Mensile di progettazione elettronica, attualità scientifica, novità tecnologiche.

Elettronica In

oltre l'elettronica

www.elettronicain.it

Anno XVIII - n. 166
Maggio 2012



Robot Area

- RoboCup Jr, i team italiani alla finale mondiale

Spiderin, Bipe e Filippo incontrano Arduino

Combinatore vocale GSM



- Lettore RFID con SD Card e bus I²C
- Programmatore HV modulare per MCU Atmel
- Minibus, un bus per l'automazione
- Shield universale per moduli GSM/GPRS/GPS
- Lettore mp3 juke box
- LTSpice IV tutorial
- Corso Atmel OPEN SOURCE

Tutorial open Picus

Ad ogni misura
la sua soluzione



Per ogni tipo di applicazione, dalle più semplici alle più complesse, la piattaforma d'acquisizione dati di National Instruments è in grado di gestire ogni misura nel modo più efficace. Qualunque sia il tuo budget, il numero di canali richiesto e quali siano le tue specifiche esigenze a livello di prestazioni, la potenza degli I/O e il software in grado di incrementare la produttività dei tuoi progetti saranno sempre all'altezza della sfida da affrontare. Questa combinazione vincente permette di bilanciare prestazioni e investimenti fornendo la massima accuratezza di misure sempre scalabili e affidabili. **Progettata per prestazioni eccellenti.**



>> Scopri la piattaforma NI per l'acquisizione dati alla pagina web **ni.com/daq/i**

NATIONAL INSTRUMENTS ITALY S.r.l. • Via Anna Kuliscioff, 22 • 20152 Milano • Tel.: 02 413091 • Fax: 02 41309215 • P.IVA/C.F.: 09779210155
CCIAA Milano 1311989 • Iscr. Trib. Milano 298373/3362/23

©2011 National Instruments. Tutti i diritti riservati. National Instruments, NI ed ni.com sono marchi registrati di National Instruments.
Altri prodotti e nomi aziendali citati sono marchi commerciali delle rispettive aziende. 02922

NATIONAL INSTRUMENTS

Elettronica In

www.elettronicain.it
Rivista mensile, anno XVIII n. 166
MAGGIO 2012

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

Redazione:
Gabriele Daghetta, Paolo Gaspari, Boris Landoni,
Marco Magagnin, Simone Majocchi,
Davide Scullino, Alessandro Sottocornola
(*redazione@elettronicain.it*)

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

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

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

DIREZIONE, REDAZIONE, PUBBLICITA':
VISPA s.n.c. - via Adige 11 - 21013 Gallarate (VA)
Telefono 0331-752668 Fax 0331-778112

Abbonamenti:
Annuo 10 numeri Euro 45,00
Estero 10 numeri Euro 90,00
Le richieste di abbonamento vanno inviate a:

VISPA s.n.c.
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.
Vispa Edizioni è iscritta al Registro Operatori della Comunicazione n. 3754 del 27/11/2001.
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 ElettronicaIn. L'invio di articoli, materiale redazionale, programmi, traduzioni, ecc. implica da parte del Collaboratore l'accettazione dei compensi e delle condizioni stabilite dall'Editore (*www.elettronicain.it/ase.pdf*). Manoscritti, disegni e foto non richiesti non verranno in alcun caso restituiti. L'utilizzo dei progetti e dei programmi pubblicati non comporta alcuna responsabilità da parte della Società Editrice.
© 1995÷2012 VISPA s.n.c.

Un sogno in 3D

Ammirato, sto guardando dalla finestra una specie di grosso ragno, le quattro zampe ben piantate per terra, muove una specie di proboscide, avanti e indietro, poi si alza un po', ancora avanti e indietro. Sta costruendo un muro, strato dopo strato, dove c'è la finestra, ovale, interrompe il getto di materiale dalla proboscide, poi continua dall'altra parte. Intorno a questo strano mostro non c'è nessuno, a parte un gruppetto di signori al di qua delle transenne che guardano in silenzio, non scuotono neanche la testa. Sul limitare del cantiere un furgone, lucido e illuminato all'interno. Si intravede un signore che lavora ad un computer. Purtroppo non ho tempo di ammirare a lungo, prima di andare al lavoro devo passare dal carrozziere, ieri ho strisciato una portiera entrando nel box. Quando arrivo il carrozziere avvicina una specie di lettore al bordo in basso della portiera, dove ormai da tempo tutte le portiere hanno annegato un tag Rfid con i riferimenti tecnici per riprodurle e mi dice che domani mattina sarà pronta.



Dentro l'officina una specie di plotter sta finendo di riprodurre il paraurti della vettura in cortile. Intanto che guardavo il ragno ho dimenticato la caffettiera sulla cucina a gas e ho bruciato il manico, stasera cercherò sul sito del produttore il file giusto e me lo riprodurrò sulla stampante (una volta si diceva stampante 3D, adesso solo stampante) sempre che mio figlio non la stia usando per qualche suo gadget e mia moglie non stia stampando i segnaposto per la cena con gli amici di sabato. Proprio buffa quella stampante, oltre ad essere carina e silenziosa quando si accorge di qualcosa che non va si accende una lucina e da quel momento pretende di ristamparsi i suoi pezzi di ricambio. Quasi quasi scarico i file e me ne stampo una tutta per me, da tenere lontano da moglie e figlio.
La sveglia. Lo sapevo, era un sogno, caffè e vado in ufficio, che mi tocca, mica come quei signori che guardano i muratori del cantiere qua sotto, scuotendo la testa. Ahia mi sono scottato, la caffettiera ha il manico rotto, e dovrò anche pensare a quel graffio sulla portiera. Bello, però, il sogno. In ufficio, trovo l'Economist, che sostiene che lo sviluppo futuro deve passare per nuovi materiali, robot economici, software intelligenti, servizi online sempre più diffusi e stampanti 3D (per ora si chiamano ancora così) di ogni tipo e dimensione. General Electric sta finalizzando risorse enormi a questo obiettivo, con lo scopo, tra gli altri, di rendere nuovamente remunerativo produrre in occidente; che sia ripresa? Ha anche finalizzato una sua campagna pubblicitaria sulla stampa 3D, il risultato su Thingiverse GE. I materiali utilizzabili con le stampanti 3D sono sempre di più: ABS, PLA (un derivato del granoturco, ecologico), resine, polimeri, carbonio, alluminio, pietra. Già, pietra, l'italiana Dinitech ha brevettato la D-Shape, il ragno del sogno. All'ultimo salone del mobile, a Milano, lampade e arredi da sogno (appunto), sono stati prototipati o addirittura realizzati in stampa (3D). E anche noi, nel nostro laboratorio, una MakerBot ce l'abbiamo. A proposito, vediamo se mi fanno il manico della caffettiera. E poi stiamo per partire per il Maker Faire in California. Nella nostra infanzia, sul Corriere dei Piccoli, Pier Lambicchi faceva un disegno, poi lo passava con l'Arcivernice, ed il disegno diventava un oggetto reale. E noi, quell'idea, non siamo mai riusciti a togliercela dalla testa. A scuola, oltre a scrivere e a far di conto, è ora di imparare a disegnare, col CAD. Per il graffio sulla portiera provo ad aspettare, chi sa mai.

Arsenio Spadoni
(Arsenio.Spadoni@elettronicain.it)

Sommario

Maggio 2012
n. 166

ARTICOLI

23 APPLICAZIONI
LETTORE RFID CON SD CARD E BUS I²C

Realizziamo un lettore di tessere RFID per controllo accessi dotato di interfaccia utente a display, una SD Card per memorizzare le liste degli utenti abilitati e indesiderati e di un protocollo I²C per comunicare con periferiche digitali. Prima puntata.



33 DIDATTICA
ALLA SCOPERTA DI OPEN PICUS

Uno dei punti di forza della piattaforma openPicus è rappresentato dalla disponibilità di un ambiente di sviluppo molto completo e strutturato, quindi facile da usare, oltre che gratuito. Vi presentiamo l'introduzione all'IDE, le modalità di installazione ed i primi passi per la creazione di un progetto. Seconda Puntata.



38 SICUREZZA
COMBINATORE VOCALE GSM

Collegato ad una centrale antifurto o antincendio, in caso di allarme effettua chiamate telefoniche riproducendo messaggi vocali. Comandato via DTMF può azionare attuatori sul posto. Prima puntata.



51 DIDATTICA
LTSPICE IV TUTORIAL

Analizziamo il comportamento degli integrati di Linear Technology facendo ricorso alla versione del simulatore SPICE appositamente personalizzato da LT. In particolare, ecco un esempio con il dispositivo di protezione della tensione di alimentazione LTC4365.





IN COPERTINA:
Dotiamo il nostro antifurto di un combinatore su rete telefonica GSM, provvisto di sintesi vocale in grado di riprodurre messaggi personalizzati.

ELENCO INSERZIONISTI

- Exposition Service
- Fortronic
- Futura Elettronica
- HAM Radio
- National Instruments
- RM Elettronica
- Robotica
- SMAU
- www.mdsrl.it

Robot Area

85
RoboCup Jr,
I TEAM ITALIANI ALLA FINALE MONDIALE

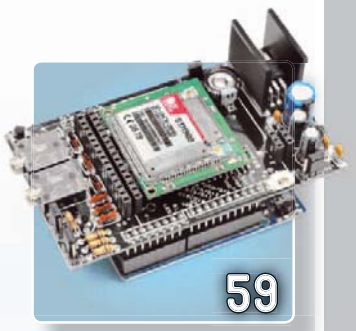
SPIDERIN, BIPE E FILIPPO INCONTRANO ARDUINO



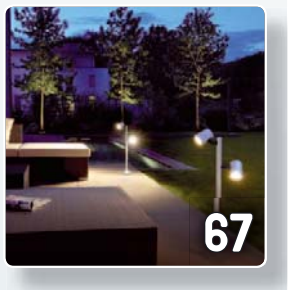
La tiratura di questo numero è stata di 18.500 copie.

Mensile associato all'USPI,
Unione Stampa Periodica Italiana

59
ARDUINO
SHIELD UNIVERSALE PER MODULI GSM/GPRS/GPS
Appositamente studiato per i moduli della SIMCom, si adatta a diversi modelli grazie ad uno zoccolo polivalente e librerie specifiche. Ultima puntata.



67
DOMOTICA
MINIBUS, UN BUS PER L'AUTOMAZIONE
Un sistema flessibile per l'automazione domestica, di apparecchiature ed impianti con un modulo centrale di controllo e moduli periferici specializzati collegati con il protocollo I2C a livello di segnale potenziato per poter coprire lunghe distanze. Prima puntata



79
IN VETRINA
LETTORE MP3 JUKE BOX
Riproduce in sequenza i brani musicali registrati su Pen Drive USB della capacità massima di 16 GB; si comanda con tre pulsanti e fornisce in uscita un segnale per un piccolo amplificatore o un Hi-Fi.



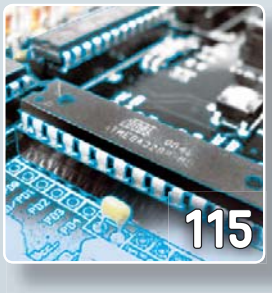
RUBRICHE

01 EDITORIALE
05 ELETTRONICA INSIEME
13 APPUNTAMENTI
17 COMPONENTI & SISTEMI
20 EVENTI
112 SCIENZA & TECNOLOGIA
124 RINNOVABILI
128 MERCATINO

101
DIDATTICA
UN PROGRAMMATORE HV MODULARE PER MCU ATMEL
Analizziamo la sezione "Keypad & Display" e spieghiamo come assemblare l'insieme ed inserirlo in un contenitore adatto. Seconda parte.



115
DIDATTICA
CORSO ATMEL OPEN SOURCE
Presentiamo gli strumenti per lavorare praticamente con il microcontrollore ATMEGA16: apprendiamo i sistemi per programmarlo costruendo una scheda di sviluppo in cui inserirlo. Quinta puntata.



PROMOSSO DA:

assodel — Associazione Nazionale Fornitori Elettronica

Gruppo E&IC — Embedded & Industrial Computer

21 GIUGNO 2012 - MILANO

CENTRO CONGRESSI MILANOFIORI
Assago (MI)

EMBEDDED IC & AUTOMATION
RF & WIRELESS

Due eventi paralleli in un'unica giornata con approfondimenti tecnologici, dibattiti e incontri tra professionisti per dare uno sguardo insieme al futuro. Due conferenze plenarie, coordinate dai comitati tecnici interni all'associazione, e una serie di workshop e seminari per fare focus su:

1. EMBEDDED – IC & AUTOMATION

- x86 vs ARM
- Android
- Windows Embedded 8
- Casi applicativi: Industrial Automation Power & Energy - Transportation
- Display: Touchscreen e E-Signage

2. RF & WIRELESS

- Energy harvesting
- Smart grid e smart building
- Benchmarking delle tecnologie wireless più utilizzate

IL TARGET

Un evento dedicato esclusivamente agli specialisti del settore: tecnici, progettisti, acquisitori, system integrator, software house, consulenti che operano nei mercati digital signage, industrial automation, logistica, medicale, military & avionics, motion control, networking, sicurezza/videosorveglianza, transportation e telecom

IL FORMAT

Una mostra convegno con conferenze, dibattiti e workshop pianificati da Assodel con esperti prescelti a fornire lo stato dell'arte che valga a riferimento del settore

Community - area demo per l'incontro tra domanda e offerta di fornitori e utilizzatori

Conference - un filo conduttore di interventi selezionati e dedicati alle novità di prodotto, alle applicazioni tecnico/commerciali, alle case history

Educational - seminari formativi e tutorial applicativo-progettuali per tecnici

EMBEDDED IC & AUTOMATION FORTRONIC
an assodel electronics forum

RF & WIRELESS FORTRONIC
an assodel electronics forum

UNO SGUARDO STRATEGICO AL FUTURO



partecipazione previa registrazione **www.fortronic.it**

an assodel electronics forum
FORTRONIC

Electronics Forum - Fortronic
*il **"miglior costo contatto"***
per qualità ed efficacia

INFORMAZIONI / SEGRETERIA:
Tel 02 210.111.236
marketing@fortronicforum.com

ORGANIZZAZIONE:
TECNO IMPRESE
Via Console Flaminio 19 - 20134 Milano
Tel 02 210.111.1 - **www.tecnoimprese.it**

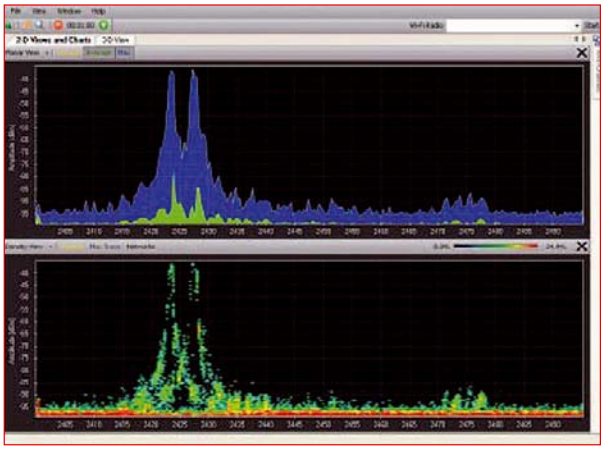
Queste pagine sono dedicate ai suggerimenti, alle richieste ed alle segnalazioni dei lettori. Raccomandiamo, per quanto possibile, di proporre argomenti di interesse generale. Contattateci numerosi all'indirizzo: *redazione@elettronicain.it*

Sui 2,4 GHz, meglio FHSS o DSSS?

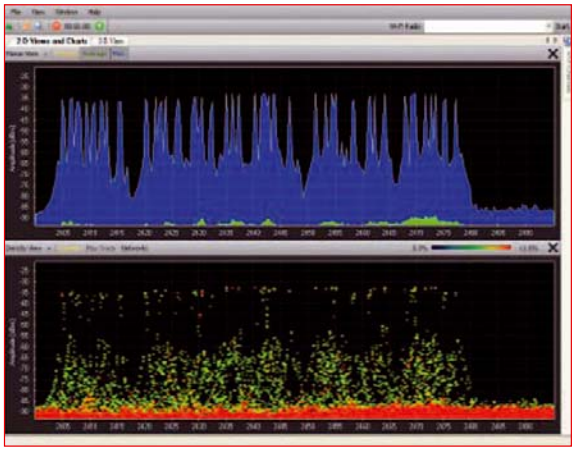
Sono un appassionato aeromodellista e da tempo utilizzo una ricetrasmittente RC di marca che funziona sulla frequenza di 40 MHz. La grossa limitazione è da sempre dovermi sincerare che nessun altro modellista usi il mio stesso canale, pena immancabili interferenze con conseguenze irreparabili per i modelli di entrambi i piloti. Al mio campo gli altri "colleghi" si stanno tutti orientando sull'acquisto di radio RC a 2,4 GHz, sostenendo che in questo modo ci si garantisce da qualsiasi possibilità di interferenza, indipendentemente dal numero di apparati funzionanti contemporaneamente. È vera questa affermazione? Inoltre ho notato che le radio delle maggiori marche che lavorano a 2,4 GHz utilizzano tecniche di gestione della frequenza FHSS e DSSS; sapete consigliarmi verso quale delle due orientarmi?

Massimo Caputi - Rieti

Come puoi immaginare, le risposte alle tue domande non possono esaurirsi in un semplice si o no oppure in "questo tipo di gestione della frequenza è migliore dell'altro"; cercheremo di approfondire gli argomenti utili ad orientarti in una scelta consapevole tra benefici e possibili problemi. Riguardo all'intolleranza alle interferenze sulle ricetrasmittenti RC in FM, il problema, come confermi anche tu, è ben noto. Le trasmittenti RC in FM per uso modellistico lavorano sulle frequenze 35, 40 e 72 MHz; ciascuna di queste bande di frequenza è divisa in canali. Nel collegamento tra RC e modello viene utilizzato un canale fisso, tra quelli disponibili. Mentre stiamo pilotando il nostro modello, se un altro modellista accende una radio sulla stessa frequenza e sul medesimo canale, i suoi comandi si sovrappongono inesorabilmente ai nostri, con le conseguenze che conosci. Il peggio avviene quando un altro modellista fuori dalla nostra vista, per esempio nel cortile di casa sua a mezzo chilometro di distanza, esegue la stessa operazione, oppure se qualcuno aziona il comando radio di un apricancello (ne esistono ancora che lavorano a 40 MHz), oppure se attraversiamo il canale di comunicazione di un ponte radio (anche questi hanno un canale dedicato nei 40 MHz). Questi ultimi sono i casi che fanno discutere al campo volo sulle motivazioni dei crash, interferenza o errore di pilotaggio? A volte le interferenze si prendono colpe non meritate. Da qui l'evoluzione degli apparati verso l'utilizzo della frequenza a 2,4 GHz e la nascita dei sistemi di autoprotezione dalle interferenze. Questi si basano sulla capacità degli apparati di modificare dinamicamente il canale di comunicazione alla ricerca di quelli più liberi da interferenze reciproche. Qui entrano in gioco le sigle che hai citato, che corrispondono a due strategie diverse per risolvere il problema delle interferenze. Il sistema DSSS (Direct Sequence Spread Spectrum) ed il sistema FHSS (Frequency Hopping Spread Spectrum). Il sistema DSSS utilizza, per la comunicazione, set di frequenze predeterminate (canale 1, 6 e 11) nello spettro 2,4÷2,485 GHz. Lo spettro 2,4 Ghz è diviso in 11 canali che condividono 79 frequenze. La necessità di robustezza di comunicazione e di larghezza di banda del sistema DSSS richiede di mantenere una separazione di 30 frequenze tra quelle utilizzate; su un totale di 79 frequenze disponibili, il campo si restringe a tre sole utilizzabili, che sono appunto le frequenze di carrier dei canali 1, 6 e 11. Il sistema DSSS risolve la ricerca delle frequenze libere tra queste tre (in realtà alcune implementazioni ne utilizzano qualcuna in più). La figura in basso a sinistra riporta l'occupazione dello spettro 2,4 GHz di un canale DSSS. Il sistema FHSS invece, come visibile nella figura



Utilizzo dello spettro 2,4 GHz da un canale DSSS.



Copertura dello spettro 2,4 GHz nel sistema FHSS.

in basso a destra, utilizza tutte le 79 frequenze dello spettro nella ricerca di un canale libero da interferenze. In un campo volo in condizioni normali, con meno di 10 ricetrasmittenti accese contemporaneamente, qualche dispositivo Bluetooth (i telefonini) e una connessione Wi-Fi nelle vicinanze, entrambi i sistemi danno buoni risultati; con bassi livelli di interferenze il sistema DSSS garantisce collegamenti a distanza maggiore del FHSS, data la maggiore robustezza del canale di comunicazione. Quando il campo si fa molto più affollato, come durante una manifestazione importante, dove dozzine di radio sono accese contemporaneamente, i sistemi in FHSS continuano a funzionare anche quando i sistemi DSSS hanno perso affidabilità da tempo. Per dare un'indicazione numerica, i sistemi DSSS diventano poco affidabili quando lo spettro è occupato per circa un 20 %, mentre il sistema FHSS si dimostra ancora affidabile con un affollamento dello spettro oltre l'80 %. Ovviamente anche questa affermazione non è vera in assoluto; può anche capitare la combinazione fortunata dove, in uno spettro ultra affollato, un sistema DSSS riesca a centrare l'unico canale assolutamente libero coincidente con uno di quelli che si trovano nella sua lista di preferenze. Per concludere, il mondo e di conseguenza le maggiori case produttrici di sistemi RC a 2,4 GHz si stanno orientando verso i sistemi FHSS; se proprio vuoi optare verso un sistema DSSS, cercane uno in grado di "saltare" tra almeno tre frequenze alternative. In ogni caso, oltre ad una scelta oculata tra i sistemi di comunicazione è bene accertarsi che anche gli altri componenti della catena RC siano scelti ed installati con cura, batterie, antenne, collegamenti alla ricevente, servocomandi.

