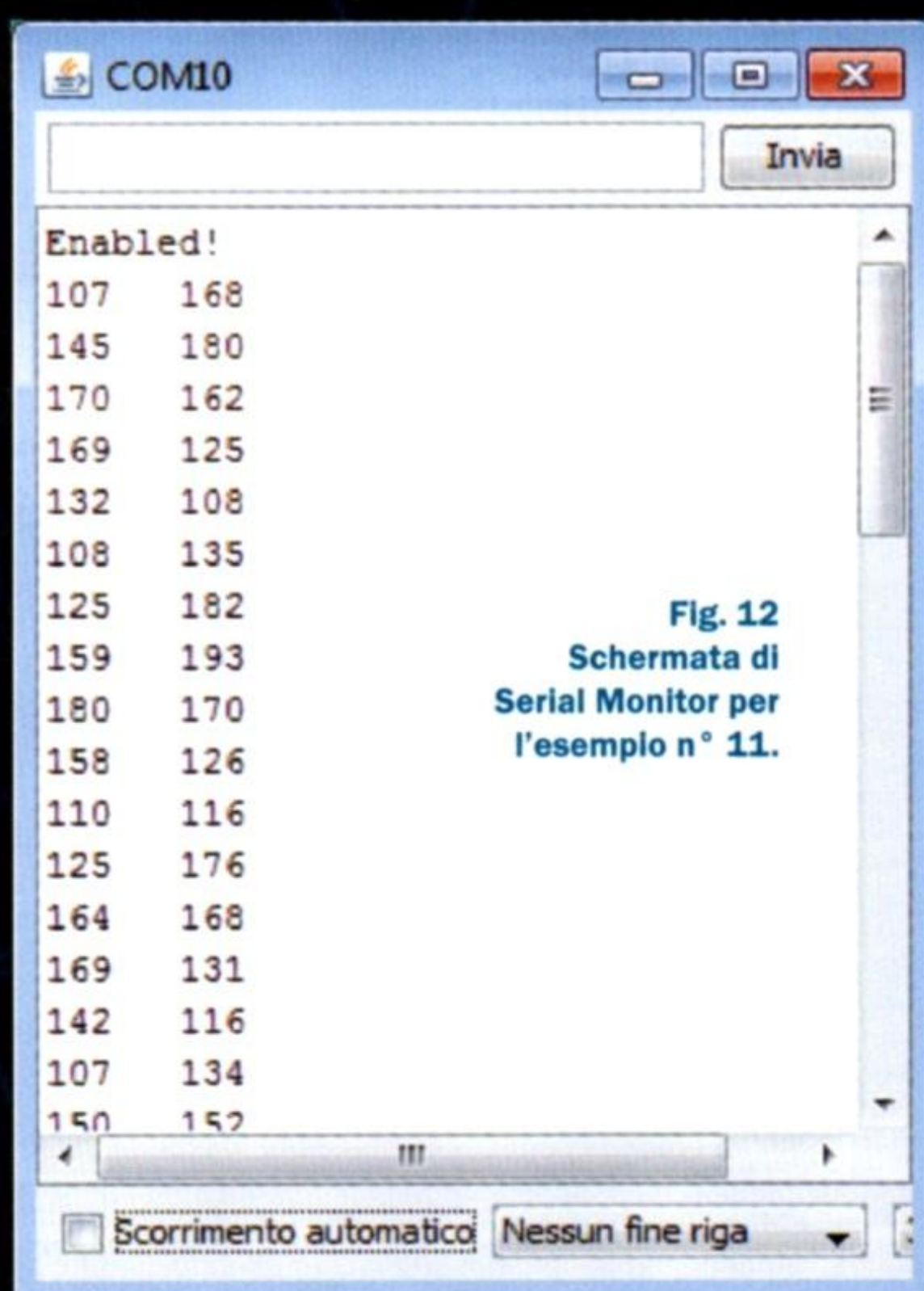


Fig. 12
Schermata di
Serial Monitor per
l'esempio n° 11.