Fari auto a LED

Ho sempre considerato le luci a LED, montate ormai di serie su moltissimi nuovi modelli di automobile, come abbastanza fastidiose; ultimamente, però ho notato delle auto, sempre con fari a LED, dalla luce assai più gradevole. Sapete darmi qualche delucidazione in merito?

Mino Desdemone - Afragola (NA)

Probabilmente ti stai riferendo alla nuova generazione di fari a LED, che non si limitano più alle luci di posizione ed agli stop, ma realizzano anche





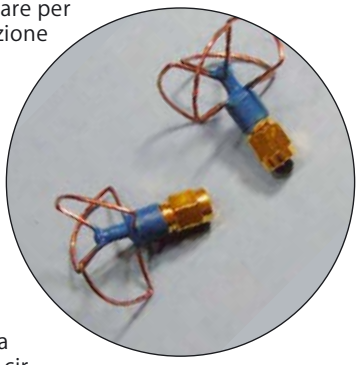
gli anabbaglianti; sono adottati, per esempio, nella nuova Audi A8, che monta un modello realizzato dalla Hella. Nella sezione anabbaglianti, questi fari incorporano dieci lenti di proiezione che direzionano correttamente il fascio di luce. L'effetto che ti ha colpito è il risultato del sistema di controllo AFS (Adaptive Frontlighting Systems), realizzato per la prima volta con i LED, che rendono possibile adattare le emissioni luminose in relazione alle condizioni di guida tramite lo spegnimento o l'accensione di ogni singolo diodo: in caso di maltempo, ad esempio, si riduce l'effetto accecante dovuto all'acqua tramite una diffusione più ampia della luce.

Antenna a trifoglio

Sul mio quadricottero ho montato una piccola videocamera con relativo trasmettitore a 2,4 GHz per l'invio a terra delle immagini raccolte in volo. Il problema è che il segnale ricevuto è abbastanza disturbato, tanto da far apparire interferenze orizzontali sullo schermo anche con il modello non particolarmente distante. Se sostituisco il trasmettitore video con uno più potente, va meglio?

Rossano Biondi - Castelsanpietro Terme (BO)

Al di là dei limiti di potenza imposti dalla normativa vigente (10 mW), la soluzione al tuo problema potrebbe essere l'adozione di un'antenna più efficiente relativamente alle condizioni di impiego su un modello volante. Le antenne classiche a stilo comunemente associate ai trasmettitori video a 2,4 GHz sono caratterizzate da una polarizzazione verticale.
Per la ricevente a terra, le migliori condizioni di ricezione si ottengono quando l'antenna trasmittente è orientata come la ricevente ed alla stessa altezza, cosa che generalmente non si verifica in un modello volante, portando ai difetti di ricezione che lamenti.
Non potendo determinare le posizioni e gli orientamenti relativi delle antenne in un modello in volo, è necessario optare per una configurazione d'antenna che permetta una trasmissione ottimale in ogni posizione e orientamento.



La soluzione migliore è rappresentata dalle antenne a polarizzazione circolare, caratterizzate dal diagramma polare di emissione visibile nelle figure.
Queste antenne si chiamano clover leaf (clover è il trifoglio e queste antenne ricordano la forma delle foglie) e si possono realizzare a tre o a quattro lobi.
Ad una data potenza di trasmissione, queste antenne non aumentano la portata ma permettono di migliorare la trasmissione e la ricezione nelle varie posizioni ed orientamenti, con una diminuzione del segnale proporzionale alla distanza.
Per la loro realizzazione esistono sul web molte istruzioni, con le misure relative alle frequenze più comuni.
Un riferimento tra i tanti è su *www.youtube.com/watch?v=submrMo5QWsrazionontenuti.*

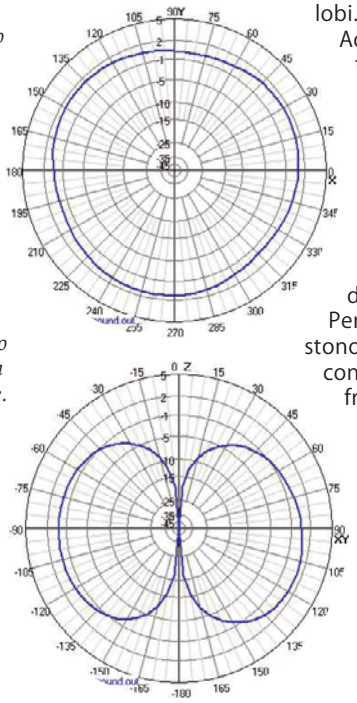
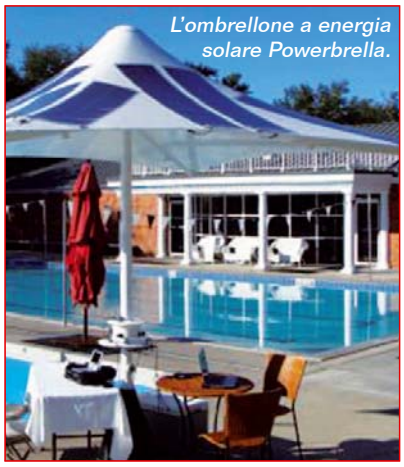


Diagramma polare sul piano orizzontale (in alto) e verticale (in basso) di un'antenna a polarizzazione circolare.

L'ufficio in spiaggia



L'ombrellone a energia solare Powerbrella.

Una delle abitudini che caratterizza il nostro modo di vivere è quella di essere sempre connessi e di utilizzare, per svago e per lavoro, dispositivi come smartphone, tablet e PC. Con l'avvicinarsi dell'estate torna a galla il sogno antico di poter utilizzare la propria dotazione di "appendici" elettroniche sulla spiaggia, sia per vedersi un film o leggersi un e-book su PC, che per consultare l'e-mail, ad esempio dell'ufficio. Ma come si fa per l'alimentazione?

Felice Latazza - Cosenza

Di fatto il "sogno antico" è legato alla possibilità di approvigionarsi di energia per assicurare la ricarica costante delle tue "appendici elettroniche". Dato che sulla spiaggia non dovrebbe mancare il sole, il suggerimento più immediato è di dotarti di un caricabatteria ad energia solare. Una soluzione decisamente più fuori dagli schemi ci viene dagli ideatori degli ombrelloni solari che, oltre a fornire ombra a chi se ne sta comodamente al di sotto, catturano l'energia del sole trasformandola in elettricità. Tra questi, Sun-brella (che sviluppa) 282 W di elettricità, Powerbrella (che cattura l'energia per poi offrirla attraverso alcune prese poste sul suo stelo) e Apsara, una commistione di tessuto e pannelli che la notte illumina le lampade poste nella parte inferiore dell'ombrellone. Ma le ultime novità hanno un marchio italiano: Kosmos (ombrellone) ed Eclipse (parasole) sono stati realizzati da un'idea di Marco Ballini, sviluppata dagli studi di design Extremis e Umbrosa. Questi prodotti sono realizzati con celle fotovoltaiche organiche: flessibili e trasparenti, possono ricavare 60 W di energia elettrica e ricaricare anche dieci iPhone e un computer portatile. Il costo è di 2.500 euro, ma a breve sarà disponibile un kit fai-da-te a "soli" 700 euro.

Vieni con noi a New York con OpenWheels



Maker Faire di San Mateo è arrivato troppo in fretta; visti i tempi effettivamente ristretti, abbiamo deciso di prorogare il termine ultimo per presentare le candidature ad OpenWheels. Se vuoi partecipare, ti comunichiamo che il termine è stato spostato al 15 luglio 2012.
Il vincitore presenterà il proprio progetto al World Maker Faire di New York, in programma il 29 e 30 Settembre 2012. Il viaggio e il soggiorno li offriamo noi.

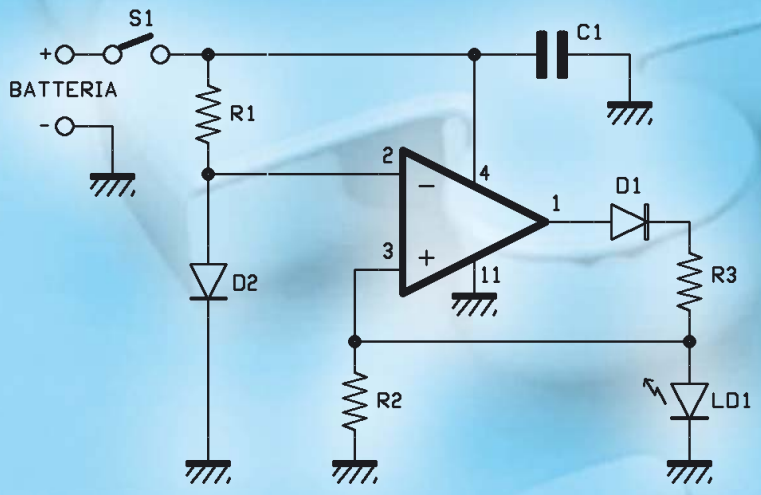
Il progetto OpenWheels

Ricordiamo che il concorso propone ai candidati di progettare l'elettronica di controllo di una sorta di Segway (un veicolo elettrico a due ruote capace di rimanere in equilibrio, avanzare o arretrare, girare e fermarsi) e immaginare applicazioni innovative ed originali di questo mezzo. Quello che vi chiediamo è di realizzare il sistema elettronico di controllo che sia open source e a basso costo; non dovete preoccuparvi della meccanica, perché l'abbiamo già realizzata noi.
Il veicolo deve essere alimentato a 24 volt e la trazione ed il controllo del movimento sono affidati a una coppia di motori da 24 V - 250 W. Il movimento richiesto viene deciso sia dal manubrio (che interviene su un potenziometro da 10 kohm che l'elettronica legge per rilevare le richieste) sia dalla pedana su cui poggia il conducente. Questi dati dovrebbero esservi sufficienti per progettare l'elettronica di controllo che dovrebbe comprendere come minimo: 1) un caricabatterie; 2) una piattaforma inerziale; 3) un sistema di controllo a microcontrollore con relativo firmware; 4) due driver di potenza per l'azionamento dei motori.
Se vuoi partecipare, scrivici all'indirizzo redazione@elettronicain.it dopo aver letto le condizioni di partecipazione nella pagina web *www.elettronicain.it/openwheels*.
Le candidature verranno presentate sul blog della rivista e sottoposte al giudizio dei nostri lettori; ai tre candidati che otterranno i maggiori consensi verranno forniti i componenti necessari alla realizzazione dell'elettronica di controllo della piattaforma OpenWheels. Al vincitore verrà fornita una piattaforma completa OpenWheels sulla quale montare l'elettronica di controllo per sperimentare le altre soluzioni innovative.
Sei un progettista e vuoi metterti alla prova? E allora ...forza! Potresti essere tu a girare sul tuo Segway per i padiglioni del World Maker Faire!



Cosa manca a questo circuito?

Soluzione al quesito del n° 164



Si conclude qui il concorso iniziato nel fascicolo n° 164, con cui vi ponevamo due quesiti: indovinare quale componente completa il circuito qui mostrato e dirci di che circuito si tratta. La risposta esatta più tempestiva ci è stata data dal Sig. Andrea Firpo di Genova, cui va, oltre ai nostri complimenti, un abbonamento annuale omaggio ad Elettronica In. Lasciamo la parola al vincitore, che ci esporrà la sua soluzione.

*"... al circuito manca un diodo al silicio (**nel circuito che vedete in chiaro è chiamato D2, ndr**) polarizzato direttamente che serve a stabilizzare a circa 0,65 volt la tensione sul piedino 2 dell'operazionale, il quale funziona da comparatore non invertente; il piedino 3 è tenuto a livello più basso del 2 dal parallelo di R2 e LD1, che è un LED rosso ad alta efficienza. Quando un intenso fascio di luce di frequenza vicina a quella di emissione della giunzione (per esempio un laser rosso) lo illumina, LD1, per effetto fotovoltaico, produce una tensione fino ad 1 volt, che alza il livello dell'ingresso 3 e conseguentemente fa commutare lo stato dell'uscita (piedino 1) causandone l'accensione; la retroazione del circuito poi lo mantiene acceso fino a che non si apre S1. Il circuito sfrutta la caratteristica della giunzione di LD1 di produrre una tensione se sollecitata dalla luce. Il circuito può essere utilizzato come collimatore laser a bersaglio o come rilevatore di scintilla."*

Web Forum e supporto tecnico

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

www.elettronicain.it/webforum

La casa sull'iPad

Mi incuriosisce lo sviluppo che sta avendo la domotica e in particolare la possibilità di poter tenere sotto controllo i consumi di corrente che nei nuovi contatori portano inesorabilmente all'interruzione della alimentazione se si superano determinati limiti per un dato periodo di tempo. Credo che fino ad ora il maggior limite sia il fatto che le diverse soluzioni usano dispositivi di controllo proprietari e sistemi di gestione poco personalizzabili e integrabili. Esistono soluzioni più aperte, flessibili e configurabili, eventualmente con la possibilità di gestione da remoto?

Saverio Ciccarola - Favara (AG)

La scarsa diffusione dei sistemi di domotica trova spiegazione nell'evoluzione storica delle tecnologie con le quali sono stati via via realizzati. I primi sistemi, anche evoluti e derivati da quelli di controllo dei grandi edifici, erano basati su PC costosi, ingombranti, con alti consumi di corrente, e facilmente vulnerabili nell'affidabilità; quando cadeva l'alimentazione il PC si spegneva e con esso tutto il sistema di controllo. Poi si è passati a sistemi basati su microcontrollori, firmware e dispositivi periferici proprietari: una volta scelta una soluzione si era legati a doppio filo per tutta la vita operativa dell'impianto, con relativi rischi di obsolescenza e di possibilità di ampliamento e ammodernamento. Attualmente, invece, la tendenza che trova sempre più seguaci tra i produttori di tecnologie di controllo industriali e domestiche è l'apertura dei propri sistemi al controllo remoto e l'integrazione sempre più spinta con i dispositivi mobile.
La portata del fenomeno è facilmente riscontrabile consultando la stampa spe-





cializzata e frequentando le fiere di settore.
Nel tuo caso di curiosità "domotica", una soluzione che ci ha particolarmente attratto è quella proposta da ERGO Design & Technology, che ha reso fruibile il sistema di gestione della casa su iPad, sia dall'interno che dall'esterno. Il motore della soluzione è il software di gestione Ergo Server, che permette di configurare ed indirizzare i vari servizi all'interno dell'abitazione, quali lampade, elettrodomestici, televisore. Al termine della configurazione, viene generato l'ambiente di controllo gestibile sia dall'iPad che via web per mezzo di un'interfaccia grafica che riproduce la pianta dell'appartamento e i diversi oggetti al suo interno.
In aggiunta, l'iPad entra a far parte fisicamente dell'arredamento mediante l'iDock, un dispositivo a muro che permette di integrare l'iPad in una parete, garantendone la ricarica e la connessione in rete.
La stessa tecnologia è in corso di adozione anche in alcuni alberghi, dove il cliente, al suo arrivo nella stanza, posiziona il suo iPad nel supporto a muro iDock rendendo, ad esempio, disponibili i suoi contenuti multimediali sul televisore della stanza.

A proposito del Programmatore HV modulare

Buongiorno, sarei intenzionato a realizzare il Programmatore HV Modulare presentato nel nº 165 della rivista, ma ho trovato delle incongruenze tra il circuito elettrico e la descrizione dello stesso, inoltre ho dei dubbi sul circuito stampato della sezione master. Potete dirmi se ci sono state revisioni del progetto?

Fabrizio Piacentini - Ravenna

Gentile lettore, hai pienamente ragione; ci siamo accorti dei refusi a rivista già in fase di stampa, per questo abbiamo messo a disposizione gratuitamente sul nostro sito la versione corretta dell'articolo del Programmatore HV. Su *www.elettronicain.it* trovi disponibile per il download anche tutta la documentazione aggiornata per realizzare le basette dei circuiti stampati (il circuito master è effettivamente a doppia faccia) quindi le tracce lato rame e i piani di montaggio esatti.

FIERE ELETTRONICA INFORMATICA e RADIANTISMO
con Mercatino Radio Amatoriale e Surplus
FIERA DI FERRARA
Via della Fiera, 11 - Autostrada A13 uscita "Ferrara sud"
16-17 GIUGNO
ORARIO CONTINUATO 9.00 - 19.00
PROSSIMI APPUNTAMENTI
15-16 SETTEMBRE FIERA DI PIACENZA PIACENZA EXPO' - LOCALITA' LE MOSE
29-30 SETTEMBRE RAVENNA PALA DE ANDRE'
13-14 OTTOBRE FIERA DI FERRARA VIA DELLA FIERA 11
3-4 NOVEMBRE BOLOGNA PALA NORD - PARCONORD
15-16 DICEMBRE FIERA DI PIACENZA PIACENZA EXPO' - LOCALITA' LE MOSE
ELETTRONICA IN
Cell. 348 9030832 - www.mondoelettronica.net
Cosmohelp
VALE COME RIDOTTO ANCHE IN FOTOCOPIA

La Più Prestigiosa Rivista di Progettazione Elettronica

per un anno, direttamente a casa tua!

Campagna abbonamenti

Ei 2012



Solo per te ...

- L'abbonamento annuo (di 10 numeri) costa € 45,00 anzichè € 60,00 con un risparmio del 20% sul prezzo di copertina.
- È il massimo della comodità: ricevi la rivista direttamente al tuo domicilio, senza scomodarti a cercarla e senza preoccuparti se il numero fosse esaurito.
- Se il prezzo di copertina della rivista dovesse aumentare nel corso dell'abbonamento, non dovrai preoccuparti: il prezzo per te è bloccato!
- Hai a disposizione un servizio di consulenza: i nostri tecnici sono a tua disposizione per fornirti tutte le informazioni necessarie riguardanti i progetti pubblicati.

Abbonarsi è facilissimo...

- **ON-LINE**
  compilando il modulo riportato nella pagina "Abbonamenti" del nostro sito www.elettronicain.it Se possiedi una carta di credito potrai effettuare il pagamento contestualmente alla richiesta.
  È anche possibile abbonarsi on-line richiedendo il pagamento tramite C/C postale.

- **COMPILANDO ED INVIANDO VIA POSTA O FAX IL MODULO DI ABBONAMENTO RIPORTATO A PIE' DI PAGINA**
  Riceverai direttamente a casa tua un bollettino personalizzato di C/C postale. L'abbonamento decorrerà dal primo numero raggiungibile.

  PER I RINNOVI ATTENDERE NOSTRO AVVISO

Elettronica In propone mensilmente - oltre a corsi, articoli didattici e notizie dal mondo dell'elettronica - tantissimi progetti tecnologicamente molto avanzati, sia dal punto di vista hardware che firmware, cercando di illustrare nella forma più chiara e comprensibile le modalità di funzionamento, le particolarità costruttive e le problematiche firmware dei circuiti proposti.

NEW

Scoprila anche su

iPad



È disponibile per gli abbonati anche la versione digitale della rivista da sfogliare e leggere on-line.



ABBONAMENTO annuale SOLO € 45,00 + OMAGGIO

Sottoscrivendo l'abbonamento annuo alla rivista riceverai in omaggio la DISCOUNT CARD



che ti consentirà di usufruire di uno sconto del 10% su tutti i prodotti acquistati direttamente (oppure on-line) da Futura Elettronica.



Gli abbonati riceveranno anche in regalo ...

- I DVD con tutti i numeri della rivista dal n. 1 al n. 123.
- Un volume a scelta della collana "L'ELETTRONICA PER TUTTI".
- La possibilità di scaricare gratuitamente dal sito www.elettronicain.it i file sorgente dei firmware utilizzati in molti dei progetti pubblicati.



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

☒ **DISCOUNT CARD "Futura Elettronica"**

Uno, a scelta, tra questi volumi della collana "L'Elettronica per tutti".

- ☐ Programmiamo con i PIC
- ☐ Alla scoperta della CCTV
- ☐ Tecnologia e apparati GPS
- ☐ Energie alternative
- ☐ Primi passi in elettronica
- ☐ Top 100 Arduino Projects

Resto in attesa del primo numero e degli omaggi.

Nome: Cognome:

Via: N°:

CAP: Città: Provincia:

Telefono: e-mail:

Data: Firma:

Resto in attesa di vostre disposizioni per il pagamento.

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

VISPA Edizioni

VISPA snc
Via Adige 11
21013 Gallarate (VA)
Fax: 0331-778112

✓ AUTO
✓ CAMPER
✓ MOTO

Fornisci la giusta carica alle tue batterie!

Utilizza questi caricabatterie automatici, di ultima generazione, per ricaricare in modo estremamente semplice e sicuro la batteria dell'auto, della moto o del camper. Estremamente efficienti e affidabili, sono protetti contro errori di utilizzo ed in grado di eseguire in totale autonomia la diagnosi, la rigenerazione, la carica ed il mantenimento.

CARICABATTERIE PER BATTERIE AL PIOMBO DA 6 E 12V

- Adatto per batterie piombo-acido ricaricabili con capacità da 1,2 a 12Ah (6V) e da 1,2 a 60Ah (12V);
- Corrente di carica ad impulsi 1A - 0,6A - 0,2A - 0,1A;
- Carica automatica in 5 fasi;
- Funzione per la carica della batteria nei periodi invernali;
- Protezione contro cortocircuito, inversione di polarità e surriscaldamento.

€ 46,00
cod. BX-3



€ 58,00
cod. BX-1

CARICABATTERIE PER BATTERIE AL PIOMBO DA 12V

- Adatto per batterie piombo-acido ricaricabili con capacità da 1,2 a 120Ah;
- Corrente di carica ad impulsi 3,8A - 0,8A;
- Carica automatica in 5 fasi;
- Funzione per la carica della batteria nei periodi invernali;
- Protezione contro cortocircuito, inversione di polarità e surriscaldamento.

Prezzi IVA inclusa.

FUTURA ELETTRONICA
www.futurashop.it

Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Maggiori informazioni su questi prodotti sono disponibili su *www.futurashop.it* tramite il quale è anche possibile effettuare acquisti on-line.

■ 19-21 GIUGNO 2012

Photonics Festival

Taiwan

OPTO Taiwan
The 21st Int'l Optoelectronics Exposition

LED Lighting Taiwan
The 8th Int'l LED Lighting Exposition

Solar Taiwan
The 6th Int'l Photovoltaic Exposition

OptoComm Taiwan
6th Int'l Fiber Communication & Broadband Exposition

Opto, LED, Solar e OptoComm: quattro eventi in uno vi aspettano al Photonics Festival di Taiwan, una manifestazione fieristica che si rivolge al mondo dell'illuminazione a LED, dell'optoelettronica in generale, dello sfruttamento dell'energia solare e delle comunicazioni in fibra ottica.

Dal 1984, Opto Taiwan rappresenta l'evento annuale che riunisce settori dell'industria di Taiwan quali quello dei semiconduttori, dell'elettronica, della comunicazione in fibra ottica, dei laser, dell'ottica e dei macchinari di precisione, nonché degli apparati per l'automazione di fabbrica e delle tecnologie software/hardware per le applicazioni di fotonica.

LED è invece l'evento di Photonics Festival dedicato all'illuminazione a LED, mercato che per la sola regione di Taiwan varrà quest'anno qualcosa come 600 milioni di dollari U.S.A. Nei tre giorni di manifestazione, LED Lighting Taiwan presenta la filiera più completa dal chip al confezionamento delle applicazioni finali di sistemi d'illuminazione a LED.

Non meno importante è Solar, la parte di Photonics Festival riservata al settore del fotovoltaico, il cui valore per Taiwan si aggira intorno ai 6,3 miliardi di dollari e raggiungerà presto un nuovo picco, consolidando il ruolo dei produttori di Taiwan nel mercato mondiale del fotovoltaico. Solar si distingue da altre manifestazioni simili perché coinvolge i produttori che adottano tutte le tecnologie per il fotovoltaico: dal monocristallino in silicio, al film sottile, all'HCPV (fotovoltaico a concentrazione) permettendo loro di dimostrare le ultime soluzioni tecnologiche sviluppate, i materiali, i componenti, le attrezzature, agli acquirenti e in generale ai visitatori. Durante l'evento, l'organizzazione ospiterà conferenze e seminari tenuti da espositori per creare le maggiori opportunità di business per l'industria del fotovoltaico.

Infine, OptoComm affronterà le problematiche connesse alle crescenti esigenze di larghezza di banda applicate a Internet e al cloud computing, che hanno incrementato la richiesta di sistemi di comunicazione in fibra ottica (OFC); in mostra, cavi in fibra ottica, sistemi laser con ricevitore ottico, PON e attrezzature. OptoComm Taiwan raccoglie i fornitori di tutti i tipi di materiali per OFC, componenti e apparecchiature di comunicazione Internet.

www.optotaiwan.com/en/

■ 13-15 GIUGNO 2012

Intersolar

Monaco • Germania

Nata a Pforzheim nel 1993 da un'iniziativa Working Group for Environmental Technology, è oggi la più importante manifestazione mondiale specializzata nel fotovoltaico, nell'energia solare termica e nell'architettura solare, aperta ad espositori e visitatori di tutta Europa attenti alle tematiche delle rinnovabili da fonte solare. All'evento, punto di riferimento per il mercato del fotovoltaico in Germania e nel resto del mondo, quest'anno sono attesi oltre 80.000 visitatori professionali da 150 Paesi, che, in un'area espositiva di 170.000 mq ricavata nei padiglioni del Munich Trade Fair, potranno vedere e toccare con mano lo stato dell'arte della tecnologia del solare termico e fotovoltaico, esposta dai 2.000 espositori previsti.

Accanto all'esposizione vera e propria (nell'International Congress Centre Munich) in quasi concomitanza (dall'11 al 14 giugno) si terrà un ciclo di conferenze sul tema dell'energia solare, dove si avvicenderanno 400 relatori.

Sull'onda del successo dello scorso anno, Intersolar torna ad ospitare PV ENERGY WORLD Special Exhibit: insieme al Fraunhofer Institute for Solar Energy Systems, Intersolar Europe presenta le ultime tecnologie di Electricity Storage e Grid Integration, offrendo uno scorcio sul mondo dell'elettricità del futuro. Inoltre, nell'ambito del PV ENERGY WORLD sarà prevista l'area tematica "PV and E-Mobility" (che prenderà posto all'esterno dell'area espositiva della Fiera di Monaco) che verterà sui veicoli elettrici e le soluzioni integrate di ricovero e ricarica. La German Solar Industry Association (BSW-Solar) di Berlino organizzerà una speciale sessione espositiva dedicata alla Rural Electrification, dove 20 aziende esporranno le proprie tecnologie in fatto di elettrificazione delle zone non raggiunte dalla rete elettrica, dove il fotovoltaico gioca un ruolo vitale e può giocarlo per lo sviluppo di intere regioni dell'Africa, dell'Asia e dell'America Latina.

www.intersolar.de



MOSTRE MERCATO

SORA (FR)
2ª MOSTRA DELL'ELETTRONICA E DEL RADIOAMATORE
Fiere di Sora, Via Barca San Domenico 1, Sora (FR)
Organizzazione: Fiere di Sora
Tel: 0776813179 - Fax: 0776813799
info@fieredisora.it – www.fieredisora.it
02 e 03 Giugno 2012

BORGO D'ALE (VC)
ELETTRONICA BORGO D'ALE
Quartiere fieristico, Borgo D'Ale (VC)
Organizzazione: EVENTI 2000
Tel: 3801808746 – Fax: 0270036123
afi2006.org@gmail.com – www.eventi2000.com
02 e 03 Giugno 2012

FIUMICINO (RM)
FIERA DI ELETTRONICA E MODELLISMO
Palazzetto dello Sport, Viale Danubio, Fiumicino (RM)
Organizzazione: Millenniumfiere
Tel: 0696840758
info@millenniumfiere.it - www.millenniumfiere.it
09 e 10 Giugno 2012

LONGARONE (BL)
COMPUTERFEST & RADIOAMATORE
Quartiere Fieristico, Via del Parco 3, Longarone
Organizzazione: Compendio Fiere
Tel: 3482105317 - Fax: 0572498205
info@compendiofiere.it – www.compendiofiere.it
09 e 10 Giugno 2012

ROSETO DEGLI ABRUZZI (TE)
MOSTRA MERCATO DEL RADIOAMATORE E DELLE TELECOMUNICAZIONI
Quartiere Fieristico, Roseto degli Abruzzi (TE)
Organizzazione: ARI Roseto
Tel: 0858931033 - Fax: 0858931033
info@ariroseto.it – www.ariroseto.it
16 e 17 Giugno 2012

FERRARA
MONDO ELETTRONICA
Quartiere Fieristico, Ferrara
Organizzazione: Exposition Service
Tel: 054527548 - Fax: 0545291179
info@expositionservice.it
www.mondoelettronica.net
16 e 17 Giugno 2012

FRIEDRICHSHAFEN (Germania)
HAM RADIO - MESSE FRIEDRICHSHAFEN
Quartiere Fieristico Messe Friedrichshafen
Organizzazione: Messe Friedrichshafen Gmbh
Tel: +49 75417084O4 - Fax: +49 754170824O5
www.hamradio-friedrichshafen.de
22, 23 e 24 Giugno 2012

MARTINA FRANCA (TA)
MOSTRA DELL'ELETTRONICA DELLA VALLE D'ITRIA
Foro Boario – Zona Ortolini
Organizzazione: SPF Srl
Tel: 3342859686 - Fax: 0804305761
spffiere@legalmail.it – www.ansefiere.it
22, 23 e 24 Giugno 2012

L'elenco aggiornato di tutte le Mostre Mercato del 2012 è disponibile sul sito *www.elettronicain.it* sul quale è anche possibile scrivere un commento sulle fiere visitate.

■ 26 GIUGNO 2012

S&PI – Sensors & Process Instrumentation

Milano

Primo evento italiano one-day interamente dedicato alla sensoristica e alla strumentazione di processo, è una mostra convegno dove, accanto all'esposizione di prodotti e soluzioni dei top player del settore, si terranno convegni su temi come wireless, sistemi di energy harvesting, diagnostica, sensoristica avanzata.
S&PI–Sensors & Process Instrumentation, proporrà un'esclusiva vetrina di prodotti, sistemi e soluzioni che trovano applicazione negli impianti e macchine per l'industria manifatturiera e di processo. Organizzata dalla rivista Automazione e Strumentazione in collaborazione con GISI, gode del supporto delle principali associazioni di utilizzatori e di media partner qualificati ed è dedicata a tecnologie in grande fermento, il cui sviluppo ha aperto nuove e interessanti prospettive a livello applicativo. Secondo i dati dell'Osservatorio sull'andamento del Mercato Nazionale Automazione e Strumentazione Industriale, Civile e di Laboratorio (realizzato da GISI e SDA Bocconi), in Italia queste tecnologie hanno generato nel 2011 un fatturato superiore a 600 milioni di euro.
S&PI–Sensors & Process Instrumentation proporrà ai visitatori un'esclusiva rassegna di sensori, trasduttori, trasmettitori, indicatori, valvole di regolazione, registratori, regolatori, controllori, attuatori, strumenti e sistemi completi per la misura di tutti i fondamentali parametri di processo: temperatura, pressione, livello, flusso, portata, umidità, forza, peso ecc. Accanto alle soluzioni per le misure di processo, uno spazio speciale sarà dedicato alla strumentazione per l'analisi e il laboratorio: apparecchiature e strumenti per determinare il pH, la torbidità, la conducibilità, la presenza di ossigeno, analizzatori di TOC, titolatori, gascromatografi, spettrofotometri, densimetri e viscosimetri, campionatori ecc.
La mostra-convegno si terrà presso il centro congressi Cosmo Hotel di Cinisello Balsamo (MI); la partecipazione è gratuita, ma bisogna registrarsi in anticipo sul sito della manifestazione.
www.mostreconvegno.it/sepi

■ 6-7 GIUGNO 2012

SMAU BUSINESS

Bologna

Smau è il più importante momento d'incontro tra i principali fornitori di soluzioni ICT, gli imprenditori e i decisori aziendali delle imprese e delle pubbliche amministrazioni, con il mondo del canale ICT; un appuntamento in cui vengono raccontate tutte le principali novità in ambito tecnologico e le implicazioni per il business derivanti dall'adozione di tecnologie digitali.
L'evento di Bologna si inquadra nel più ampio progetto Smau Business Roadshow 2012, un format ideato congiuntamente agli operatori dell'Ecosistema ICT (vendor, canale, università e business school, stampa specializzata) e alle Associazioni dell'Industria e del Commercio (territoriali di Confindustria e locali di Ascom) per consentire alle imprese del comparto dell'Information & Communication Technology partecipanti di generare nuove opportunità di business con le imprese e pubbliche amministrazioni locali, sviluppare e/o rafforzare la propria brand awareness in modo capillare sul territorio, incontrare i partner ICT locali e reclu-

L'elenco completo dei più importanti eventi nazionali e internazionali di elettronica, sicurezza e fonti rinnovabili è disponibile sul nostro sito: www.elettronicain.it/MyMain/riviste/p_rivista.asp?pagina=fiereeventi



smau business 12
BOLOGNA 6-7 GIUGNO

tare nuovi operatori trade per ampliare il proprio network.
Ciascun visitatore, attraverso un articolato sistema di profilazione online, costruisce la propria agenda di visita ad ogni evento sulla base delle proprie esigenze di business, individuando i momenti formativi e informativi di proprio interesse (tra gli oltre 30 workshop da 50 minuti a cura dei migliori docenti e analisti indipendenti) i casi di successo di imprese italiane utenti nei bacini di riferimento (Premio Innovazione ICT) i migliori fornitori di tecnologie riuniti in un unico appuntamento per orientare il visitatore nei suoi investimenti ICT.
Ultimo appuntamento del Roadshow 2012, quello di Bologna si focalizzerà sui vantaggi per il business derivanti dall'utilizzo di tecnologie come cloud computing, tablet e app. Non mancheranno momenti formativi dedicati alle unified communication & collaboration, alla virtualizzazione e alle applicazioni di business analytics, alla fatturazione elettronica, al web 2.0, al marketing digitale e all'innovazione nel retail raccontati da docenti e analisti delle più prestigiose business school italiane, che si alterneranno nelle arene dislocate nei padiglioni.
Per aggiornare gli operatori del settore sugli ultimi cambiamenti in atto, è in programma Smau Trade, la dimensione del Salone rivolta agli operatori del canale ICT (rivenditori, VAR, ISV, system integrator, software house, ecc.) che presenterà percorsi formativi ad hoc con workshop tenuti da analisti, docenti e dai player del comparto, al termine dei quali i visitatori riceveranno un attestato di partecipazione. Presente anche un'area dedicata agli incontri one-to-one che offrirà agli addetti ai lavori un'importante occasione per fare networking.
Sul modello dell'esperienza milanese, verrà riproposta anche Smart City, l'iniziativa dedicata alle città intelligenti focalizzata sui progetti in corso e le best practice per migliorare la qualità della vita urbana.
www.smau.it



■ **21-22 GIUGNO 2012**

ALL DIGITAL
Vicenza

TV, reti, casa, cinema: sono le quattro grandi aree tematiche che animeranno la terza edizione di AllDigital, in programma il 21 e 22 giugno 2012 in Fiera a Vicenza.
La due giorni dell'Expo-Forum professionale, dedicato alle tecnologie digitali, coincide con l'ultima tranche dello switch-off analogico che segnerà il definitivo passaggio della televisione italiana all'era del digitale; un traguardo importante che AllDigital taglia assieme a tutti i professionisti del settore, per gettare le basi di nuovi modelli di business quali sistemi integrati multiutenza, home and building automation, sicurezza residenziale, network security, reti wireless e applicazioni wi-fi. AllDigital si ripresenta come il luogo di incontro per eccellenza dei big player e delle associazioni del settore con i tecnici installatori, e riconferma la sua triplice formula: espositiva, convegnistica e formativa.
Al centro dell'Expo, le novità e le anteprime sulla produzione, commercializzazione e installazione dei più aggiornati dispositivi per sicurezza, domotica, entertainment e interattività, ma anche TV digitale, servizi a banda larga, sistemi di smart home e cinema digitale via satellite (tecnologie HD e 3D).
Accanto all'area espositiva, i partecipanti troveranno seminari e convegni tenuti da esperti e tecnici di enti e aziende del settore, che verteranno su temi come l'evoluzione della TV con l'avvento dell'LTE, l'alta definizione e il cinema 3D, l'installazione di sistemi cablati negli edifici, le reti wireless e i sistemi multiutenza, ed altro ancora.
www.alldigitalexpo.it



Dai LUCE al tuo tempo libero!



Lampada a LED DC/AC ricaricabile



Efficiente e a basso consumo, può essere utilizzata come normalissima lampada o come torcia. Alimentazione 85-265Vac; 32 LED; 220 lumen; potenza 3W; Ø70mm. Non dimmerabile.

€29,00
cod. LED-ADC-02


Attacco E27


Impugnatura telescopica

Lampada ricaricabile magnetica a 27 LED



- Impermeabile
- Base magnetica
- Gancio estraibile
- Con caricabatterie da auto e da rete

€38,00
cod. ZLLWL2

Lampada multiuso magnetica a 72 LED



- 72 LED ad alta luminosità
- Base magnetica
- Gancio estraibile

€13,00
cod. EWL27

Lampada a LED con timer



- 18 LED bianchi
- Timer 15-30-60 minuti
- Funzione luce notturna
- Luminosità regolabile
- Gancio estraibile

€11,00
cod. EFLC3

Lampada multifunzione da campeggio



- Lanterna 6 LED bianchi
- Lanterna LED giallo
- Torcia 3 LED bianchi

€11,00
cod. ZLLC6

Mini lanterna a 5 LED alta intensità



- Lanterna 5 LED bianchi
- Torcia 1 LED bianco

€7,50
cod. ZLLC4

Lanterna a LED con radio FM



- 6 LED bianchi e 6 LED rossi lampeggianti
- Caricabatteria a dinamo
- Radio FM e sirena

€24,00
cod. EDL4

Prezzi IVA inclusa.

FUTURA ELETTRONICA

www.futurashop.it

Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Maggiori informazioni su questi prodotti e acquisti on-line su *www.futurashop.it*.

Gate driver ad alta tensione per applicazioni automotive da Fairchild

I progettisti che operano in campo automotive devono affrontare le sfide legate alla realizzazione di inverter più efficienti, con una tensione massima sempre più elevata ed una superiore immunità al rumore, specialmente nei veicoli ibridi (HEV) ed elettrici (EV).
Per risolvere questi ostacoli Fairchild Semiconductor ha sviluppato il gate driver ad alta tensione high-side FAN7171 e il gate driver ad alta tensione high- e low-side FAN7190.
I dispositivi FAN7171 e FAN7190 mettono a disposizione un inverter integrato più efficiente dalle tensioni più elevate e con le garanzie di robustezza richieste dalle applicazioni automotive.
Il modello FAN7171 è un gate driver high-side monolitico in grado di azionare MOSFET e IGBT ad alta velocità



funzionanti fino a +500V, mentre il dispositivo FAN7190 può pilotare MOSFET e IGBT operanti fino a +600V. Entrambi i dispositivi dispongono di uno stadio di uscita bufferizzato con tutti i transistor NMOS progettati per lavorare con impulsi di corrente elevata minimizzando i rischi di cross-conduction. Una immunità al rumore con un Voltage Swing (VS) negativo di -9,8V a 15V VBS aumenta l'affidabilità del design e la vita utile del sistema. Qualificati secondo gli standard AEC Q100 Automotive Class 1, i dispositivi FAN7171 e FAN7190 offrono elevata integrazione e una maggior gamma di funzionalità consentendo di ridurre il numero di componenti, il costo dei materiali, lo spazio occupato su scheda, e potenzialmente anche i tempi di progettazione.
Un circuito di level-shift avanzato consente al gate driver high-side di funzionare fino a VS = -9,8V (tipico) per VBS = 15V. Un circuito UVLO (Under Voltage Lockout) evita i malfunzionamenti quando la VBS scende sotto la soglia di tensione specificata. Entrambi i dispositivi sono prodotti in un package SOP (Small Outline Package) a 8 pin e sono conformi RoHS.
www.fairchildsemi.com

Intel Annuncia le Unità Disco a stato solido (SSD)



Disponibili nelle capacità più richieste, da 60 GB, 120 GB e 180 GB, le unità Disco a stato solido SSD Intel serie 330 incrementano le prestazioni complessive di sistema e offrono tempi di risposta più rapidi per un'ampia gamma di applicazioni. L'interfaccia SATA a 6 Gb/s offre una larghezza di banda doppia rispetto agli attuali SSD Intel® serie 320 basati su SATA a 3 Gb/s, con velocità fino a 500 Megabyte al secondo (MB/s) per le operazioni di lettura sequenziale e fino a 450 MB/s per le operazioni di scrittura sequenziale, accelerando i trasferimenti di dati. Le prestazioni possono raggiungere fino a 22.500 IOPS (Input-Output Operations Per Second) in lettura casuale e 33.000 IOPS in scrittura.
www.intel.com/go/ssd

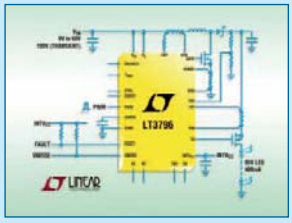
Transceiver RF Atmel a 2.4GHz per applicazioni industriali a consumi ridotti

Il nuovo transceiver Atmel AT86RF233, con un consumo ridotto del 60% rispetto ai suoi concorrenti, si posiziona come soluzione ottimale nelle applicazioni alimentate a batteria come i misuratori di consumi, i sistemi industriali di monitoraggio e controllo ed i dispositivi in tecnologia Energy-Harvesting. L'Atmel AT86RF233 è compatibile con gli standard ZigBee/IEEE 802.15.4, IPv6 over low-power wireless personal area networks (6LoWPAN) e con tutte le applicazioni industriali, scientifiche, medicali (ISM) nella banda 2.4GHz. In trasmissione assorbe 14mA di corrente, in ricezione 6mA mentre l'assorbimento a riposo è di soli 0.02uA, 20 volte meno delle soluzioni finora presenti sul mercato.
www.atmel.com

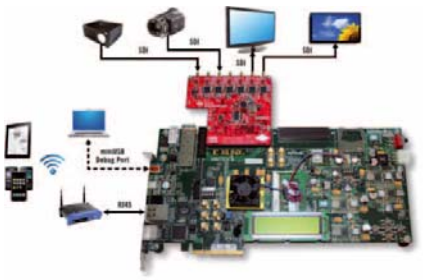


LT3796, controller a corrente e tensione costanti da 100V

Linear Technology annuncia l'LT3796, un controller DC/DC da 100V con rilevamento della corrente lato fronte alto, progettato per regolare una corrente o una tensione costante, ideale per i LED ad alta luminosità (HB). L'intervallo di tensioni di ingresso compreso tra 6V e 100V supporta una vasta gamma di applicazioni, inclusa l'illuminazione architettonica e del settore automobilistico e industriale. L'LT3796 utilizza un MOSFET esterno a canale N ed è in grado di pilotare LED bianchi fino a 85V da un ingresso nominale da 12V, per oltre 50 Watt. L'LT3796 è ideale per caricare le batterie o i supercondensatori a corrente e tensione costanti. Integrando il rilevamento della corrente sul lato alto, questo nuovo dispositivo può essere utilizzato in modalità boost, buck o buck-boost o per topologie SEPIC e flyback. Inoltre, offre sia la limitazione che il monitoraggio della corrente di ingresso e di uscita per migliorare ulteriormente l'affidabilità e la flessibilità di progettazione. L'LT3796 ha un rendimento superiore al 94% in modalità boost, eliminando la necessità di installare dissipatori di calore esterni. Un pin di regolazione consente all'utente di programmare la frequenza tra 100kHz e 1MHz per ottimizzare il rendimento, riducendo al tempo stesso la dimensione e il costo dei componenti esterni. Offerto in un package TSSOP-28 con funzionalità termiche avanzate, l'LT3796 rappresenta una soluzione driver per LED HB estremamente compatta.