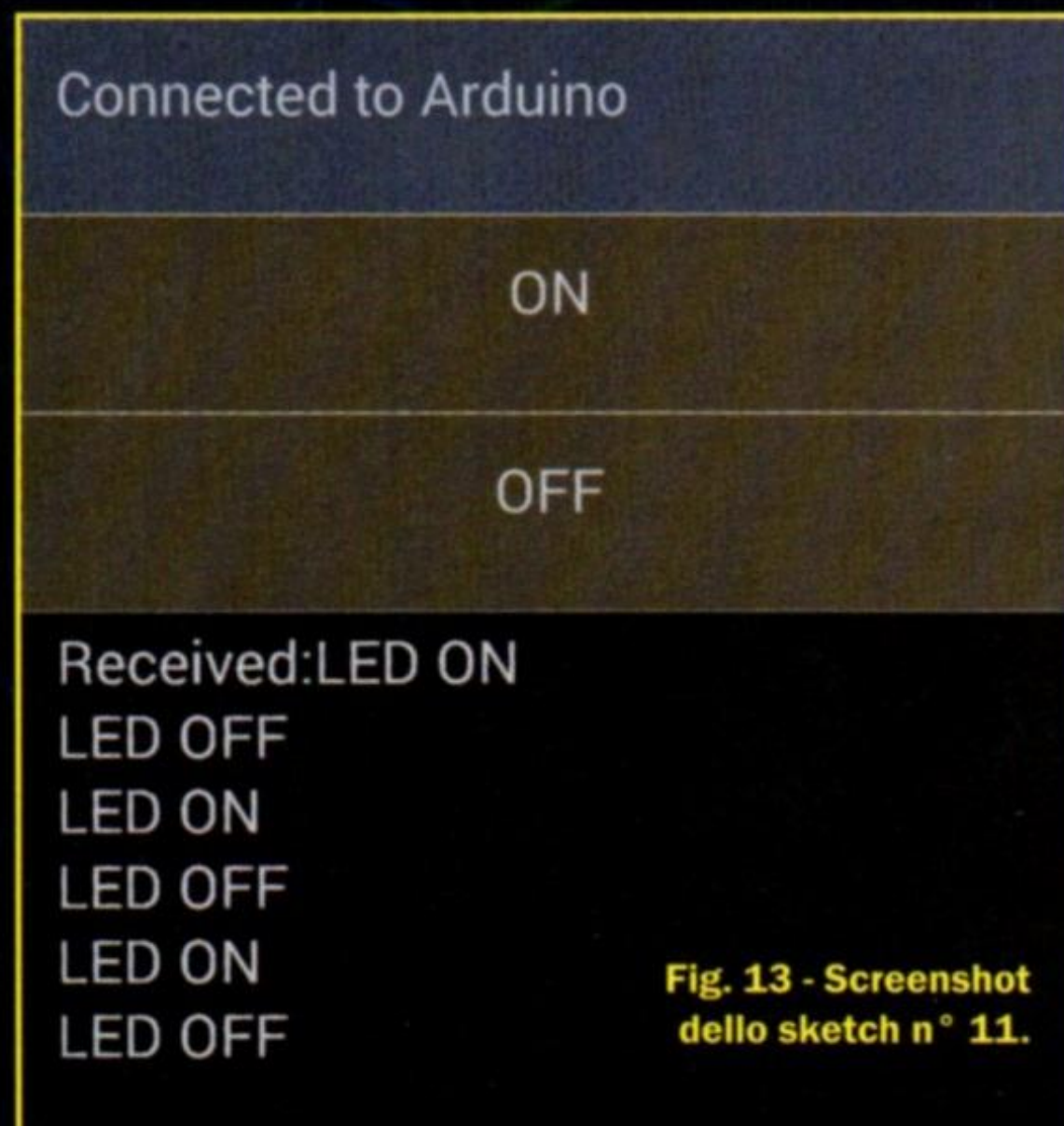


Fig. 13 - Screenshot
dello sketch n° 11.

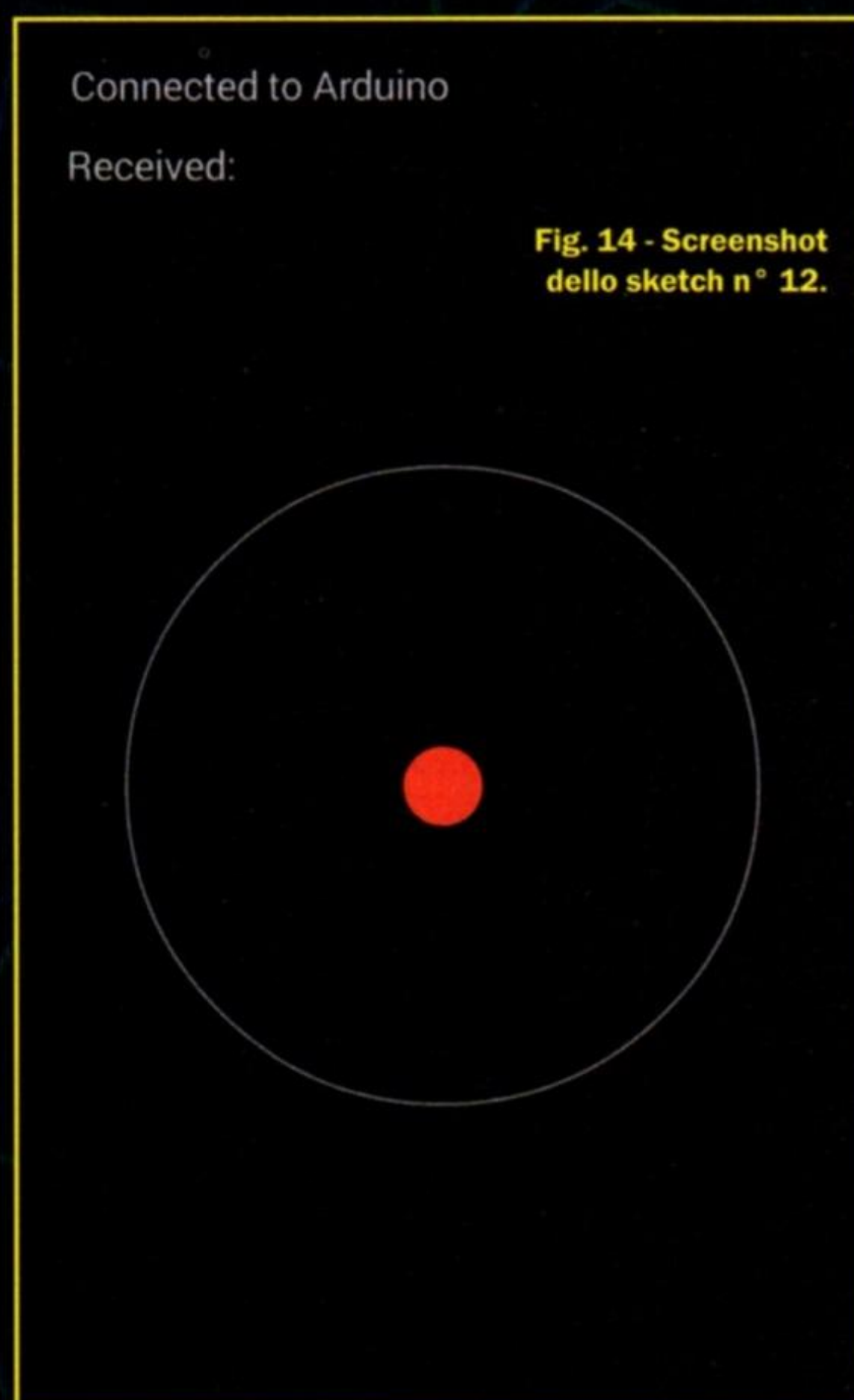


Fig. 14 - Screenshot
dello sketch n° 12.

con i moduli Bluetooth Low Energy (BLE 4.0).

CONCLUSIONI

Per questa puntata è tutto; nella quinta ci occuperemo della comunicazione WiFi, sempre prendendo ad esempio l'interfacciamento con una scheda Arduino.