www.linear.com



Il motore Video Real-Time di Xilinx riduce i tempi di sviluppo della nuova generazione di dispositivi di gestione video

L'architettura del Video Real_Time (RTVE) permette ai costruttori di dispositivi di potenziare i processori video IP LogiCORE(TM) di Xilinx per mettere a punto dispositivi di video-processing intensivo. La architettura RTVE rende compatibili agli standard AMBA® AXI-4 i dispositivi FPGA Kintex(TM)-7 per interconnettersi all'ultima generazione di multiviewer.
Le FPGA Kintex-7 sono dispositivi da 28nm che offrono il doppio di potenza elaborativa con consumo dimezzato rispetto alla generazione precedente di FPGA. La architettura RTVE è ottimizzata per realizzare switcher e router broadcast, multiviewers e sistemi di visione, permette di elaborare due canali video, offre funzionalità grafiche basate su web, e qualità video con le seguenti caratteristiche: data path video a 10-bit (VDP), Resampler Chroma OmniTek, matrice di correzione colore OmniTek (CCM) e supporto per il processing nativo RGB.
www.xilinx.com/applications/broadcast/brtve-tdp/index.htm

Nuovo ricetrasmettitore wireless SPIRIT1 di ST per contatori intelligenti



STMicroelectronics ha presentato un ricetrasmettitore wireless ad alte prestazioni per le applicazioni Automatic Meter Infrastructure (infrastruttura per la gestione dei contatori elettronici intelligenti). La possibilità di un controllo remoto completo, analisi dei consumi e della potenza di picco, meccanismi anticontraffazione, segnalazione dei guasti, tariffazione differenziata in base alle fasce orarie, permette di garantire una fornitura e un utilizzo più efficiente ed economico di elettricità, gas e acqua.
Il ricetrasmettitore SPIRIT1 utilizza frequenze inferiori a 1 GHz per la trasmissione su brevi distanze e garantisce che il segnale wireless possa attraversare anche la struttura di un edificio e le pareti in cemento armato.
SPIRIT1 offre un consumo di corrente ridotto del 50% rispetto alle alternative disponibili in commercio. In questo modo, le applicazioni possono addirittura raddoppiare il tempo di funzionamento rispetto ad altre soluzioni, senza che sia necessario sostituire le batterie.
SPIRIT1 vanta una sensibilità eccezionale in ricezione pari a -120dBm. Ciò permette al progettista di ridurre la potenza d'uscita del trasmettitore diminuendo ancora il consumo complessivo di corrente e garantendo comunque un canale di comunicazione molto robusto. Il ricetrasmettitore supporta il "frequency hopping", l'auto-acknowledgment e la possibilità di utilizzare diverse antenne per garantire una trasmissione dei dati priva di errori.
www.st.com

Sharp presenta la nuova generazione dei suoi LED Mini Zenigata



La nuova generazione di prodotti è basata su una piccola piastra realizzata in ceramica tecnica (ossido di alluminio Al2O3) delle dimensioni di 15 mm x 12 mm x 1,6 mm. L'efficienza è stata migliorata del 11% grazie ai progressi tecnici. La nuova generazione di sistemi di illuminazione da 4 e da 7 W è caratterizzata da quattro proprietà: compattezza, leggerezza, convenienza e luminosità visibilmente migliore di prima.
A seconda del modulo, i nuovi modelli garantiscono un flusso luminoso fino a 585 lumen, con un'efficienza luminosa fino a 91 lm/W, un elevato valore IRC pari a 82 e una vita di servizio di 40.000 ore a una temperatura di esercizio fino a 90 °C.
La nuova generazione di sistemi di illuminazione da 4 W ha una tensione diretta di 9,6 V e una corrente costante pari a 400 mA, mentre quelli da 7 W hanno una tensione di 12,3 V e una corrente pari a 520 mA. Tuttavia, i sistemi di illuminazione da 4 W possono anche funzionare con una corrente costante di 500 mA, cosa che li rende in grado di raggiungere un flusso luminoso massimo di 400 lm. Con i sistemi di illuminazione da 7 W, può essere raggiunto un flusso luminoso fino a 830 lumen in condizioni di funzionamento con una corrente pari a 680 mA. Le temperature di colore delle nuove serie di LED a luce bianca raggiungono valori da 2.700 a 6.500 Kelvin, con le gradazioni "bianco naturale", "bianco caldo" e "bianco puro".
I moduli Mini Zenigata sono adatti per applicazioni come punti luce, e lampade retrofit a LED.
www.sharpsme.com

MCP3911, nuovo Front-End analogico Microchip con ADC a 24-bit

Il nuovo dispositivo MCP3911 AFE (analogue front end) è dotato di due ADC delta-sigma a 24-bit che operano a 3V e che garantiscono i migliori livelli di precisione attualmente disponibili in commercio: SINAD di 94,5 dB e THD di 106,5 dB. La disponibilità di quattro modalità operative offre la flessibilità necessaria per realizzare progetti con consumi estremamente contenuti – fino a 0,8 mA per canale – o destinati a gestire contenuti armonici e segnali ad altissima velocità. Il range di temperatura consente di operare da -40°C a +125°C. L' MCP3911 ha la possibilità di operare in modo analogico e digitale da 2,7 a 3,6V, caratteristica che semplifica l'interfaccia grazie alla possibilità di condividere la stessa linea di alimentazione del microcontrollore. Microchip annuncia anche la MCP3911 Evaluation Board per MCU a 16-bit (ADM00398), un nuovo tool che consente lo sviluppo di progetti basati sul nuovo AFE. L' MCP3911 è disponibile in campionature e in volumi di produzione, in package SSOP e QFN a 20-pin.
www.microchip.com/get/XA2A



Kit di sviluppo per azionamenti automobilistici da Toshiba

Toshiba Electronics Europe ha annunciato una piattaforma di valutazione e sviluppo per sistemi embedded, destinati al controllo di motori automobilistici che richiedono la conformità allo standard ISO 26262 sulla sicurezza funzionale.



Il nuovo IAR KickStart Kit è progettato per velocizzare lo sviluppo, la prototipazione e il collaudo di sistemi basati sul microcontrollore automobilistico TMPM354 ARM Cortex™-M3. Si tratta di un chip che integra funzioni specifiche per il controllo del motore e tecniche di sicurezza funzionale, che è particolarmente adatto per realizzare sistemi di controllo del moto in veicoli elettrici e ibridi. Progettato per un utilizzo diretto e immediato, il kit di avviamento comprende una scheda di valutazione, una piattaforma di sviluppo integrata (IDE), una sonda di debug hardware JLink e tutta la documentazione necessaria. È anche fornito un esempio di codice applicativo come ausilio allo sviluppo. Il microcontrollore TMPM354F10TFG comprende al suo interno un azionamento programmabile avanzato, un blocco di elaborazione dedicato al calcolo vettoriale, per gestire il controllo ad orientamento di campo (FOC) o il controllo vettoriale (VC) del motore, riducendo il carico sulla CPU di oltre il 50%, un controllore CAN e tre canali CAN. Inoltre, Toshiba integra un convertitore resolver-digitale che permette ai segnali in uscita dal resolver o dai sensori di posizione del rotore, di pilotare direttamente il microcontrollore.

www.toshiba-components.com

Soluzione miniaturizzata ed economica da Intersil per i Pico videoproiettori a LED



Sviluppato per realizzare pico videoproiettori embedded o applicazioni accessorie, economiche e a basso consumo, il Pico-qHD di Intersil include un motore ottico ad alto volume di produzione, hardware, firmware e file di progetto personalizzabili. La soluzione utilizza il motore ottico compatto da 5,5 cm3 qHD E330 di Micron, che permette una integrazione semplice in un'ampia gamma di applicazioni,

Il Pico-qHD è montato su una scheda economica da 1,7" x 2" ed è stato progettato per rispondere alle necessità di bassi costi di produzione delle applicazioni consumer come fotocamere digitali, smartphone, tablet, videoriproduttori e altri prodotti palmari. Grazie al crescere del mercato di tali applicazioni, la società di ricerca Pacific Media Associates prevede per i pico videoproiettori vendite per 58 milioni di unità all'anno entro il 2015.

La piattaforma è incentrata sul nuovo integrato Intersil ISL97901, un driver LED buck/boost RGGB integrato e specializzato per applicazioni pico videoproiettori. La risoluzione di 2,0mA permette una regolazione ottimale del livello del bianco.

www.intersil.com/Pico-projector/PicoqHd.asp

National Instruments rilascia nuovi dispositivi I/O per NI CompactRIO



Il nuovo gateway wireless e i due nuovi nodi di misura che estendono le soluzioni offerte dal wireless sensor network (WSN) di National Instruments, sono realizzati per connettersi direttamente allo chassis di ogni famiglia di dispositivi CompactRIO.

Il Gateway della serie NI 9795 C è in grado di concentrare e aggregare i dati raccolti dai nodi WSN distribuiti, rendendoli disponibili alle applicazioni LabVIEW Real-Time funzionanti su CompactRIO.

I due nuovi nodi seriali NI WSN-3230 (1 porta RS232) e NI WSN-3231 (1 porta RS485) presentano ciascuno una porta seriale e due canali di I/O in ingresso per input discreti, rilevamento di eventi e controllo programmatico. Utilizzando il modulo WSN di LabVIEW, permettono di integrare algoritmi di interrogazione e analisi, direttamente nei nodi WSN, per costruire interfacce autonome verso una vasta gamma di strumenti, sensori e schede di controllo dotate di interfaccia seriale.

www.ni.com/wsn

HealthGO, telemedicina su piattaforma Health Hub di Freescale

HealthGO è una piattaforma tecnologica per il controllo remoto di pazienti, progettata per le socierà di monitoraggio pazienti che sviluppano dispositivi medici e soluzioni innovative, personalizzate ed affidabili, in tempi ridotti. Le soluzioni di teleassistenza medica permettono di migliorare la qualità di vita del paziente e la qualità della assistenza per mezzo di un accurato monitoraggio delle condizioni del paziente stesso, direttamente presso la sua abitazione.

Le esigenze di telesassistenza medica sono in crescita e in accelerazione in seguito all'aumento dell'aspettativa di vita e della pressione economica sui sistemi sanitari. Alla fine del 2011 solo 2,2 milioni di pazienti nel mondo godevano di teleassistenza con connettività integrata, contro 220 milioni di persone in EU e US che soffrono di malattie per le quali sarebbe utilizzabile con profitto la teleassistenza. Si ipotizza che il numero dei sistemi di teleassistenza crescerà nei prossimi 5 anni al tasso del 34,6% all'anno.

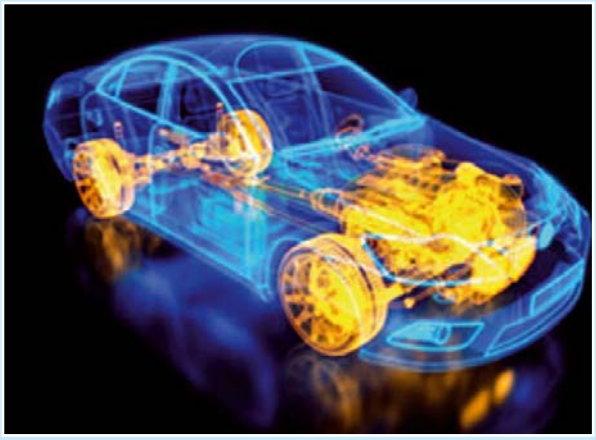
I monitor di HealthGO sono completi ma personalizzabili; sono disponibili con il monitoraggio dei parametri vitali già integrato, interfacce USB e Bluetooth, monitor della pressione sanguigna integrato (opzionale in HealthGO+), touch-screen, microfono e altoparlante integrato, funzionalità di gestione della malattia, capacità di comunicare informazioni, messaggi, informazioni meteorologiche e tutorial di informazione sanitaria. Sia l'hardware che il software sono personalizzabili dagli utilizzatori. Sono disponibili le API per lo sviluppo di applicazioni personalizzate e di interfacce grafiche; anche la configurazione del contenitore può essere modificato per adattarsi alle esigenze dei clienti.



HealthGo è una delle componenti della più ampia offerta di assistenza remota che comprende la connettività cellulare a copertura completa ed il portale Services & Logistics.

www.edevice.com
www.freescale.com

NI Automotive Forum 2012



Secondo le stime, la presenza dell'elettronica a bordo delle automobili crescerà esponenzialmente entro il 2015. La crescente complessità costruttiva dei veicoli che ne deriverà, implicherà una maggiore integrazione dei sistemi elettronici di bordo e una profonda conoscenza delle normative internazionali che mirano ad aumentare il grado di sicurezza delle autovetture.

L'ottava edizione di NI Automotive Forum 2012 si terrà il 6 giugno presso il centro congressi del Museo Nazionale dell'Automobile di Torino e rappresenta l'occasione ideale per confrontarsi su temi importanti come la riduzione del rischio, la conformità alle normative e agli standard di Functional Safety, la validazione e le certificazioni di conformità, i sistemi Safety Critical e di test real-time. Tutte le sessioni previste si articoleranno tra casi applicativi reali e presentazioni tecniche relative alla semplificazione dei sistemi.

Durante la tavola rotonda di fine mattinata, illustri esponenti e decision-maker del settore apriranno una finestra sulle regole da creare per strutturare la complessità dell'auto del futuro. Per tutta la durata dell'evento, i relatori si confronteranno con il pubblico sui risvolti delle tecnologie odierne sul mercato, come gli sviluppi futuri dell'elettrificazione del veicolo e le moderne sfide di validazione per ridurre consumi ed emissioni.

Novità dell'edizione di quest'anno, sono le due sessioni pratiche sulle tecniche di test real-time con NI VeriStand, che si svolgeranno in parallelo alla sessione pomeridiana in sala plenaria; in esse i partecipanti avranno a disposizione una postazione dedicata con hardware e software, che potranno prenotare anticipatamente inviando una e-mail a italy.marketing@ni.com e indicando la sessione preferita.

Durante la giornata di NI Automotive Forum, sarà premiata la "migliore applicazione per il settore Automotive": un comitato tecnico sceglierà e premierà la migliore applicazione del 2012 tra quelle pervenute.

Il Museo Nazionale dell'Automobile di Torino (*www.museoauto.it/website*), nato nel 1932, vanta una delle collezioni più rare ed interessanti nel suo genere, composta da quasi 200 automobili originali, dalla metà dell'800 ai giorni nostri, di oltre ottanta diverse marche.

Automotive Forum sarà anche un'occasione, per i partecipanti, di fare una visita guidata del museo.

ni.com/italy/automotiveforum

EMBEDDED IC & AUTOMATION

Il 21 giugno, presso il Centro Congressi Milanofiori di Assago (MI) vi aspetta una giornata di approfondimenti, dibattiti e incontri tra professionisti, per dare uno sguardo insieme al futuro; un doppio appuntamento, in realtà, che riguarderà sia il settore dell'elettronica embedded, sia le tecnologie wireless. Il termine embedded identifica genericamente un sistema elettronico a microprocessore progettato espressamente per un'applicazione o funzione specifica e identifica una vasta serie di sistemi per complessità (funzionalità) e dimensioni, in relazione al tipo di microprocessore, al sistema operativo, ed alla complessità del software. Al forum si focalizzerà l'attenzione sulle architetture più importanti e sul confronto tra x86 ed ARM; si parlerà inoltre di Android, Windows Embedded 8 e di casi applicativi.

L'evento è una mostra convegno con conferenze, dibattiti e workshop pianificati da Assodel con esperti prescelti per fornire lo stato dell'arte; durante la giornata ci sarà spazio per:

- Community - area demo per l'incontro tra domanda e offerta di fornitori e utilizzatori;
- Conference - un filo conduttore di interventi selezionati e dedicati alle novità di prodotto, alle applicazioni tecnico/commerciali, ai case history;
- Tutorial - seminari formativi e tutorial applicativo-progettuali per tecnici;
- Arena - Incontri "aperti" su confronti snelli e incisivi relativi all'utilizzo dei LED, con domande/risposte per ogni tipo di utente, coordinati da product manager.

Invece, per la sessione RF&wireless, il focus sarà sulle soluzioni per energy harvesting, Smart Grid e benchmarking delle tecnologie wireless più evolute.

L'evento è riservato agli specialisti del settore, come tecnici, progettisti, acquisitori, system integrator, software house; per parteciparvi occorre registrarsi on-line su *www.fortronic.it/page.asp?pag=1282&eveid=269&sec=REG#eve.*

www.fortronic.it

Festival dell'energia 2012

Torna, dal 15 al 17 giugno, il Festival dell'Energia, che si propone ancora una volta di affrontare le tematiche energetiche a 360° attraverso il dibattito e l'intrattenimento. Per la sua quinta edizione, il Festival si trasferisce a Perugia, proponendo al pubblico oltre 40 appuntamenti tra talk-show, convegni, presentazioni di libri, mostre, spettacoli e iniziative specifiche per il pubblico più giovane.
Il filo conduttore di questa edizione sarà "Energia per Ripartire", perché l'attuale congiuntura economica esige scelte complesse ma urgenti sul fronte della politica energetica, da cui dipendono competitività e crescita; perché la "smart city" non è più soltanto una provocazione su cui speculare, ma pone amministrazioni, ricercatori e imprese davanti a sfide concrete, destinate a cambiare il volto delle nostre città. E inoltre, perché il necessario riequilibrio tra fabbisogni e risorse del pianeta deve passare anche da un utilizzo più sostenibile ed equo dell'energia.

La tre giorni del Festival dell'Energia sarà preceduta, il 14 giugno, dal workshop Climate Summit, promosso dal Centro Interuniversitario di Ricerca sull'inquinamento da Agenti Fisici: una giornata di approfondimento sul tema dei cambiamenti climatici, che vedrà la partecipazione di docenti e ricercatori da tutto il mondo. Di particolare rilevanza, il controverso tema del futuro delle rinnovabili in Italia e un focus specifico su biomasse e agroenergie.
Tutte le novità, i protagonisti, le informazioni utili, gli appuntamenti chiave del Festival dell'Energia 2012, oltre allo storico delle passate edizioni, sono disponibili sul sito www.festivaldellenergia.it ma anche Facebook, Twitter, Flickr e You Tube. Il Festival dell'Energia 2012 si svolge con il patrocinio della Commissione Europea, della Presidenza del Consiglio dei Ministri, del Ministero dell'Ambiente e del Ministero dello Sviluppo Economico.
www.festivaldellenergia.it



2012
L'ENERGIA SPIEGATA
FESTIVAL DELL'ENERGIA
PERUGIA 15-17.06.2012

Altera DSP Seminar

Le soluzioni DSP di Altera, esposte e spiegate durante un tour europeo di seminari organizzati dal 12 al 21 giugno. Durante ciascun evento, gli esperti di Altera forniranno informazioni su applicazioni specifiche Digital Signal Processing (DSP) e metodologie di progettazione mirate per FPGA Altera.
I partecipanti potranno così scoprire come il modello basato sul flusso di progettazione nel DSP Builder, insieme con gli strumenti di MathWorks, aiuta i progettisti ad implementare applicazioni ad alte prestazioni in virgola mobile con i DSP.
Il team di Altera illustrerà il DSP Builder e alcune soluzioni ottimizzate basate su DSP per i settori militare (sviluppo end-to-end di soluzioni per Pulse-Doppler radar, strumenti di MathWorks da Altera per la simulazione) industriale (progettazione di controller per motori con trasformata di Park Clarke e integrale PI, implementazione fissa e in virgola mobile all'interno di un FPGA Altera, uso delle FPGA Altera per migliorare l'efficienza dei motori) video (nozioni di base di elaborazione video e utilizzo di Altera quadro video per costruire intere catene di elaborazione video) wireless (tecniche di mitigazione non lineare, amplificatori ad alta potenza utilizzati nei sistemi senza fili attraverso la riduzione del fattore di cresta -CFR- e la pre-distorsione digitale DPD, simulazione, misurazione e implementazione di algoritmi CFR in FPGA Altera).
Gli appuntamenti con i seminari di Altera si terranno presso le sedi locali della casa, in queste date:

- **12 giugno a Stoccarda (Germania);**
- **14 giugno ad Hannover (Germania);**
- **19 giugno ad Oxford (Inghilterra);**
- **21 giugno a Parigi (Francia).**

Per maggiori informazioni e per registrarsi, basta visitare la pagina *web www.altera.com/education/events/europe/evt-dsp-seminar-series.html.*
Tutti i seminari sono gratuiti ma bisogna registrarsi preventivamente.
www.altera.com

ALTERA
DSP Seminar Series

Sorveglia la tua attività con i nostri DVR

VIDEOREGISTRATORI DIGITALI



SET COMPLETO

€ 499,00

cod. FR368

SET completo di videosorveglianza con monitor TFT 7"

Rendi più sicuro il tuo negozio o il tuo ufficio con questo impianto potente ed affidabile. Si installa in pochi minuti! Possibilità di connessione ad un router ADSL per controllo remoto tramite Internet. Più di tre mesi di registrazione continua sull'hard disk interno.

Il sistema comprende:

- 1 pz DVR 4 canali H.264 con telecomando
- 1 pz Hard Disk 320 GB SATA
- 1 pz Monitor TFT 7"
- 2 pz Telecamera a colori da esterno con IR
- 2 pz Adattatore BNC maschio / RCA femmina
- 1 pz Adattatore BNC maschio / RCA maschio
- 2 pz Cavo video con plug d'alimentazione 10m
- 1 pz Alimentatore da rete 12V / 1,2A per monitor

DVR 4 CANALI H.264 con telecomando IR

- Interfaccia Ethernet 10/100 Base-T
- Interfaccia USB per salvataggio su chiavetta
- Porta USB per collegamento mouse
- Hard Disk SATA (non incluso)
- Compressione H264
- Frame rate 100 IPS
- Uscita VGA e Videocomposito
- 1 IN e 1 OUT audio
- 4 IN e 1 OUT allarme
- Controllo PTZ (RS485)

New

€ 199,00

cod. DVR4L6



FUTURA ELETTRONICA

Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Prezzi IVA inclusa.