Entra in esercizio negli USA l'impianto Goodwell di Enel Green Power

Enel Green Power (EGP) ha completato e connesso alla rete Goodwell, un impianto eolico da 200 MW situato nella contea di Texas County, in Oklahoma. "L'Oklahoma è uno dei migliori luoghi al mondo in termini di risorse eoliche," ha dichiarato Francesco Venturini, Amministratore Delegato e Direttore Generale di EGP. "Goodwell è il quinto parco eolico che EGP ha messo in esercizio dalla sua entrata, nel 2012, nel mercato dell'Oklahoma. Il progetto conferma il continuo investimento di EGP nel settore delle rinnovabili negli Stati Uniti. Stiamo compiendo rapidi progressi rispetto ai nostri obiettivi di

espansione della capacità, insieme ad iniziative dirette alla comunità nel campo delle infrastrutture e dell'istruzione, per garantire uno sviluppo realmente sostenibile in tutte le regioni in cui siamo presenti." La realizzazione del parco eolico, di proprietà di Goodwell Wind Project LLC, controllata da Enel Green Power North America Inc. (EGPNA) ha richiesto un investimento complessivo di quasi 310 milioni di dollari statunitensi, in linea con gli obiettivi di crescita dell'attuale business plan di EGP.

Nel luglio 2014 EGPNA ha firmato un *capital contribution agreement* con un consorzio guidato da JP Morgan a copertura parziale dell'investimento. Goodwell, uno dei più grandi impianti eolici del portfolio di EGPNA, è in grado di generare circa 860 milioni di kWh all'anno, equivalenti al fabbisogno di consumo annuale di oltre 73 mila famiglie americane, evitando l'emissione in atmosfera di circa 450 mila tonnellate di CO₂ ogni anno. All'impianto di Goodwell è associato un contratto di vendita ventennale dell'energia prodotta (Power Purchase Agreement - PPA). Enel Green Power in USA detiene e gestisce oltre 90 impianti in 21 Stati degli USA e in due province canadesi. Ad oggi, la Società ha una capacità installata totale di oltre 2.200 MW che comprende quattro tecnologie di generazione: eolico, geotermico, solare e idroelettrico.

www.enel.it

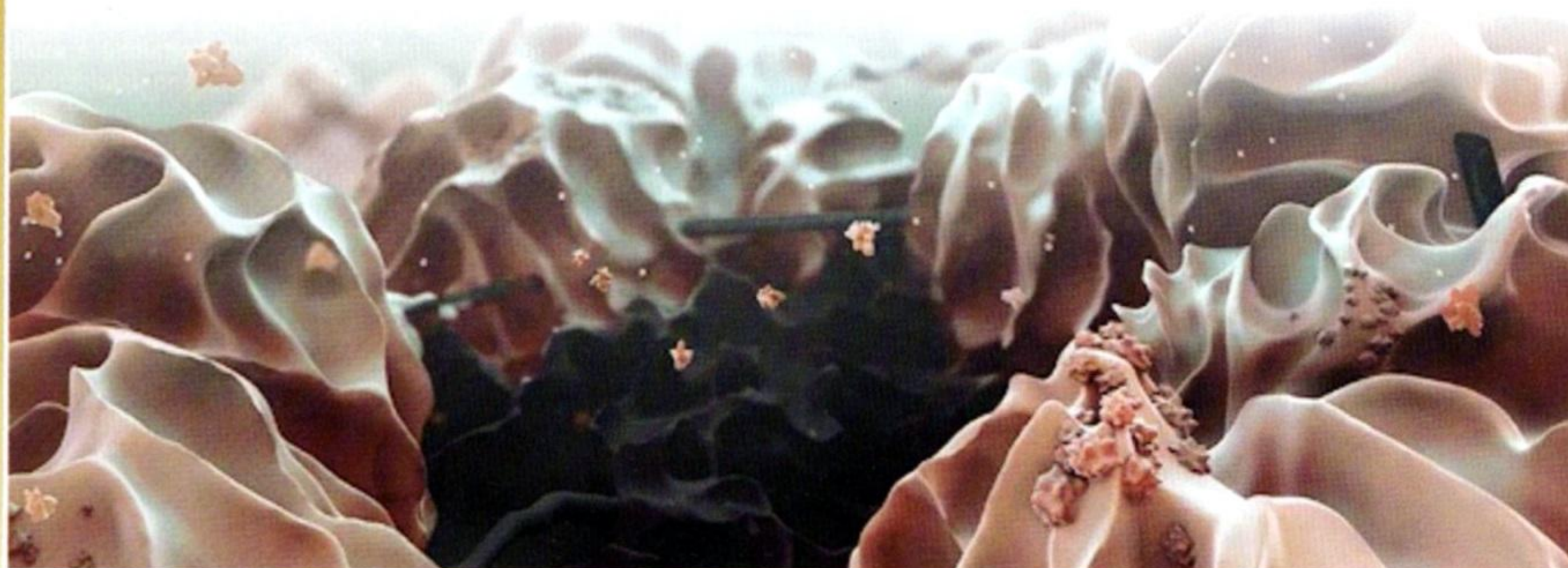
Batteria litio-aria: 650 km con una sola carica

I ricercatori dell'Università di Cambridge hanno realizzato un nuovo prototipo di batteria litio-aria dalla densità energetica molto elevata, capace di sostenere duemila ricariche, con un'efficienza energetica del 90%. Tale tipologia di batterie potrebbe offrire una densità energetica capace di assimilare in futuro l'autonomia elettrica a quella dei motori endoter-

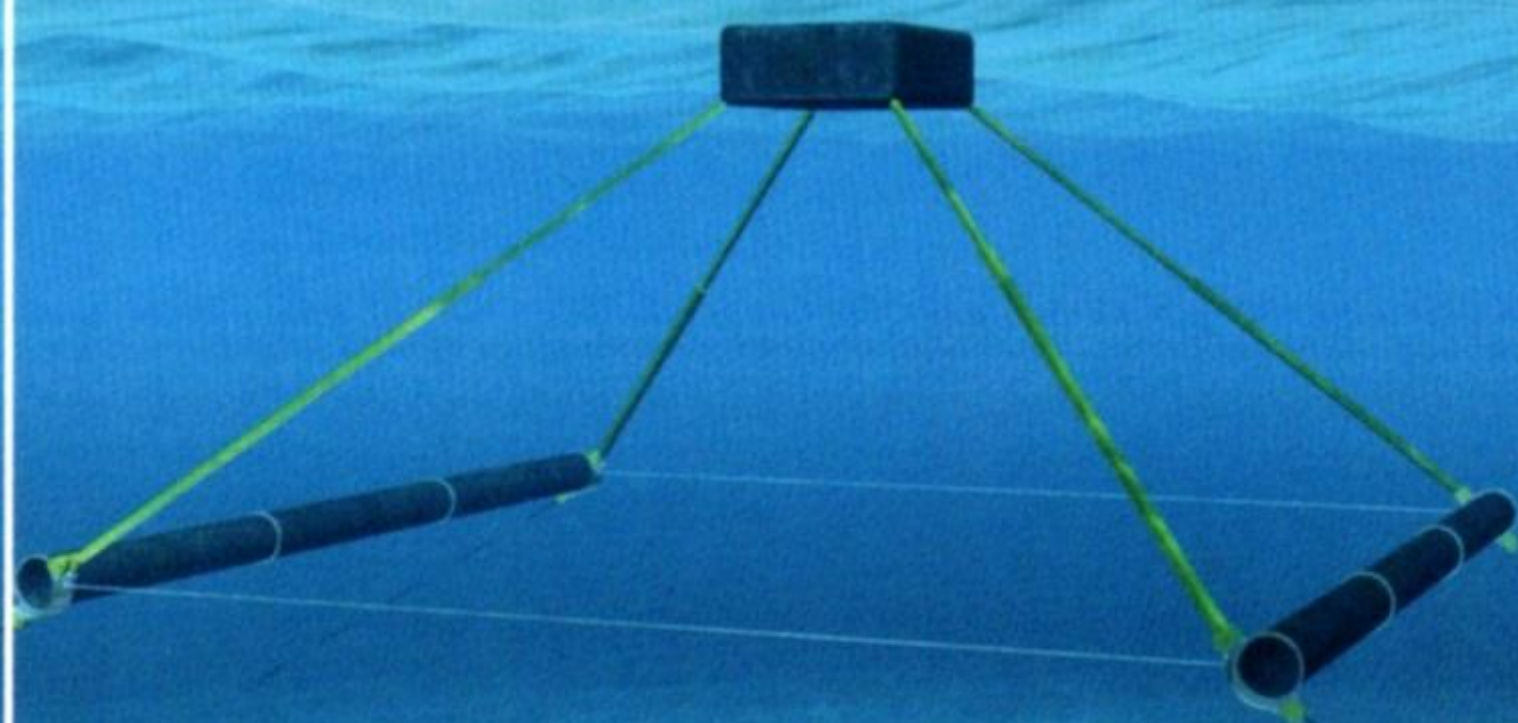
mici, consentendo ad una vettura di percorrere 650 chilometri con una sola carica a un quinto del costo di oggi e ad un peso minore. I ricercatori di Cambridge hanno impiegato un elettrodo di grafene molto poroso, sfruttando l'idrossido di litio al posto del perossido e utilizzando lo ioduro di litio come mediatore, ottenendo come risultato una batteria più stabile