Maggiori informazioni su questi prodotti e su tutte le altre apparecchiature sono disponibili sul sito *www.futurashop.it* tramite il quale è anche possibile effettuare acquisti on-line.

LETTORE RFID CON SD CARD E BUS I²C



dell'ing. LUCA PASCARELLA

Realizziamo un lettore di tessere RFID per controllo accessi dotato di interfaccia utente a display, una SD Card per memorizzare le liste degli utenti abilitati e indesiderati e di un protocollo I²C per comunicare con periferiche digitali. Prima puntata.

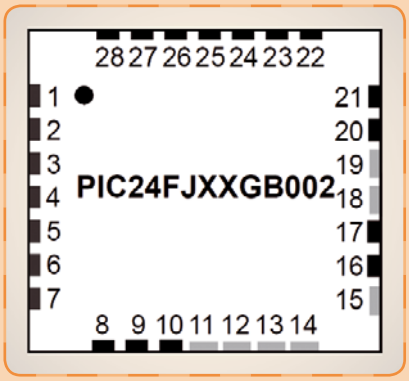
Il progetto presentato in questo numero nasce dall'idea di creare un sistema di abilitazione a chiave elettronica per il controllo di accessi ad ambienti e per l'abilitazione all'uso di veicoli e macchinari. Il sistema, così ideato, sfrutta la tecnologia RFID.

In generale è accezione comune associare il concetto di protezione al termine chiave; dove per chiave intendiamo quell'oggetto metallico dalle più svariate forme e combinazioni. L'obiettivo di questa articolo, oltre a presentarvi un progetto pratico ed innovativo, è anche quello di sorpassare questo preconcetto, illustrandovi come sia possibile, con un piccolo circuito elettronico, creare un meccanismo alternativo al solito blocco meccanico. La conseguenza immediata di questa combinazione di componenti elettronici crea il vantaggio di portarsi dietro non più quelle pesanti ed ingombranti chiavi metalliche, odiate dai metaldetector, bensì una comodissima tessera plastica dotata di chip con codice univoco attivabile anche ad alcuni centimetri di distanza dal lettore.

Le chiavi elettroniche in realtà sono sempre più presenti nella nostra vita quotidiana, ad esempio,

Caratteristiche del **PIC24FJ64GB002**

Il microcontrollore scelto in questo progetto è un PIC della Microchip famiglia 24F. Questa grande famiglia di dispositivi a 16 bit racchiude un insieme corposo di dispositivi dalle caratteristiche più disparate.
Ogni unità è in grado di lavorare su word da 16 bit ad una velocità massima di 16 MIPS con clock da 32 MHz, queste discrete prestazioni consentono di soddisfare molte esigenze e di ospitare, nei modelli più completi, come nel caso scelto, anche una periferica USB On The Go. Inoltre molti dispositivi della famiglia 24F appartengono all'architettura XLP (*Extreme Low Power*) disegnata per avere bassissimi consumi energetici in qualsiasi condizione di funzionamento ed in particolar modo in modalità *sleep*. Nella tabella sono descritte alcune delle caratteristiche più importanti del PIC24FJ64GB002.



Caratteristiche tecniche principali del PIC24FJ64GB002.

Nomi parametri	Valore
Architettura	16 bit
Velocità CPU (MIPS)	16
Tipo di memoria	Flash
Quantità di spazio in Flash (KB)	32
Quantità di spazio in RAM (Bytes)	8,192
Range di Temperatura (°C)	da -40 a 125
Intervallo di alimentazione (V)	2 to 3.6
I/O Pins	21
Numero di Pin	28
Caratteristiche del sistema	BOR, LVD
POR	Yes
WDT	Yes
Oscillatori interni	8 MHz, 32 kHz
Costruzione in nanoWatt	Low Sleep/Fast Wake/Fast Control
Periferiche di comunicazione digitali	2-UART, 2-SPI, 2-I2C
Periferiche analogiche	1-A/D 10x10-bit @ 500(ksps)
Comparatori	3
Capture/Compare/PWM Peripherals	5/5
Risoluzione del PWM (bits)	16
Timers	5 x 16-bit
Porta Parallela	PMP
Hardware RTCC	Yes
USB	On-The-Go
XLP	Yes
Canali Touch Capacitivi	10

quando si acquista un'automobile, la preoccupazione che affligge indistintamente ogni conducente è la paura per il furto. La possibilità di poter parcheggiare l'auto in un luogo qualunque e ritrovarla integra, al proprio ritorno, non è poi così scontata. Per questo motivo ogni casa costruttrice fornisce di serie o tramite optional, un insieme di funzionalità meccaniche od elettro-meccaniche volte a scongiurare le possibilità di atti criminali.
Le strategie di difesa, in questi casi, sono tante, si va dal semplice ma efficace blocco meccanico degli sportelli e dello sterzo, all'immobilizzazione elettrica del veicolo, ai sistemi di richiamo dell'attenzione dei passanti con suoni evidenti e talvolta molesti, oppure, in caso di estrema protezione si possono scegliere meccanismi basati sulla combinazione di due tecnologie ben affermate, il GPS ed il GPRS (sistemi conosciuti anche come antifurti satellitari).
Tradizionalmente, però, l'oggetto comune a tutte le autovetture che allo stesso tempo consente di diversificare un veicolo dall'altro sono le chiavi. Le chiavi servono a creare particolari geometrie variabili le cui sagome, realizzate tramite opportune sporgenze e/o rientranze, permettono di creare un unico profilo (combinazione) valido a far ruotare il chiavistello; con il fine di abilitare solo il reale possessore del bene all'apertura delle portiere o alla messa in moto nel caso degli autoveicoli.
Nel corso del tempo il concetto di chiave meccanica ha subito una notevole evoluzione. Un primo aggiornamento, legato soprattutto al mondo degli autoveicoli, ha portato alla fusione in un unico oggetto di un trasmettitore RF e della reale chiave di avviamento. Quest'insieme di due tecnologie (meccanica ed elettrica) ha semplificato per molti anni la vita dei conducenti, i quali hanno potuto sbloccare le porte dell'automobile al sol tocco di un pulsante, evitando particolari acrobazie nel caso in cui le braccia e quindi le mani fossero state impegnate in altri compiti.
Dal punto di vista della sicurezza, l'evoluzione si è avuta con l'aggiunta di un ulteriore sistema di codifica inserito all'interno della chiave tradizionale. Con questo nuovo dispositivo (chip RFID) l'automobile può essere avviata solo quando la chiave ha l'esatta sagoma attesa dal chiavistello e quando il codice numerico, contenuto nel chip elettronico,

coincide con quello presente nella centralina dell'automobile. Questo meccanismo a doppia chiusura ha l'obbiettivo di scoraggiare i tentativi di clonazione (data la maggiore complessità dell'azione furtiva) riducendo di conseguenza i casi di vandalismo o di furto del bene mobile.
L'evoluzione in termini di sicurezza e di praticità d'uso dell'autoveicolo hanno sempre spinto i ricercatori alla creazione di nuove soluzioni, a volte ottenendo risultati davvero bizzarri. Ad esempio, nei modelli di autovetture più ambiti vengono impiegati sofisticati sistemi elettronici che essendo impopolari sulle utilitarie sono considerati esoterici, è questo il caso dell'uso delle impronte digitali o dei comandi vocali come chiave d'accesso.
Nel progetto di seguito presentato l'idea di base consiste nel creare un meccanismo di protezione elettronico (**Fig. 1**) dotato di microcontrollore al quale è stato affiancato un chip lettore di tessere RFID. Il circuito, non è una novità assoluta del settore ma rispetto ad altri progetti simili differisce per l'assenza di un relè o di un Mosfet direttamente controllabile; questa particolarità consente di diminuire i casi di manomissione, infatti, il consenso proveniente dalla chiave elettronica verso l'applicazione di utilizzo avviene solo tramite opportuni segnali digitali difficilmente clonabili con meccanismi rudimentali.
Infine per rendere il circuito universale e personalizzabile in base

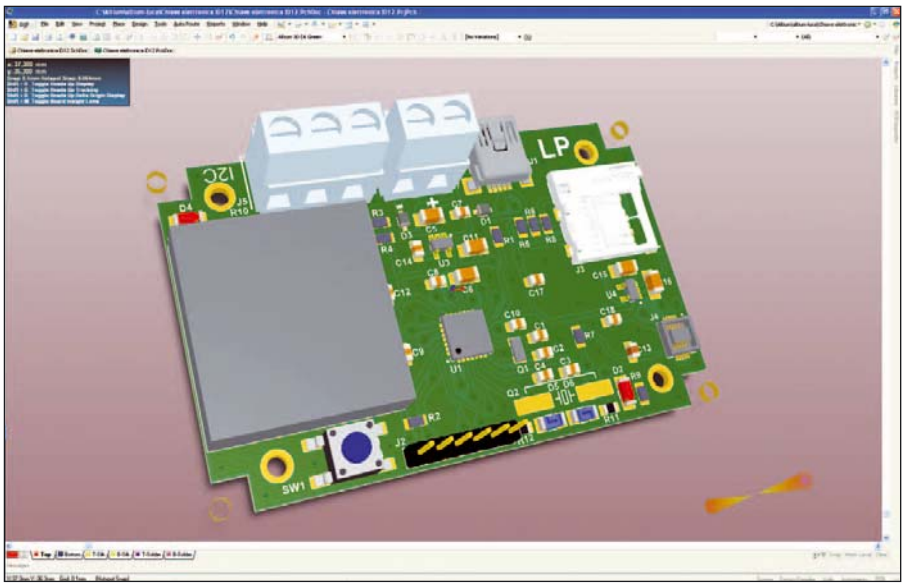
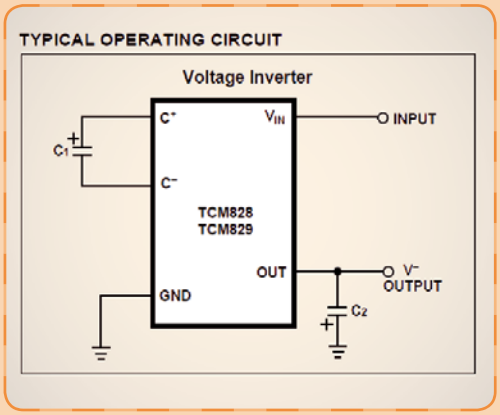
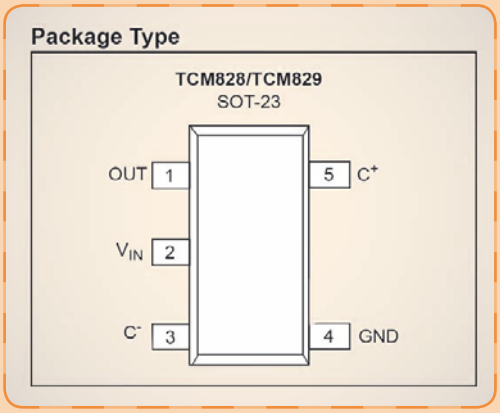


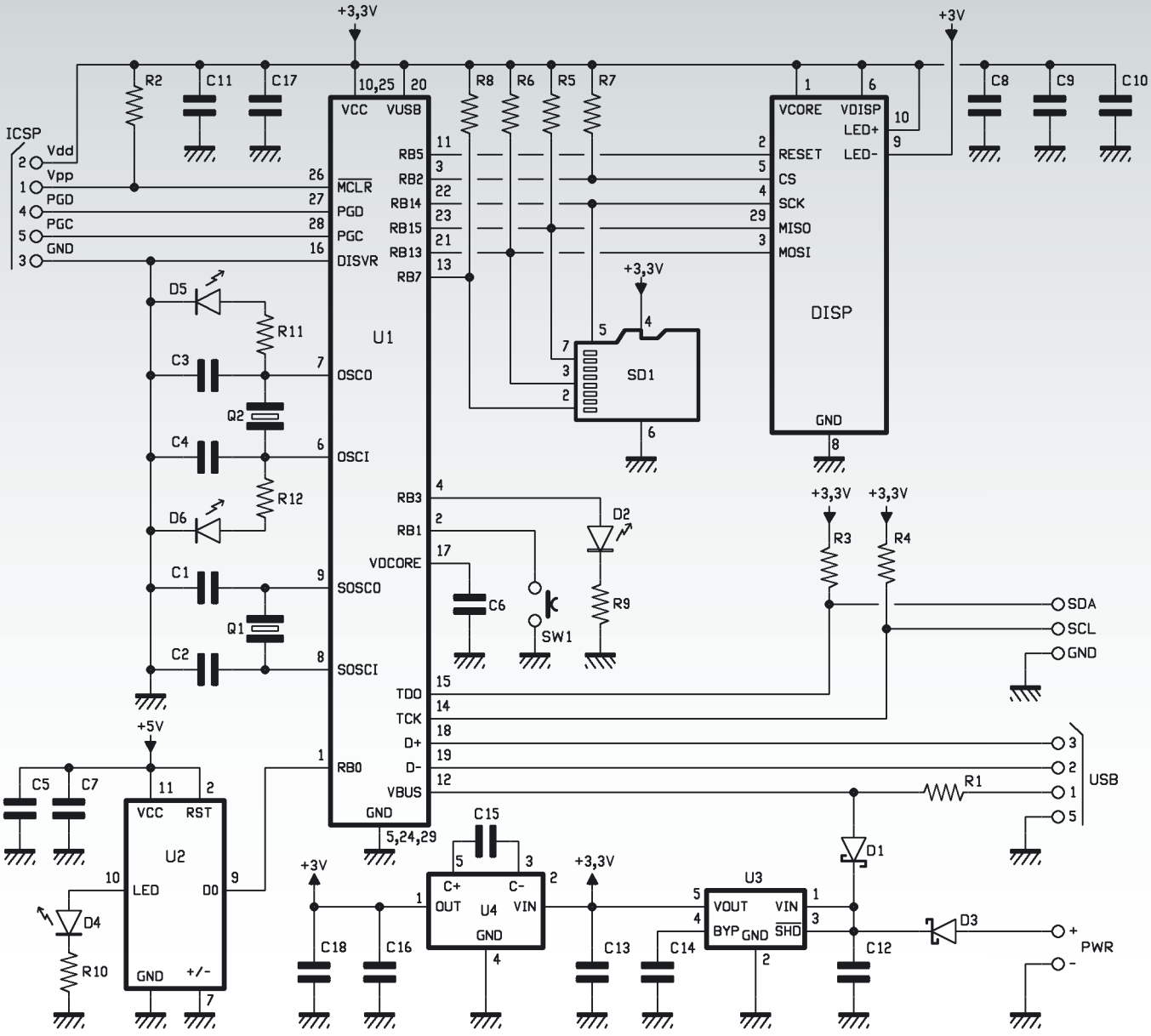
***Fig. 1** - Anteprima del circuito realizzato ottenuta durante la fase di progettazione del PCB.*

Caratteristiche del **TCM828/829**

I circuiti integrati TCM828/829 sono dei Charge pump con efficienza del 95%, consentono di ottenere una corrente in uscita dell'ordine delle decine di milliampere. I due modelli di IC possono erogare una tensione negativa oppure una doppia rispetto a quella d'ingresso. La tensione d'alimentazione può variare nel range 1,5V – 5V.
L'impiego di questo tipo di convertitori DC-DC è particolarmente adatto per i circuiti in cui c'è bisogno di una grande linearità e stabilità della tensione d'uscita, legata ad un minimo ingombro possibile. Nella configurazione minima i due convertitori DC-DC necessitano di solo due condensatori da 3,3uF. L'aspetto negativo è la bassa corrente d'uscita che si aggira intorno al valore massimo di 25 mA per questi due componenti. Nel caso in cui si richiede una corrente superiore si può preferire l'impiego di un regolatore switching accettando il compromesso di ospitare a bordo del circuito stampato anche un induttore ed una minore precisione in termini di linearità del segnale d'uscita (affetto da maggiore rumore).
In figura è riportato un tipico schema di montaggio dei componenti TCM828/829.







alle esigenze di progetti analoghi abbiamo previsto a bordo dello stampato anche un display LCD colorato sul quale possono essere visualizzati svariati messaggi informativi utili per interagire con l'utente ed aggiornarlo in tempo reale sullo stato del sistema.

SCHEMA ELETTRICO

Apriamo questo nuovo paragrafo con la descrizione dello schema elettrico per poi proseguire con la spiegazione del principio di funzionamento e del software appositamente sviluppato.

Il cuore di tutto il sistema (individuato, nello schema elettrico, dal simbolo U1) è ancora una volta un microcontrollore della casa costruttrice di semiconduttori Microchip, in particolare, l'unità di calcolo selezionata è il modello *PIC24FJ64GB002*.

Questo chip appartenente alla famiglia dei 24F è un nuovo microcontrollore non ancora presentato su questa rivista. Il chip individuato è capace di eseguire il firmware, residente all'interno della propria memoria FLASH, ad una velocità di 16 MIPS, con una frequenza di clock auto-generata di 32 MHz (l'IC ha un generatore di clock interno da 8 MHz, frequenza incrementabile dall'unità PLL fino a 32 MHz). Questo microcontrollore è stato impiegato per l'elevata quantità di periferiche hardware di cui esso è dotato, in particolar modo è stato selezionato per la periferica USB On-The-Go, funzionante, tramite opportune librerie, sia come periferica Host che come periferica Client. Il chip principale (microcontrollore) denominato U1 nello schema elettrico viene



Credi di passare inosservato?

Attento all'occhio elettronico che ti trova in un secondo tra 36 milioni di facce

Nelle serie TV e nei film si vedono spesso software in grado di trovare un volto tra le migliaia che passano in un luogo affollato.
Fantascienza? Non più!
La Hitachi Kokusai Electric, una società giapponese, ha sviluppato un nuovo rivoluzionario sistema di sorveglianza capace di scandagliare 36 milioni di volti al secondo ed individuare al volo persone scomparse o ricercate. È in grado di studiare praticamente qualsiasi immagine – sia presa da uno smart phone, che da telecamera.
Il software utilizza un sistema algoritmico per effettuare un match tra i volti rilevati e un database di immagini delle persone da cercare.
È in grado di operare su qualsiasi volto purché abbia una dimensione minima di 40 pixel per 40 pixel e un'inclinazione del volto non superiore ai 30°. È perfetto quindi per essere utilizzato in posti affollati con continuo passaggio come un aeroporto, una stazione o un centro commerciale. Arriverà sul mercato il prossimo anno.
Vi sembra un Grande Fratello discutibile? Be' sicuramente lo è ma potrà rivelarsi anche estremamente utile, non solo per la cattura di ricercati ma anche per trovare persone scomparse contro la propria volontà, bambini persi o rapiti, vecchietti con Alzheimer che non si rendono conto di dove sono, persone che hanno perso la memoria.

fabbricato con diversi footprint per adattarlo a tutte le esigenze di mercato, il modello da noi impiegato è il QFN (*Quad Flat No-leads*) da 28 pin (7 pin per ognuno dei 4 lati, con l'aggiunta di un ulteriore pad centrale denominata *Thermal pad*).
Il PIC24FJ64GB002, inoltre, appartiene alla categoria XLP (*eXtreme Low Power*) di casa Microchip, caratteristica essenziale per le applicazioni a batteria dove è fondamentale assorbire pochissima energia elettrica in tutte le condizioni di funzionamento ed in particolar modo nella modalità Sleep.
Come già anticipato in precedenza, il PIC24FJ64GB002, dispone di una sorgente di clock interna da 8 MHz con 0.25% di accuratezza, caratteristica che consente di risparmiare sulla BOM (*Bill Of Material*), favorendo così una diminuzione del costo di produzione ed un'integrazione più spinta, conseguenza della minore complessità circuitale.
Passiamo ora alla descrizione degli altri componenti che popolano il circuito stampato. Proseguendo per ordine incontriamo U2 (*ID-12*), questo chip è probabilmente già noto a molte persone, esso è un lettore di tessere RFID con antenna integrata. U2 è un componente molto versatile configurabile in tre modalità distinte (*ASCII, Magnet Emulation e Wiegand26*) tramite un unico PIN denominato *Format Selector*. Nel nostro caso è stata scelta la modalità *ASCII*, questa modalità è la più semplice di tutte dato che fornisce in uscita sul PIN 9, denominato *TTL Data*, l'*ID* sotto forma di caratteri *ASCII* contenuto all'interno della tessera RFID accostata all'antenna integrata del lettore. U2 richiede una tensione di alimentazione positiva di 5 V. Per ottenere questo valore il circuito deve essere alimentato con una tensione stabilizzata dello stesso valore (5 V) applicabili indistintamente al morsetto d'ingresso a due poli (in sigla J7) o alla porta *mini USB* (in sigla J1); il circuito è autonomo e privo della necessità di escludere manualmente una delle due alimentazioni. In realtà, le tensioni provenienti dalle due possibili sorgenti a 5 V attraversano rispettivamente i diodi D1 o D3 per poi raggiungere U2; per tal motivo subiscono una caduta di tensione compresa tra 0,4 V e 0,5 V. Questa attenuazione è dovuta alla natura intrinseca dei diodi capaci di generare ai loro estremi una funzione non lineare nel piano tensione corrente; tale da approssimare nel migliore dei modi un diodo ideale (nel nostro caso una banale funzione di non ritorno). I diodi vanno scelti opportunamente in base a molte considerazioni, in questo caso abbiamo tenuto presente la corrente massima diretta e la caduta di tensione diretta, scegliendo un *Diodo Schottky* che soddisfa pienamente i nostri requisiti. Dalle



***Fig. 2** - Display LCD 130 x 130 pixel colorato con profondità di colore di 12 bit, 4 bit per colore.*

scelte tecniche precedenti facendo qualche semplice conto otteniamo che se la tensione d'ingresso proveniente dal morsetto J7 o dal connettore *mini USB* è di almeno 5V e l'attenuazione massima a valle del *diodo Schottky* è di 0,5 V, la tensione d'alimentazione di U2 sarà di 4,5 V. Il chip U2 da manuale accetta un intervallo di alimentazione molto ampio ed infatti con 0,5 V in meno rispetto al valore nominale è ancora capace di funzionare correttamente e leggere gli *ID* delle tessere RFID accostate a qualche centimetro di distanza; inoltre dalle prove pratiche è emerso che il chip continua a funzionare correttamente anche con un paio di decimi di volt in meno al valore minimo dichiarato.
A questo punto sorge spontaneo chiedersi come risolvere il problema di compatibilità elettrica dei livelli TTL generati da U2 e ricevuti da U1, fortunatamente il PIC24FJ64GB002 ha la soluzione in tasca, infatti, tutte le porte digitali a 3,3V sono anche compatibili, in ingresso, ai 5 V positivi, garantendo così una soluzione pulita ed efficace.