ed efficiente. Secondo le previsioni serviranno circa dieci anni per avere una soluzione litio-aria commercializzabile. La tecnologia, che per ora funziona solo con ossigeno puro, è stata già brevettata e sarà venduta attraverso la divisione commerciale dell'Università.

www.cam.ac.uk



Energia dalle onde: il progetto di 40South Energy

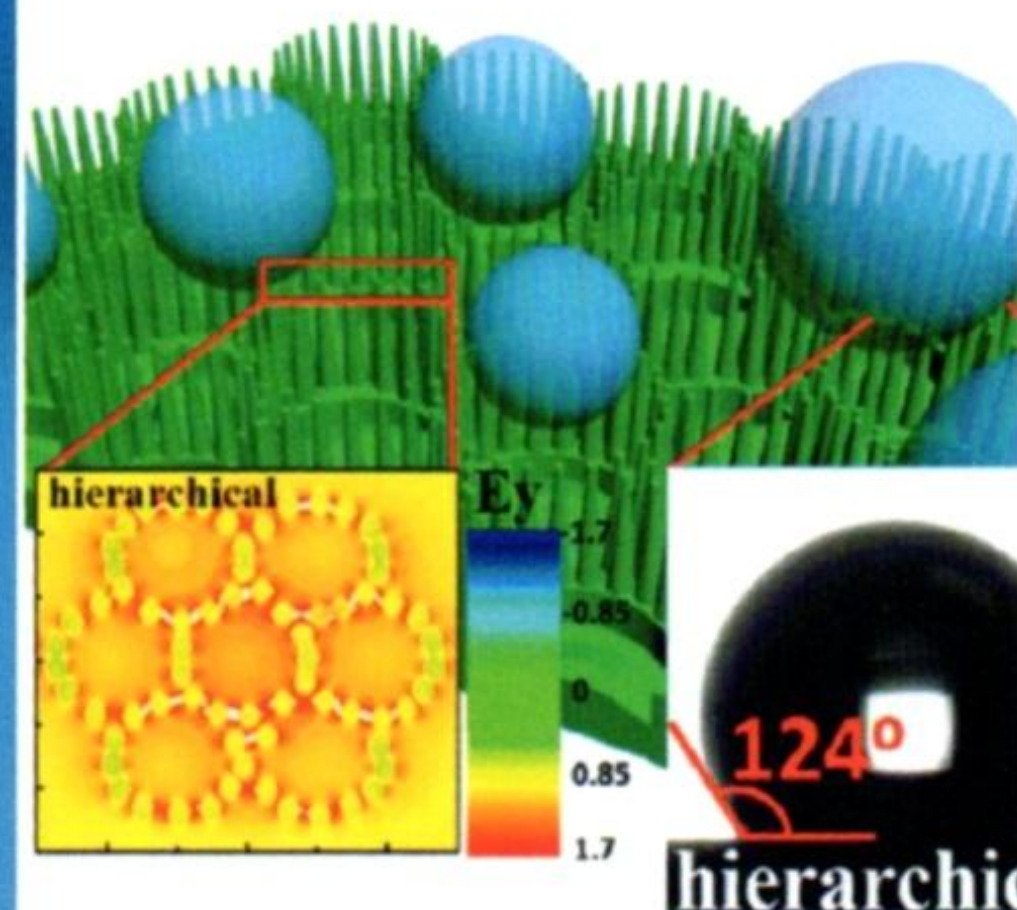


Al largo di Marina di Pisa, il professor Michele Grassi ha realizzato una macchina capace di garantire il fabbisogno energetico di quaranta famiglie. Grassi, creatore di 40South Energy, una delle start up più innovative d'Italia, ha ideato il convertitore di energia ondosa R115 da 200 kW, posto a sei metri di profondità dinanzi alle coste di Marina di Pisa. Il funzionamento di questa sorta di ottovolante è semplice: il sistema incanalizza la forza delle onde per poi trasfor-

marla in elettricità, dando vita anche al progetto Blu Marina per creare una rete elettrica speciale. Grassi lancerà anche una campagna di crowdfunding che consentirà a chiunque di acquistare una quota del progetto. Le singole macchine hanno un costo di 150 mila euro l'una e generano energia con un guadagno medio di circa 26 mila euro l'anno.

www.40southenergy.com

Energia solare: un nano vetro per catturare la luce da ogni angolazione



“Hydricity” il solare termico per produrre elettricità e idrogeno

Hydricity è il termine utilizzato da alcuni scienziati statunitensi e svizzeri per definire la possibilità (per ora teorica) di sfruttare l'energia del sole per produrre idrogeno tramite gli impianti solari termici, garantendo così una produzione di energia elettrica 24 ore su 24 grazie all'utilizzo dell'idrogeno. Per giunta con un rendimento simile a quello delle migliori celle fotovoltaiche. Gli impianti solari termici operano oggi con temperature massime di 625°C; gli scienziati hanno notato che con temperature operative superiori è possibile incrementare in maniera significativa l'efficienza. Per di più, con temperature superiori a 725°C risulta possibile, utilizzando il vapore, separare facilmente le due componenti fondamentali dell'acqua, l'idrogeno e l'ossigeno. Un sistema integrato “Hydricity” potrebbe

dunque produrre sia vapore per generare elettricità durante il giorno che idrogeno per immagazzinare l'energia. Utilizzato per produrre esclusivamente idrogeno, un impianto solare di questo tipo potrebbe raggiungere un'efficienza del 50% grazie all'alta pressione del vapore che facilita la produzione di idrogeno (elettrolisi ad alta temperatura).

Secondo gli scienziati questo nuovo sistema di generare energia elettrica sfruttando il calore del sole potrebbe raggiungere un'efficienza del 46%, grazie anche all'impiego di turbine ad alta pressione seguite da turbine a pressione più bassa. Bruciando l'idrogeno accumulato, il sistema è in grado di produrre energia elettrica senza interruzioni nell'arco delle 24 ore, con un'efficienza per quanto riguarda l'idrogeno, fino al 70%, pari a quella delle migliori

pile a combustibile. Complessivamente, l'efficienza di un impianto “Hydricity” potrebbe raggiungere nell'arco delle 24 ore un'efficienza di circa il 35%, pari a quella delle migliori celle fotovoltaiche a multi-giunzione. Inoltre, l'idrogeno prodotto potrebbe trovare impiego anche nel campo dei trasporti, della produzione chimica e in altre industrie. Con l'ulteriore vantaggio di un'assenza di degrado nel tempo dell'energia immagazzinata, al contrario delle batterie elettrochimiche, le quali hanno anche un limitato numero di cicli di carica/scarica. Questa iniziativa vede in prima linea gli scienziati della Purdue University di West Lafayette in Indiana (USA) e quelli svizzeri del Federal Polytechnic School of Lausanne.

www.purdue.edu

Il nuovo rivestimento per fotovoltaico progettato dai ricercatori del King Abdullah University of Science & Technology (KAUST) e del National Central University di Taiwan, migliora sensibilmente la raccolta della luce solare.