Proseguendo con la descrizione dello schema elettrico incontriamo altri due chip, in particolare U3 ed U4, questi due componenti rispettivamente consentono di diminuire il livello della tensione d'ingresso (+5 V) stabilizzandola a 3,3 V (necessari per alimentare il PIC, l'LCD e la micro SD) e di generare una tensione negativa di -3 V necessaria ad accendere le lampade della retroilluminazione del display LCD impiegato (il cui connettore è individuato nello schema dalla sigla J3). Il display LCD selezionato, montato sul connettore J3, necessita di una tensione di + 3,3 V per alimentare le logiche interne e di una tensione di circa 6 V per alimentare i diodi LED bianchi della retroilluminazione; questa tensione non essendo necessariamente riferita alla massa comune del circuito stampato può essere ottenuta applicando sul morsetto positivo la tensione di alimentazione del display di 3,3 V e sul morsetto negativo la tensione negativa ottenuta da U4.
La differenza di potenziale che si ottiene tra i due morsetti è di circa 6 V, esattamente il valore richiesto dalle lampade per ottenere un bianco brillante.
Continuando a parlare del display LCD (**Fig. 2**), possiamo dire che questo non è nuovo alla rivista, infatti è già comparso in altri progetti tipo "*Il voip senza PC*" sul numero 148. Questo display è dotato di un'interfaccia SPI ridotta (ovvero è privo del segnale *MISO Master Input Slave Output*) e di un PIN di reset. Tramite bus SPI è possibile impartire i comandi base per accendere, spegnere, cambiare l'intensità o il colore di un singolo pixel; per ottenere effetti grafici più sofisticati si possono usare le stesse librerie software che abbiamo impiegato in questo progetto, le quali forniscono delle primitive grafiche per poter disegnare forme geometriche standard, testo ASCII o immagini bitmap. Questo display viene commercializzato in due modelli apparentemente identici, in realtà ciò che cambia è la logica di pilotaggio interna che può essere dell' EPSON (modello con chip *S1D15G00*) o della PHILIPS (modello con chip PCF8833). In base alla logica installata nel display, il set di istruzioni impiegato per l'inizializzazione ed il controllo in generale (ad esempio per il contrasto) differiscono per alcuni valori, rendendo i due modelli incompatibili.
Per facilitare l'impiego di quest'oggetto abbiamo previsto all'interno delle librerie la possibilità di abilitare l'uno o l'altro modello tramite l'impiego alternativo di una delle due seguenti definizioni: *#define PCF8833_PHILIPS oppure #define S1D15G00_EPSON.*
Infine per concludere la descrizione dello schema elettrico mancano da nominare i componenti discreti ad emissione di luce ed i componenti passivi. Il LED D2 è pilotato da una porta di I/O del PIC ed è usato come indicatore di attivazione della scheda, esso lampeggia regolarmente ad 1 Hz. Il LED D4 è controllato direttamente dal lettore RFID (attivato ogni qual volta il codice seriale di una tessera RFID è correttamente riconosciuto). I LED D5 e D6 sono impiegati per segnalare lo stato della connessione USB verso l'host (questi due indicatori possono essere installati alternativamente al quarzo Q2 da 8 MHz), il loro lampeggio è dettato dalla funzione *BlinkUSBStatus()* che farà alternare la loro luminosità proporzionalmente all'avanzamento dello stato della connessione USB. Il pulsante SW1, invece, è stato previsto per

Output Data Structure – ASCII

STX (02h)	DATA (10 ASCII)	CHECK SUM (2 ASCII)	CR	LF	ETX (03h)

[The 1byte (2 ASCII characters) Check sum is the "Exclusive OR" of the 5 hex bytes (10 ASCII) Data characters.]

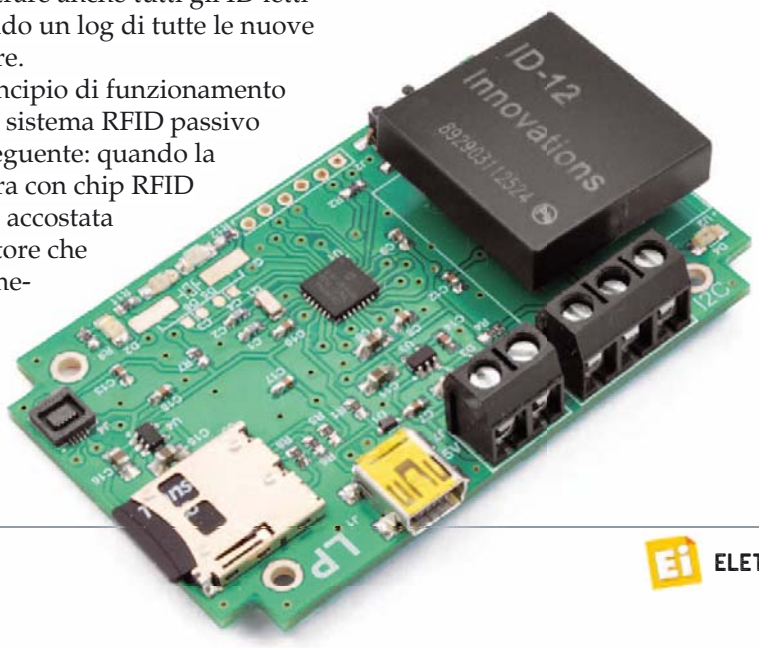
***Fig. 3** - Descrizione della stringa inviata dal modulo ID-12 nella modalità ASCII.*

consentire un'interazione in fase di debug. I restanti componenti non menzionati come i condensatori di bypass e le resistenze di pull-up servono rispettivamente, per filtrare i segnali digitali e fornire alimentazioni stabilizzate e prive di interferenze elettromagnetiche a tutte le logiche elettroniche ed ad inizializzare opportunamente tramite pull-up o pull-down i segnali presenti sulle interfacce di comunicazioni digitali come richiesto dalle opportune specifiche (interfaccia SPI per la micro SD ed interfaccia I2C per la comunicazione con le schede esterne).

PRINCIPIO DI FUNZIONAMENTO

Il principio di funzionamento di questo circuito è molto semplice e facile da descrivere. L'idea di base è quella di fornire un consenso qualora la scheda RFID, accostata all'antenna ricevitore presente all'interno dell'integrato U2, risulta registrata ed abilitata, quindi presente in una *white list*. Per raggiungere questo obiettivo il circuito è stato dotato di un connettore in grado di ospitare una memoria *micro SD* nella quale possono essere registratati gli ID delle tessere RFID accostate. La memoria SD oltre a fornire la lista di consenso permette di registrare anche tutti gli ID letti creando un log di tutte le nuove tessere.

Il principio di funzionamento di un sistema RFID passivo è il seguente: quando la tessera con chip RFID viene accostata al lettore che chiameremo anche master, la tessera viene colpita dalla radiofrequenza irradiata continuamente dall'antenna del dispositivo master. L'energia posseduta dalle radiazioni RF viene convertita in energia elettrica dall'antenna presente all'interno della tessera; quest'energia così catturata viene convogliata ed immagazzinata in un piccolo condensatore che a sua volta sarà la fonte di energia continua per alimentare la logica interna ed avviare la comunicazione con il lettore. Dato che la tessera o slave non ha sufficiente energia per trasmettere una propria onda elettromagnetica, il principio sul quale si basa la comunicazione consiste nell'assorbimento modulato dell'energia irradiata dall'RFID master. Così facendo il lettore fornisce nello stesso tempo sia l'energia per attivare il chip affogato nella tessera sia l'onda che, opportunamente modulata dal dispositivo slave funzionerà come canale di comunicazione bidirezionale. Dato che il quantitativo di informazioni scambiate sono molto limitate la transizione dura poche decine di micro secondi. Chiaramente il tempo di comunicazione è un fattore determinante dato che l'energia presente nel condensatore può essere davvero minima. Lo stesso principio di funzionamento, descritto a grandi linee, è adottato anche per il lettore di tessere RFID capaci di memorizzare i dati, l'unica differenza consiste in una più complessa logica presente nel chip RFID che sarà dotato anche di una memoria riscrivibile di tipo EEPROM o simili.



Tornando alla descrizione del principio di funzionamento del nostro lettore, dopo aver scoperto come U2 riconosce le tessere, resta da descrivere come il sistema sia capace di fornire consenso solo ad un gruppo ristretto di tessere.

Per questo motivo U2 è configurato in modo che quando una tessera viene accostata all'antenna, sul PIN 9 (porta seriale a 9600 bps) viene trasmesso l'ID sotto forma di caratteri ASCII presente all'interno della tessera stessa. La comunicazione inizia con un codice di apertura (*0x02*) e termina con un codice di chiusura (*0x03*), ad esempio se la tessera ha un ID ASCII del tipo *4400F598557C* i codici esadecimali inviati tramite la porta seriale corrispondono alla sequenza: *0x02 0x34 0x34 0x30 0x30 0x46 0x35 0x39 0x38 0x35 0x35 0x37 0x43 0x0D 0x0A 0x03.* Nella **Fig. 3** è possibile vedere la schematizzazione dei messaggi inviati dal chip ID12 al riconoscimento di una tessera RFID.

A questo punto dopo che U2 ha inviato la giusta combinazione di byte ed il microcontrollore PIC ha effettuato il parsing del codice estraendo solo i caratteri ASCII corrispondenti all'ID della tessera può procedere con la

***Tabella 1** - Descrizione dei comandi accettati dal firmware nella versione base.*

Comando	Direzione	Descrizione	Risultato
0x00	Lettura	Legge l'anno	Da 0 a 99. 2000 sottointeso
0x01	Lettura	Legge il mese	Da 1 a 12
0x02	Lettura	Legge il giorno	Da 1 a 31
0x03	Lettura	Legge il giorno della settimana	Da 0 a 6
0x04	Lettura	Legge le ore	Da 0 a 23
0x05	Lettura	Legge i minuti	Da 0 a 59
0x06	Lettura	Legge i secondi	Da 0 a 59
0x07	Lettura	Legge lo stato	0 (disabilitato) o 1 (abilitato)
0x00	Scrittura	Imposta l'anno	Da 0 a 99. 2000 sottointeso
0x01	Scrittura	Imposta il mese	Da 1 a 12
0x02	Scrittura	Imposta il giorno	Da 1 a 31
0x03	Scrittura	Imposta il giorno della settimana	Da 0 a 6
0x04	Scrittura	Imposta le ore	Da 0 a 23
0x05	Scrittura	Imposta i minuti	Da 0 a 59
0x06	Scrittura	Imposta i secondi	Da 0 a 59

verifica, che consiste nel confrontare il codice così ottenuto con i codici memorizzati all'interno delle liste presenti nella micro SD. Nel caso in cui il codice risulta assente dalla lista di consenso o di dissenso questa stringa verrà memorizzata al'interno di un altro file che conterrà traccia di tutte le tessere accostate al lettore.

Le liste presenti di default all'interno della micro SD sono tre e rispettivamente servono ad abilitare i dispositivi accettati, pertanto chiamata *WHITE.TXT*; a disabilitare i dispositivi rifiutati, denominata *BLACK.TXT* ed infine l'ultima lista denominata *NEW.TXT* contenente l'elenco degli ID non ancora registrati.

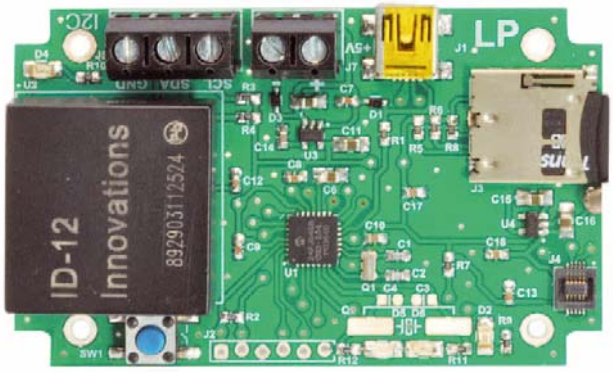
Il sistema, può diventare più complesso aggiungendo poche righe di codice, infatti sfruttando l'RTCC (*Real Time Clock Calendar*) interno al microcontrollore si può creare un file di testo, denominato ad esempio *LOG.TXT*, in cui memorizzare la tripletta di dati: ID del dispositivo, la data e l'ora ottenendo così una registrazione minuziosa di tutti i movimenti. Tutte queste informazioni possono essere consultate tramite due meccanismi: il primo consiste nell'estrarre la micro SD dal dispositivo ed inserirla in un apposito lettore di schede di memoria per PC; con la seconda modalità, invece, è possibile collegare la porta *mini USB* a bordo dello stampato ad un PC. In entrambi i casi ciò che viene riconosciuto dal PC è un *FileSystem* di tipo FAT (*File Allocation Table*) con all'interno 3 file denominati

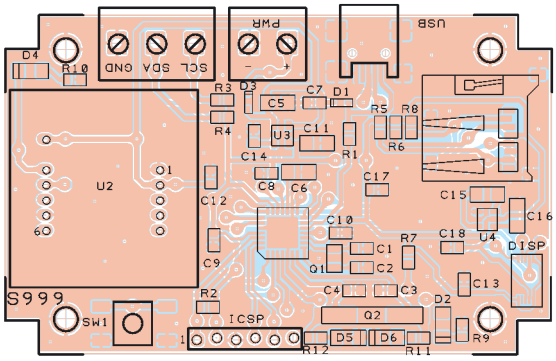


***Fig. 4** - La prima immagine da sinistra verso destra mostra lo stato del display dopo che il dispositivo ha terminato l'inizializzazione; proseguendo, la seconda immagine avvisa l'utente che il consenso è stato dato dopo aver accostato una scheda il cui ID è presente nella White list; infine l'ultima immagine mostra il consenso ad una azione di uscita da un ambiente, o di disattivazione in genere, dopo aver riaccostato la tessera corretta.*



***Fig. 5** - La prima immagine della serie mostra l'assenza dell'ID della tessera accostata nelle liste di consenso e di dissenso e l'aggiunta di questo nuovo valore nella lista NEW.TXT; la seconda immagine si ottiene quando la tessera è solo registrata e non abilitata o disabilitata; nella terza immagine l'ID della tessera appartiene alla White list quindi il dispositivo invertirà lo stato di consenso o di dissenso precedentemente memorizzato; nell'ultima immagine l'ID appartiene alla Black list quindi il dispositivo non compierà nessuna azione.*





Elenco Componenti:

R1, R3÷R8: 10 kohm (0805)
R2: 47 kohm (0805)
R9÷R12: 1 kohm (0805)
C1÷C4: 15 pF ceramico (0805)
C5, C6: 10 µF ceramico (1206)
C7÷C9: 100 nF ceramico (0805)
C10: 100 nF ceramico (0805)
C11: 10 µF ceramico (1206)
C12, C13: 100 nF ceramico (0805)
C14: 470 pF ceramico (0805)
C15, C16: 10 µF ceramico (1206)
C17, C18: 100 nF ceramico (0805)
D1: PMEG4010CEJ (SOD323)
D3: PMEG4010CEJ (SOD323)
D2: LED rosso (1206)
D4: LED verde (1206)
D5, D6: LED rosso (1206)
Q1: Quarzo 32 kHz
Q2: 8 MHz (XT49M)
U1: PIC24FJ64GB002-QFN28 (MF999)
U2: ID-12
U3: TC1015-3V3 (SOT23)
U4: TCM828/829 (SOT23)
USB: Mini USB
DISP: Display Nokia 128x128
SW1: Microswitch (PTS645)

Varie:
- Morsetto 2 poli passo 5,08 mm
- Morsetto 3 poli passo 5,08 mm
- Connettore SMD per Display
- Connettore microSD
- Circuito stampato

WHITE.TXT, BLACK.TXT e NEW.TXT.
Inoltre è anche possibile usare il set di istruzioni previste nella **Tabella 1** per conoscere lo stato del sistema.
Leggendo l'articolo sarà emerso che in realtà sulla scheda non è presente nessun relè per cui il consenso o il dissenso non abilita direttamente l'apertura o la chiusura di un circuito elettrico.
Per comunicare con il circuito della chiave elettronica, da parte di un sistema di controllo esterno, si utilizza il protocollo I2C, tramite il quale è possibile interrogare il master e determinare la validità o meno di una chiave avvicinata alla stesso. Un'ulteriore funzionalità è data ancora dalla porta *mini USB*, infatti collegando quest'ultima ad un PC, il computer, oltre a riconoscere un dispositivo di archiviazione di massa e visualizzare tutto il suo contenuto, rileverà, contemporaneamente, anche una porta di comunicazione seriale sulla quale verranno inviati i caratteri ASCII relativi alle tessere riconosciute dall'antenna.
L'ultima caratteristica del circuito è la presenza di un display LCD che può essere programmato in modo da comunicare lo stato del sistema all'utente. Nel nostro caso verranno visualizzati due messaggi: attivato e disattivato in una modalità bistabile senza memoria permanente. Ciò significa che partendo all'accensione il sistema è disattivato e neutro, accostando una tessera il cui codice è presente in *WHITE.TXT* lo stato del sistema passa ad attivo. Per disattivare lo stato (interrogabile anche tramite *I2C* con comando *0x07*) sarà necessario riaccostare una tessera il cui codice è presente in *WHITE.TXT*.
Per aggiungere un TAG ad una lista è sufficiente accostarlo una prima volta al lettore, il display nell'ultima riga mostrerà un colore di fondo giallo volto ad indicare l'aggiunta dell'ID letto in *NEW.TXT*, a questo punto si può accedere al contenuto della micro SD e copiare l'ultimo ID presente in *NEW.TXT* nel file *WHITE.TXT*.
Nella **Fig. 4** è possibile vedere schematizzati i diversi stati logici che mostrerà il display in funzione del consenso o del dissenso; invece nella **Fig. 5** si possono vedere i messaggi indicativi che mostrerà il display al seguito del riconoscimento degli ID delle tessere RFID accostate. Infine nella **Tabella 3** vengono descritti i colori che può assumere lo sfondo del display per segnalare visivamente lo stato del sistema.

REALIZZAZIONE PRATICA

Passiamo finalmente all'aspetto probabilmente più interessante

***Tabella 2** - Descrizione dei colori dello sfondo e dei caratteri del display in relazione alla presenza o all'assenza dell'ID ricevuto dal lettore RFID.*

Stato	Sfondo	Colore caratteri
ID del Tag riconosciuto nella lista	Verde	Nero
ID del Tag non riconosciuto nella lista	Rosso	Nero
Aggiunto l'ID nella lista	Giallo	Nero
Errore nell'apertura della lista	Nero	Rosso

da un punto di vista pratico, ovvero l'assemblaggio della scheda con i componenti necessari alla messa in funzione per poi proseguire, nella prossima puntata, con la spiegazione del funzionamento del firmware realizzato appositamente per soddisfare le esigenze presentate nel principio di funzionamento.
Prima di tutto occorre procurasi tutti i componenti necessari all'assemblaggio, quindi i circuiti integrati, le resistenze, i condensatori, il circuito stampato, i connettori e le diverse altre minuterie (led, pulsanti, quarzi, ecc.), al termine del reperimento potrete passare alla saldatura di ognuno di essi che potrà avvenire con discreta manualità in poche ore.
Come di norma, il consiglio è quello di procedere per ordine, saldando prima i componenti più complessi, in questo caso U1 e successivamente i componenti più semplici o voluminosi che una volta montati riducono la manovrabilità della punta del saldatore.
Per ottenere un'ottima saldatura di U1 è consigliabile cospargere la sagoma del componente disegnata sul circuito stampato con del flussante gel, questa sostanza ha la duplice funzione: attivare la saldatura favorendo la creazione del giunto di saldatura e quindi la formazione dell'intermetallo e di deossidazione dello stagno. Posizionato e saldato U1 con la solita prassi, potete procedere con il connettore micro SD ed il connettore per il display per il quale è richiesta una punta sufficientemente fine. In **Fig. 6** si può apprezzare la saldatura del connettore per il display dopo aver seguito accuratamente i consigli forniti.
Un ulteriore consiglio, quando si saldano componenti SMD, è quello di non eccedere con la temperatura della punta del saldatore, in genere è cattiva abitudine impostare una temperatura superiore ai 400 °C questa pratica non favorisce la saldatura ma anzi ne pregiudica la buona riuscita compromettendo talvolta anche il circuito stampato o il componente stesso. Infatti l'alta temperatura della punta fa evaporare rapidamente il flussante contenuto all'interno dello stagno o quello da voi aggiunto manualmente, con l'inevitabile accelerazione del processo di ossidazione della lega saldante che tende a diventare pastosa e poco saldabile. Impostando una temperatura molto bassa, compresa tra i 370 °C ed i 380 °C il flussante evapora meno rapidamente ed i benefici da esso apportato durano più a lungo consentendo di operare con tutta calma e tranquillità, inoltre questa temperatura inferiore evita di surriscaldare eccessivamente il circuito stampato ottenendo un risultato piacevole anche alla vista.
Il circuito non è particolarmente complesso per cui dopo aver saldato correttamente tutti i componenti a montaggio superficiale potrete proseguire con la saldatura dei componenti tradizionali per i quali è richiesta una minore manualità o precisione, ma allo stesso tempo cura e pazienza.

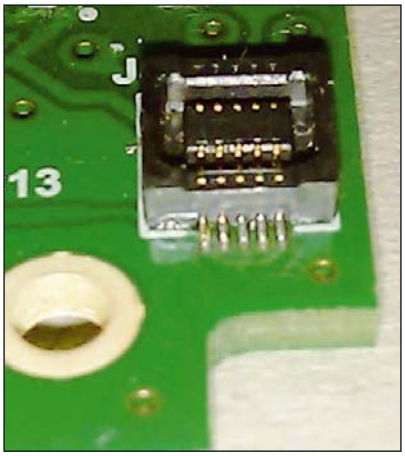


***Fig. 6** - Vista della saldatura del connettore per il display LCD Nokia. Il passo dei pin di connessione è molto ridotto, quindi facilmente cortocircuitabile in fase di saldatura. L'uso del flussante potrà evitare la formazione di corti circuiti.*

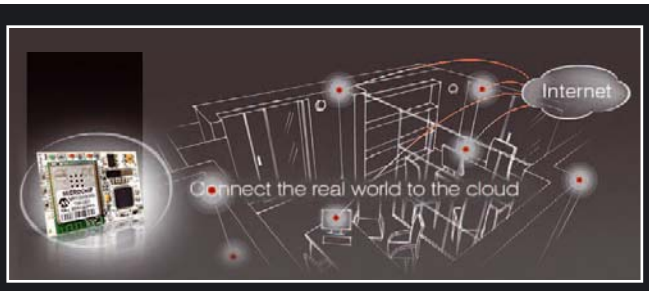
CONCLUSIONE

Per questo mese vi abbiamo presentato le caratteristiche hardware di questo progetto, descrivendovi il circuito elettrico, il principio di funzionamento ed alcuni dei componenti più importanti impiegati. Nella prossima puntata vedremo la descrizione del firmware e di come questo dispositivo si integra con altri progetti tramite interfaccia di comunicazione a due fili I2C.

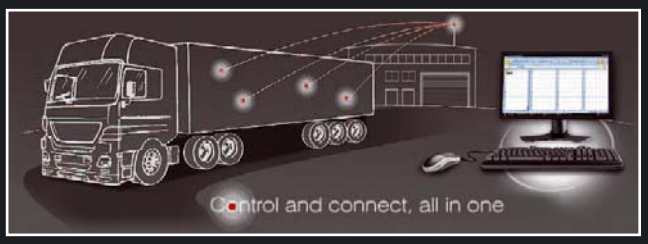
per il MATERIALE

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. Il lettore RFiD a 125 kHz ID12 costa 33,00 Euro 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







Alla scoperta di openPicus

Uno dei punti di forza della piattaforma openPicus è rappresentato dalla disponibilità di un ambiente di sviluppo molto completo e strutturato, quindi facile da usare, oltre che gratuito. Vi presentiamo l'introduzione all'IDE, le modalità di installazione ed i primi passi per la creazione di un progetto. Seconda Puntata.

2

di Gabriele Allegria

Uno dei punti forti del progetto OpenPicus è costituito dall'ambiente di sviluppo, un'IDE (Integrated Development Environment) fornita gratuitamente, che consente di gestire tutti gli aspetti legati alla creazione di un progetto. A partire dall'importazione delle pagine HTML per il webserver, alla gestione dei parametri di rete, alla programmazione del firmware ed al relativo trasferimento sul dispositivo.
Vediamo ora nel dettaglio come procedere per sviluppare una prima applicazione. Il primo passo consiste nello scaricare l'ambiente di sviluppo, disponibile per la piattaforma Windows, dal sito *www.openpicus.com* nella sezione Download. L'IDE richiede la presenza del framework .NET 4.0 sul PC. Nel caso il PC ne fosse sprovvisto sarà necessario installarlo.
Nella stessa sezione è presente anche un link per il download del compilatore C30 versione lite di Microchip, necessario per la compilazione del codice da parte dell'IDE. Allo stesso indirizzo sono presenti documentazione, librerie, alcuni webinar e progetti di esempio.

L'IDE, PRIMI PASSI E CREAZIONE DI UN NUOVO PROGETTO

Una volta lanciata l'IDE (**Fig. 1**) si ha la possibilità di creare un nuovo progetto o di caricarne uno preesistente. Al momento andremo a creare un nuovo progetto, che chiameremo "Hello world". Cliccando su "new project" viene data la possibilità di scegliere il **template** da utilizzare come base di partenza.
Di default sono caricati all'interno dell'IDE tre template: "WiFi 2.1 Basic no webserver", "WiFi 2.1 - Webserver blank example" e "WiFi 2.1 - Webserver example". Il primo è un template completamente vuoto, privo di webserver. Nel secondo si ha già un esempio un po' più avanzato, in cui il FlyPort instaura una connessione WiFi, ma senza integrare pagine web. L'ultima rende disponibile una gradevole pagina HTML in cui è possibile controllare alcuni ingressi digitali e analogici del dispositivo e attivare le uscite (**Fig. 2**).
Nel nostro caso scegliamo il terzo template, più completo rispetto agli altri e in grado di fornire un piccolo assaggio di quelle che sono le potenzialità del device.
Una volta completato il primo step, nella sezione "project explorer" vengono elencati i file di progetto:

- **taskFlyport.c**: è il file che conterrà l'applicazione dell'utente.
- **WF_Events.c**: in questo file vengono gestiti

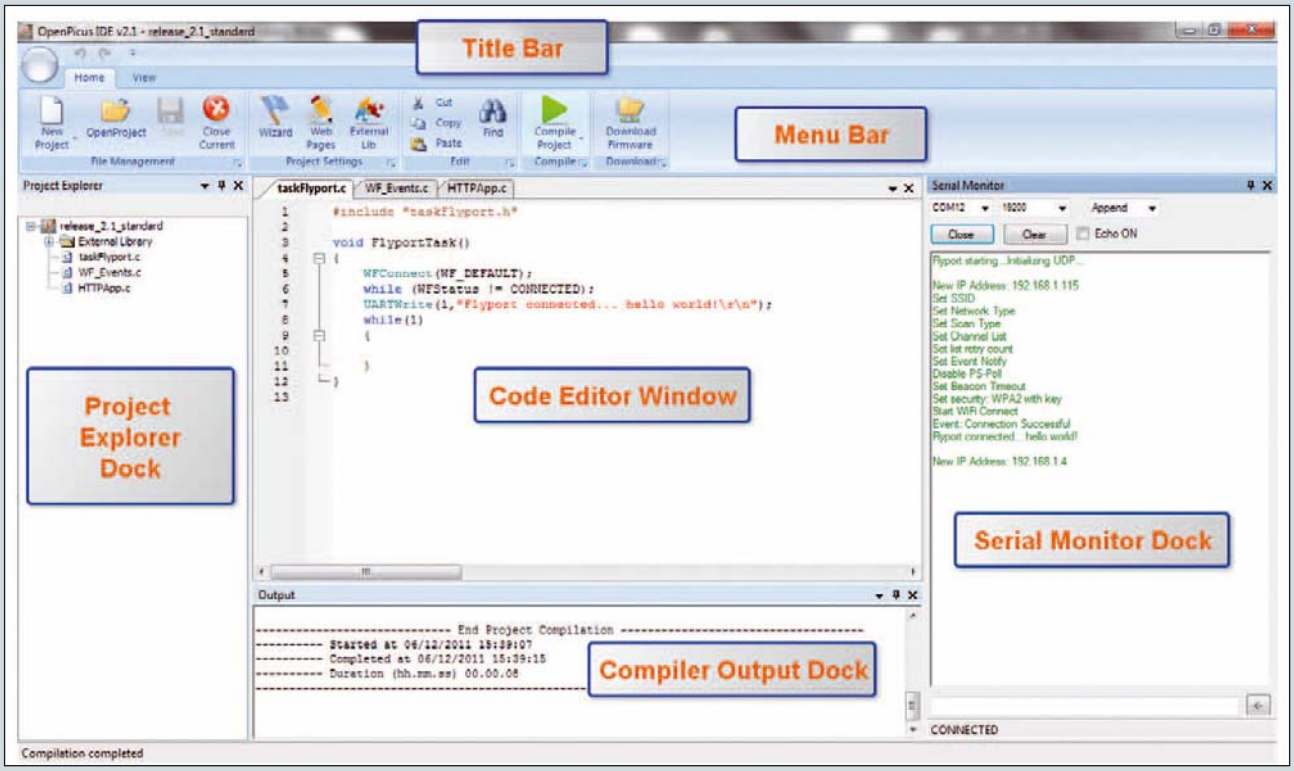


Fig. 1 - La suddivisione dell'interfaccia IDE.

tutti gli eventi relativi alla rete, come la connessione riuscita o fallita. In questo esempio, quando la connessione viene stabilita, viene acceso il led sull'uscita 5.

- **HTTPApp.c**: contiene il codice necessario alla gestione del webserver. È in questo file che vengono ricevute le richieste dal browser e gestite di conseguenza. Se si disabilita la funzionalità webserver, questo file non sarà presente nel project explorer nè nella finestra di editor del codice.

A questo punto è doverosa una piccola precisazione. Nella prima parte di questo tutorial abbiamo detto che il FlyPort integra un sistema operativo. Più specificatamente, i task operativi sono due, uno relativo all'applicazione utente, e uno che gestisce lo stack TCP/IP. Facendo riferimento ai file sopra elencati, il primo file viene eseguito nel primo task, ossia quello relativo all'applicazione utente, mentre gli altri due file, vanno ad agire sullo stack TCP/IP, e quindi verranno eseguiti nel secondo task. È di fondamentale importanza, quindi, evitare di inserire funzioni "bloccanti" all'interno di questi due file. Bloccare l'esecuzione di uno di questi file (magari rimanendo in attesa indefinita di un carattere sulla seriale, o che un ingresso si chiuda) comporta il blocco a tempo indefinito dell'esecuzione dello stack TCP/IP, e quindi un concreto rischio di compromettere la connettività con la rete. Al contrario, il task relativo al codice utente (il task "FlyportTask") può essere bloccato senza nessun problema proprio perchè non influisce direttamente sull'esecuzione dello stack, i due task sono indipendenti, quindi se si blocca il primo, lo stack continua a girare tranquillamente. Questo ci mette al riparo anche da eventuali bug bloccanti sul task utente, che non andrebbero a compromettere la funzionalità dello stack e, di conseguenza, del webserver.

Torniamo ora al nostro progetto. La prima cosa da fare è configurare correttamente il nostro dispositivo, per far sì che si adatti alle nostre esigenze e alla nostra rete Wi-Fi. Questa configurazione avviene in maniera "visuale", attraverso un **wizard di configurazione** cui si accede premendo appunto il pulsante "wizard" (**Fig. 3**).
Diamo un'occhiata alla prima pagina del wizard.
Come si può vedere dalla figura, è possibile selezionare quali sono i servizi che si vogliono attivare sullo stack, come ad esempio il webserver, il DHCP client o server (utile nel caso



Fig. 2 - La pagina web disponibile di default con il template standard.

di connessioni ad hoc, per assegnare un indirizzo ad un client). È possibile anche attivare il debug sulla seriale 1 (con il baudrate 19200) in modo da ricevere un feedback diretto dallo stack sulla seriale riguardo l'indirizzo IP del dispositivo e in generale sugli eventi relativi al Wi-Fi. Le pagine successive del wizard riguardano principalmente la configurazione dei parametri di rete, sarà quindi possibile decidere se creare una rete ad hoc o connettersi ad un access point preesistente, specificare indirizzo IP, gateway e subnet mask proprio come si fa per qualunque dispositivo di rete. Sarà inoltre possibile decidere il livello di sicurezza della rete, infatti il Flyport può connettersi sia a reti aperte, che a reti protette con crittografia WEP o WPA/WPA 2.
La pagina del wizard intitolata *TCP Socket Configuration*, è anch'essa dedicata alla customizzazione dello stack, in particolare è possibile definire il numero di socket che si andranno ad usare e il relativo buffer per ognuna di esse.
Una volta configurato opportunamente il dispositivo si può salvare il lavoro fatto, in modo da non dover rifare tutte le volte lo stesso iter. Sarà sufficiente selezionare "save as template" per salvare le impostazioni e il proprio codice in un unico file .zip. Posizionando questo file nella cartella ..\Template\WiFi\ all'interno del percorso dell'IDE, sarà possibile, al successivo riavvio, avere il proprio progetto disponibile come template, e quindi pronto ad essere avviato senza bisogno di nessun tipo di configurazione. Nel template vengono salvati anche eventuali cambiamenti effettuati sulle pagine del webserver e sul codice relativo.
Un'altra opzione interessante riguarda la possibilità di importare eventuali **librerie esterne**, che consentono di semplificare e standardizzare operazioni specifiche, come interfacciarsi con sensori esterni (molte librerie sono presenti sul sito *www.openpicus.com*, nella sezione download). Il processo di importazione è molto semplice, è sufficiente selezionare il pulsante "external lib", da cui si apre una finestra di selezione che consente l'importazione di eventuali file .c, .h o .s (file assembler). Automaticamente verranno importati nel progetto e, alla chiusura della finestra, sarà possibile trovarli nel project explorer, per editarli. All'interno del programma dovranno essere richiamati i file header, attraverso la classica direttiva `#include` del linguaggio C.
Veniamo dunque ad uno dei punti di forza di questo piccolo ma potente device, ossia il **webserver**. Come dicevamo in precedenza è possi-

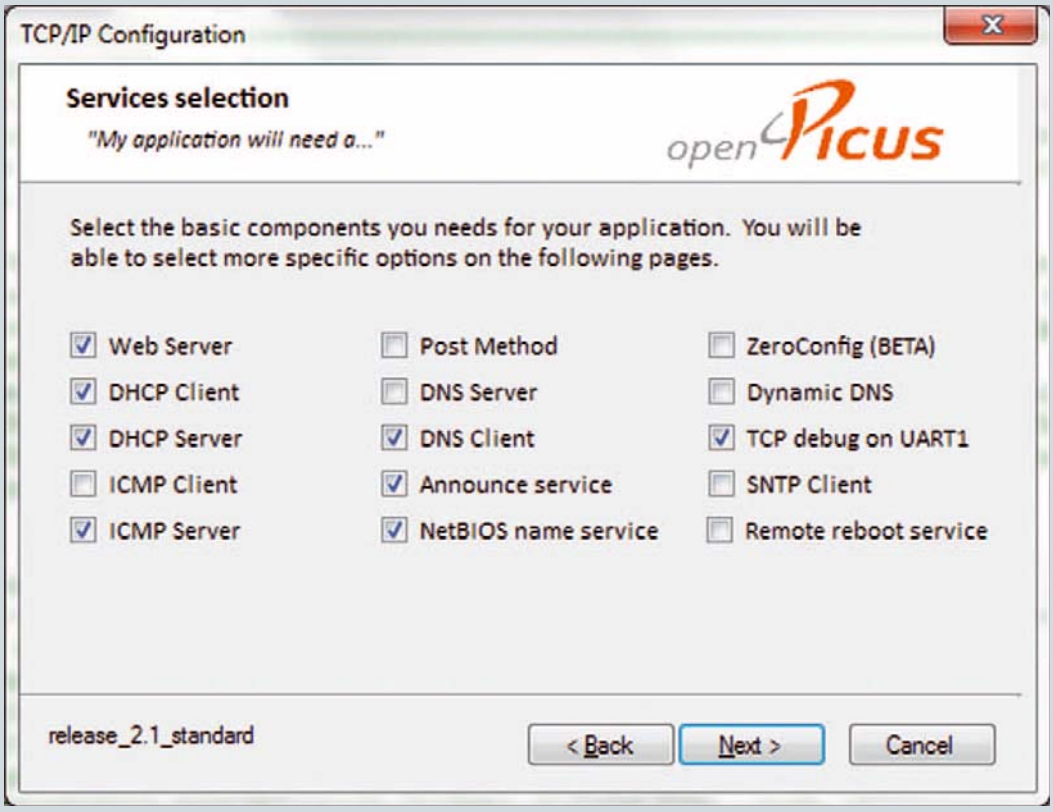


Fig. 3 - La prima pagina del wizard.

bile integrare all'interno del proprio progetto delle vere e proprie pagine web, che possono anche contenere javascript che andranno poi a interagire con il nostro firmware e quindi con l'hardware. Il primo passo è creare le nostre pagine HTML, aggiungendo eventuali fogli di stile css, immagini, suoni o animazioni. La limitazione è, al momento, data dalla dimensione della memoria del dispositivo, ma attraverso un sapiente uso di immagini non troppo pesanti è possibile realizzare anche pagine estremamente gradevoli, come dimostra l'esempio fornito nel template di base. Per inserire il proprio "sito" all'interno del webserver non bisogna far altro che inserire tutti i file necessari all'interno di una cartella, all'interno della quale possono anche essere organizzati in ulteriori sottocartelle, e selezionare il pulsante "web pages". A quel punto sarà possibile selezionare la cartella in cui è presente il sito e confermare, e così l'IDE andrà in automatico ad importare pagine HTML, fogli di stile, javascript e ogni altro file presente all'interno della cartella selezionata. A quel punto il sito è già importato all'interno del nostro progetto. La homepage predefinita è index.htm. Si può dare un'occhiata a come sono organizzate le pagine web del template standard guardando la cartella \webpages all'interno della nostra cartella di progetto. Una volta terminata l'importazione delle pagine web, si può procedere alla scrittura della propria applicazione, compilare il tutto attraverso Compile project -> Compile ed in seguito scaricare il firmware sul dispositivo, attraverso il pulsante "Download". Una volta terminata l'operazione, è possibile monitorare il funzionamento del nostro firmware attraverso il monitor seriale, in pratica una sorta di hyper terminal interno all'IDE, in modo da avere il tutto integrato nello stesso ambiente di sviluppo, senza dover aprire altri programmi. Nella prossima puntata cominceremo ad approfondire la programmazione del FlyPort, commentando il codice del template standard e scrivendo qualche primo semplice programma.

I moduli FlyPort Wi-Fi sono disponibili nella versione con antenna integrata (Cod: 8100-FLYPORTWI-FI, Euro 59,90) e con connettore per antenna esterna (Cod: 8100-FLYPORTWIFI-uFL, Euro 59,90) presso:
Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331/799775 - Fax:0331/792287
http://*www.futurashop.it*

smau

Save the date:
mercoledì 17
giovedì 18
venerdì 19 Ottobre
fieramilanocity

smau business

Vieni in fiera per decidere come innovare la tua impresa!

I principali player del mondo ICT pronti a illustrare le ultime novità nell'ambito delle tecnologie digitali **a supporto del tuo business.**



NOVITÀ 2012 Tablet e cloud per il business

App? Tablet? Cloud? A Smau tutto quello che la tua azienda non può non sapere sui dispositivi, le tecnologie e i servizi che rivoluzioneranno per sempre il nostro modo di lavorare.

Ti aspettano | i migliori fornitori, le School of Management più prestigiose e tutte le anteprime di prodotto.

Smau Mob APP Awards | Le migliori APP in ambito business e consumer basate sui dispositivi mobili di nuova generazione (new tablet, pad e smartphone).

Scopri anche...

School of Management

- ✓**Cloud Computing** e ICT as a Service
- ✓**Fatturazione** Elettronica e **Gestione Documentale**
- ✓**Unified Communication** & Collaboration
- ✓**Progettazione** e Product Lifecycle Management
- ✓**Mobile** business

SDA Bocconi School of Management

Workshop esclusivi di **aggiornamento per la formazione manageriale**. Partecipa e riceverai un attestato SDA Bocconi

arena SOCIAL MEDIA

Tutto quello che non puoi non sapere su **facebook**, **twitter**, **google adwords**, advertising online, ...

smau trade

3ª edizione

Giunto alla sua 3ª edizione, **Smau Trade** si è affermato come il più importante appuntamento dell'ICT Community **per fare il punto sulle nuove sfide offerte dall'industria mondiale delle tecnologie digitali**: dal cloud computing, alle unified communication & collaboration, alla virtualizzazione, e molto altro ancora.

Una piattaforma unica di incontro e networking tra i maggiori **Vendor, i distributori e il mondo delle terze parti** - rivenditori, VAR, system integrator, software house - dove scoprire i programmi di recruiting e le politiche di fidelizzazione dei grandi Player ICT e prendere parte alle sessioni di training e certificazione dedicati ai professionisti del comparto.

smart city ROADSHOW

2ª edizione

un'iniziativa di: smau

La via italiana alle città intelligenti

Dopo il successo della 1ª edizione del 2011, un nuovo appuntamento interamente dedicato al tema delle Smart City e alle più interessanti **progettualità evolute** in atto sul territorio italiano. Dedicato a **Sindaci** e **Assessori** dei Comuni italiani per capire come le moderne tecnologie e i principali fornitori possano supportarli concretamente nel cammino verso lo sviluppo sostenibile della loro città.

! Ingresso riservato a professionisti e manager, dirigenti e operatori della Pubblica Amministrazione e al mondo del Canale ICT previa registrazione al sito **www.smau.it**

contatti: **tel.** 049 8808444 - **email** contact@smau.it - **www.smau.it**



I telecontrolli che abbiamo pubblicato in questi anni possono essere proficuamente utilizzati come teleallarmi, perché una delle loro funzioni è inviare SMS o chiamate telefoniche al verificarsi di una variazione di stato o di una condizione elettrica sugli ingressi di cui dispongono. Ci siamo però resi conto che a qualcuno ciò non basta, perché sentire squillare il telefono o una serie di toni non è come una voce che ci dice chiaramente cose del tipo: "incendio in corso", "ladri in casa" ecc... Abbiamo dunque pensato di progettare e proporvi un teleallarme, basato sull'hardware dei nostri telecontrolli ma dotato di sintesi vocale; visto che c'eravamo, abbiamo previsto la possibilità di accettare comandi via DTMF, il che ci consente di configurarlo a distanza con semplicità e rapidità. Questo nuovo dispositivo può quindi essere comandato chiamando il numero associato alla sua SIM, attendendo la risposta e premendo i tasti della tastiera corrispondenti ognuno ad una specifica funzione; per ragioni di sicurezza, accetta le chiamate provenienti solo da una ristretta cerchia di numeri telefonici (8) che bisogna memorizzare in un'apposita lista (numeri abilitati) prima di utilizzarlo oppure da qualsiasi numero, purché completati dalla password. I comandi riguardano la gestione dei relé di uscita, la lettura dello stato degli ingressi, oltre ad alcune impostazioni di funzionamento.