Il vetro nanostrutturato imballato con un certo criterio aumenta l'efficienza di conversione della cella solare se applicato ad un pannello fotovoltaico, tanto da consentire l'assorbimento della luce solare da diverse angolazioni.

Il rendimento dei pannelli fotovoltaici di solito è maggiore quando i raggi solari colpiscono il modulo in maniera perpendicolare. Il rivestimento di vetro vanta una struttura gerarchica con nanotubi ultrasottili interposti a nanopareti organizzate a nido d'ape, che disperdono i raggi solari in entrata, incrementando la capacità di raccolta da più angolazioni.

L'efficienza della cella passerebbe dal 5,2% al 27,7% a seconda dell'angolo incidente della luce, migliorando notevolmente le prestazioni del sistema. Jr-Hau Lui, Kun-Lai Yu e colleghi hanno pubblicato il loro studio su ACS Nano.

<http://pubs.acs.org>

Le mega turbine eoliche galleggianti del progetto LIFES50+



LIFES50+ è un progetto che vede la partecipazione di dodici partner europei nato per promuovere lo sviluppo di una nuova generazione di turbine eoliche galleggianti e che al momento ha raccolto 7,3 milioni di euro di finanziamenti.

A guidare l'iniziativa la norvegese MARINTEK, che mira a realizzare innovative sotto-strutture galleggianti per le turbine eoliche da 10 MW di potenza e per una profondità superiore a 50 metri.

La scelta di creare mega turbine è considerata rilevante al fine di ridurre il costo

dell'energia prodotta da fonti rinnovabili offshore. Il Politecnico di Milano, che partecipa attivamente al progetto, si occuperà della validazione sperimentale di avanzati modelli di calcolo per il calcolo aerodinamico, che sarà svolta testando nella Galleria del Vento modelli in scala delle turbine. Il galleggiamento sarà simulato tramite un robot esapode capace di riprodurre gli spostamenti alla base della macchina in maniera simile a ciò che avverrà nella realtà.

<http://lifes50plus.eu>

BiosPHera 2.0: la casa passiva del futuro

È stato presentato ad Aosta, il progetto BiosPHera 2.0 il modulo abitativo itinerante, che sarà realizzato adottando le migliori tecnologie ad oggi disponibili per la massima efficienza energetica e il più avanzato know-how sul benessere di chi vi abita.

Il progetto è promosso da Aktivhaus, Politecnico di Torino DAD, Università della Valle d'Aosta, Vallée d'Aoste Structure e dagli istituti Zephir, Minergie e PEFC, con il patrocinio della Regione Valle d'Aosta e la partecipazione di diverse aziende nazionali e internazionali.

Il modulo abitativo BiosPHera 2.0 sarà installato e presentato al pubblico e agli operatori a Courmayeur, nel mese di febbraio, per poi partire verso Torino, a seguire verso Rimini e il mare Adriatico, infine per Lugano.

Il modulo verrà abitato e sottoposto agli stress più impegnativi: inverno in una località di montagna con temperature molto rigide che raggiungeranno con ogni probabilità diversi gradi sottozero, in estate in una località marittima, in

presenza di temperature molto alte. I dati raccolti per più di 12 mesi saranno monitorati dal Politecnico di Torino. Temperatura dell'aria e dei muri, umidità, qualità dell'aria e presenza di particolato, ma anche dati relativi alle reazioni fisio-

logiche ed emotive degli abitanti condizionate dall'ambiente come temperatura corporea, conduttività della pelle, battito cardiaco.

www.biosphera.ph