Il teleallarme può effettivamente essere considerato un combinatore vocale GSM, dato che attivandone un ingresso effettua telefonate durante le quali riproduce messaggi a voce.

Abbandoniamo dunque i preamboli e diciamo subito che il sistema è formato da tre schede montate a sandwich: quella base è sostanzialmente il nostro TDG133 descritto nel fascicolo n° 148 (cui rimandiamo per eventuali approfondimenti e dettagli qui non spiegati) ed ospita il microcontrollore cui è affidata la gestione dell'intero teleallarme, oltre ai relé per l'attivazione di azioni locali e ai fotoaccoppiatori per rilevare le condizioni di allarme agli ingressi; su questa scheda si montano "a sandwich" il circuito contenente il modulo GSM/GPRS e quello che ospita la sintesi vocale cui è affidato il compito di registrare i messaggi e riprodurli quando richiesto. Nella stessa risiedono i messaggi del menu vocale che il sistema presenta quando riceve una chiamata dall'utente.

SCHEDA BASE

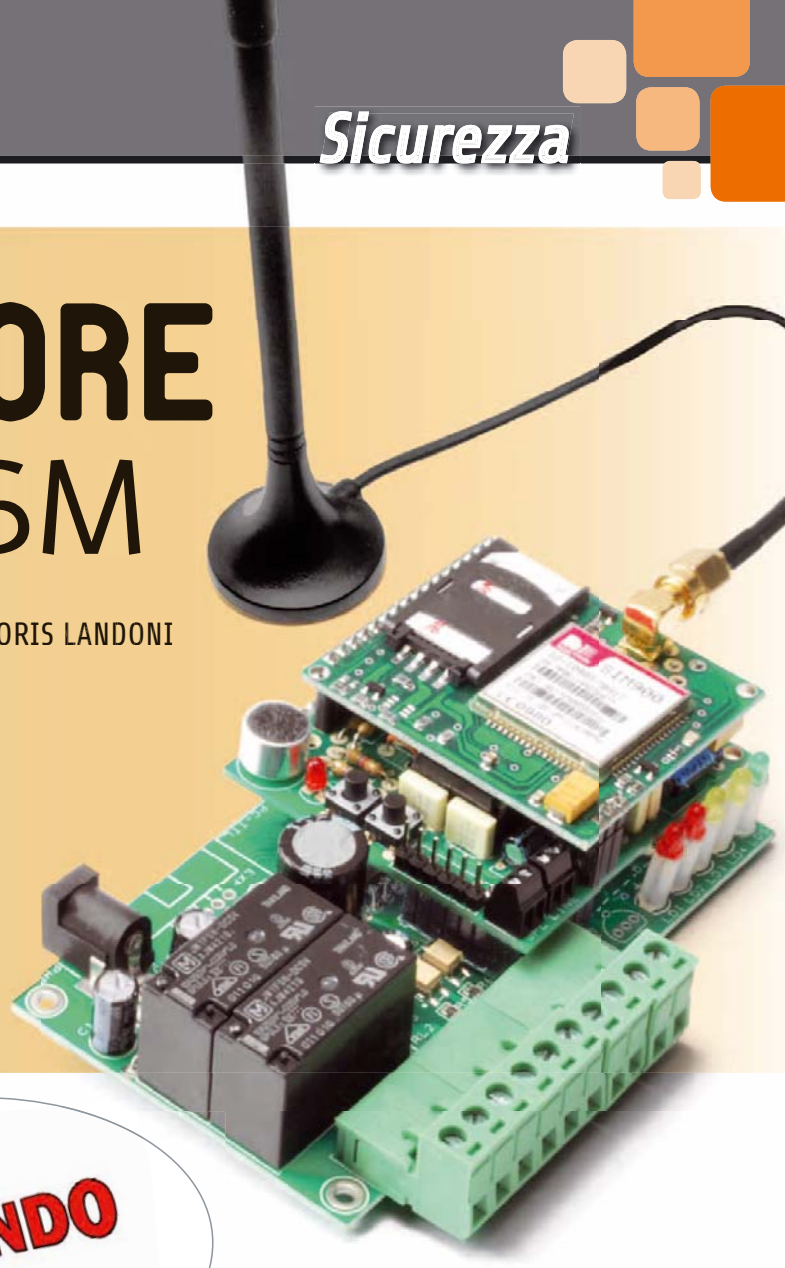
Analizzando il circuito dell'unità base: i più attenti di voi, dallo schema elettrico avranno notato che manca un componente, ossia il decoder DTMF. Eppure il nostro combinatore si gestisce via DTMF, quindi dov'è l'arcano? Ebbene, possiamo fare a meno dell'8870, cioè del decoder multifrequenza che ha sempre equipaggiato i nostri telecontrolli, perché dall'inizio dell'anno la SIMCom, produttrice del modulo GSM/GPRS che montiamo nel circuito, ha rilasciato un upgrade firmware che permette al suo SIM900 di gestire la decodifica dei bitoni standard DTMF. Ciò vuol dire che tutti i nuovi SIM900 sono in grado, mediante appositi comandi dall'esterno, di informare il microcontrollore che governa il nostro circuito dell'arrivo dei bitoni, distinguendo chiaramente un bitono dall'altro.

Detto questo, passiamo a descrivere il circuito del teleallarme, con una premessa: il cellulare e la sintesi vocale si

COMBINATORE VOCALE GSM

di BORIS LANDONI

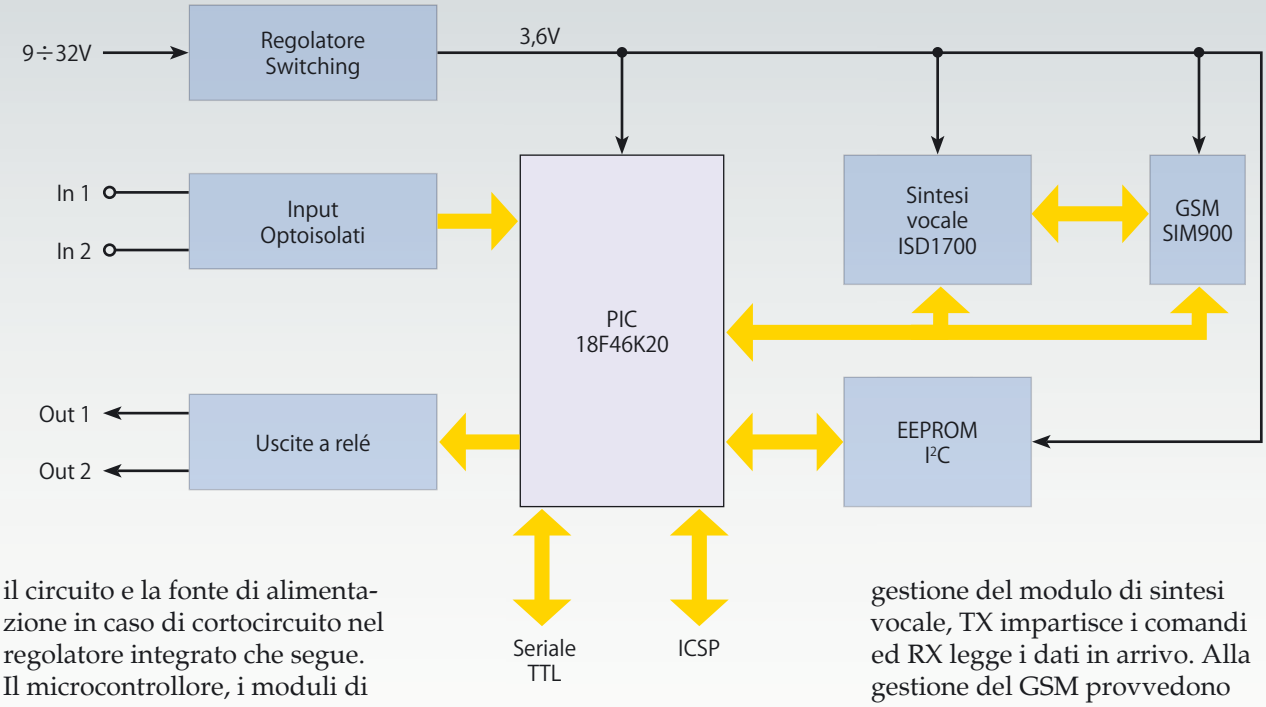
Collegato ad una centrale antifurto o antincendio, in caso di allarme effettua chiamate telefoniche riproducendo messaggi vocali. Comandato via DTMF può azionare attuatori sul posto. Prima puntata.



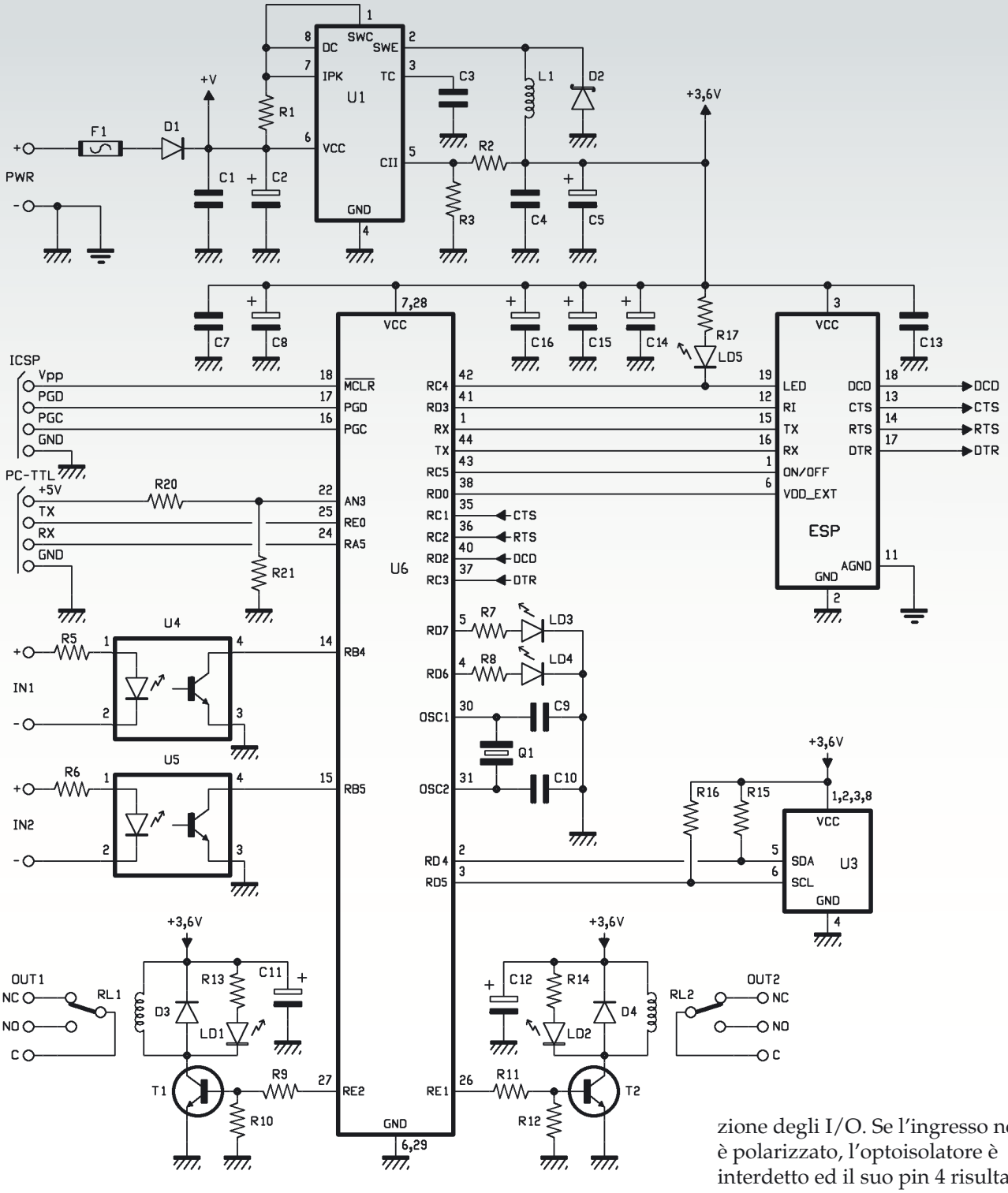


montano sullo zoccolo che nello schema elettrico viene siglato ESP (espansione) quindi i collegamenti e i segnali che arrivano ad esso e vi partono sono riferiti al blocco delle due schede. Bene, partiamo con l'analisi della scheda base vedendo il suo componente principale, ossia il microcontrollore che governa l'attività del modulo GSM/GPRS, controlla la condizione logica dei due ingressi optoisolati e comanda i due relé. L'alimentazione è a tensione continua, anche non stabilizzata (applicata ai punti + e - PWR) di valore compreso tra 9 e 32 V; viene filtrata a valle del diodo di protezione dall'inversione di polarità (D1) mediante i condensatori C1 e C2. Il fusibile F1 ci permette di proteggere

Schema a blocchi

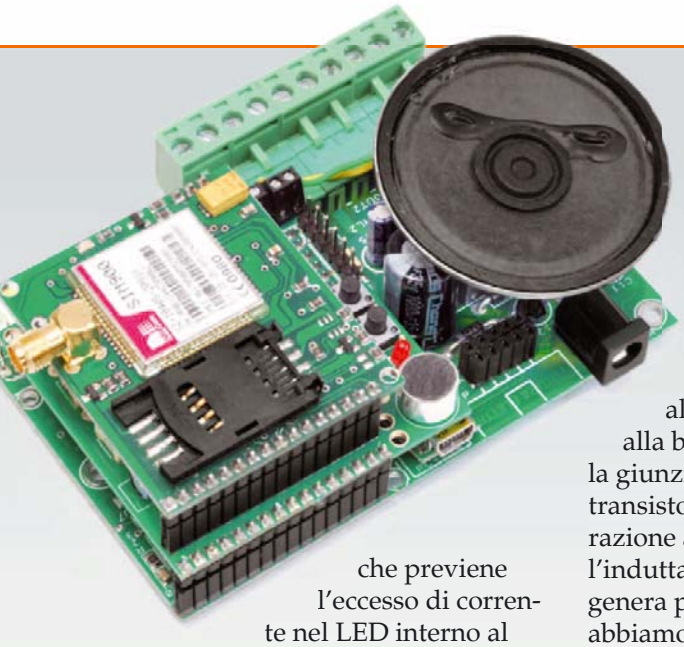


il circuito e la fonte di alimentazione in caso di cortocircuito nel regolatore integrato che segue.
Il microcontrollore, i moduli di sintesi vocale e GSM, nonché il resto del circuito, vengono fatti funzionare a 3,6 V, ricavati da uno switching (U1) basato sull'integrato MC34063 e utilizzato come regolatore PWM serie a carica d'induttanza, la cui tensione di uscita dipende dall'energia immagazzinata in L1; la tensione d'uscita viene fissata dalla retroazione operata dal partitore resistivo R2/R3, che retrocede al piedino 5 parte della componente d'uscita. C4 e C5 livellano gli impulsi prodotti dall'U1. La tensione a valle di questi condensatori viene poi ulteriormente filtrata da C7, C8, C13, C14, C15 e C16, situati sulle linee di alimentazione del microcontrollore e del modulo GSM, che servono a sopperire ai picchi di assorbimento del cellulare, evitando la propagazione alla logica degli impulsi di disturbo che ne conseguono.
Il microcontrollore usato per gestire il telecontrollo è un PIC18F46K20-I/PT il quale, una volta inizializzate le linee di I/O, verifica la condizione logica degli ingressi optoisolati a livello di tensione (RB4 ed RB5) e quella delle linee RC4, RC5, RD0, RD3, RX, che servono a ricevere dal modulo cellulare le principali segnalazioni; precisamente, RD3 viene usata per rilevare l'arrivo delle chiamate entranti (si interfaccia con RI del modulo cellulare) mentre RC4 controlla il "LED di campo" del GSM, la cui uscita LED pulsa alla frequenza di 1 Hz quando il modulo sta cercando la rete telefonica, mentre fornisce un impulso a zero logico della durata di 0,5 s seguito da una pausa di 2 s quando il modulo ha agganciato il segnale.
L'UART interno al PIC, accessibile dai piedini 44 (trasmissione) ed 1 (ricezione) serve ad impartire i comandi al cellulare ed al modulo vocale, che condividono lo stesso bus; per quanto riguarda il modulo GSM, mediante il TX lo interroga ciclicamente allo scopo di verificare l'arrivo di SMS, mentre TX ed RX insieme servono, durante le procedure di chiamata, ricezione e trasmissione messaggi, per la comunicazione tra il microcontrollore e il GSM. Per quanto riguarda la gestione del modulo di sintesi vocale, TX impartisce i comandi ed RX legge i dati in arrivo. Alla gestione del GSM provvedono anche RC5 ed RD0: la prima comanda l'accensione e lo spegnimento del GSM (tramite un transistor sistemato nella basetta del cellulare) mentre la seconda effettua il reset del cellulare. I segnali di controllo CTS (Clear To Send) RTS (Request To Send) e DCD (Data Carrier Detect) in realtà non vengono usati come tali, ma formano il bus SPI a quattro fili che il microcontrollore usa per dialogare con l'integrato della sintesi vocale; non arrivano al cellulare in quanto sulla scheda base si monta il modulo vocale e su quest'ultimo prende posto il circuito del cellulare e il c.s. della sintesi vocale ha i collegamenti passanti eccetto che per tali linee, le quali si fermano su di esso.
E vediamo ora gli ingressi di allarme, che vengono letti mediante gli I/O del micro RB4 ed RB5, entrambi configurati come input e provvisti di resistore di pull-up interno; ognuna legge lo stato del transistor di uscita del fotoaccoppiatore corrispondente. Ogni ingresso (IN1 e IN2) è attivo quando sottoposto ad una tensione di valore compreso fra 3 e 30



V. Il collettore di ciascun optoisolatore è collegato ad un pin del micro (14 per U4 e 15 per U5). Quando all'ingresso IN1 è applicata una tensione di almeno 3 volt, il fotoaccoppiatore ha il LED acceso e il fototransistor d'uscita in conduzione, cosicché il collettore (piedino 5) è a circa zero volt, per effetto della caduta sul resistore di pull-up interno configurato durante l'inizializzazione degli I/O. Se l'ingresso non è polarizzato, l'optoisolatore è interdetto ed il suo pin 4 risulta a livello alto.

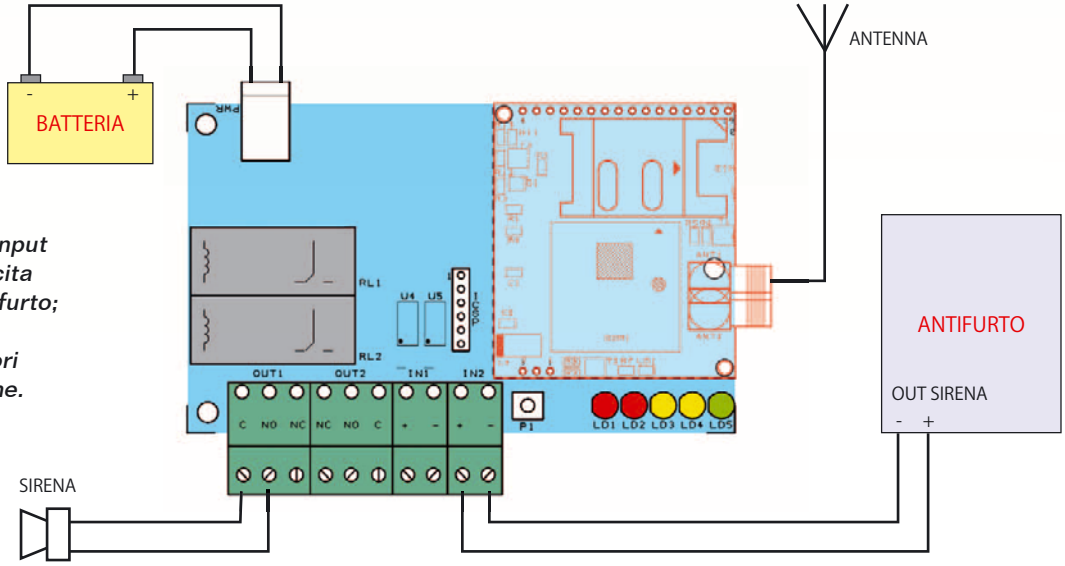
La scelta di dotare IN1 e IN2 di fotoaccoppiatori nasce per mantenere il microcontrollore galvanicamente isolato dal circuito sul quale va a leggere la tensione; ciascuna linea di ingresso viene inoltre protetta da un resistore



che previene l'eccesso di corrente nel LED interno al fotoaccoppiatore corrispondente. Attualmente le resistenze sono da 4,7 kohm, ma si possono modificare dimensionandole con la formula:

R = (Vin - Vled) / 0,008

in cui Vin è la tensione in ingresso, Vled è la caduta di tensione del diodo (2 V circa) e 0,008 è la corrente diretta del LED.
Quanto ai relé, vengono comandati dalle linee RE1 ed RE2 del micro, mediante due transistor NPN usati da amplificatori di corrente; le linee RE1 ed RE2 comandano i transistor, rispettivamente, T1 e T2. Lo stato logico alto manda in saturazione il transistor, determinando lo scorrimento di corrente nella bobina del relé corrispondente. L'attivazione viene segnalata con un LED alimentato in parallelo alla bobina. Per proteggere la giunzione di collettore dei transistor nei passaggi dalla saturazione all'interdizione, quando l'induttanza della bobina del relé genera picchi di tensione inversa, abbiamo collegato un diodo in parallelo alla bobina, che sopprime tali impulsi.
Dei relé viene reso accessibile tutto lo scambio, così da poter gestire circuiti che richiedono un contatto normalmente chiuso o uno normalmente aperto.
Vediamo adesso come avviene la comunicazione con il PC per effettuare la configurazione: allo scopo è prevista un'interfaccia seriale cui si può applicare il convertitore TTL/USB per interfacciare l'unità via USB. Avendo già impegnato l'UART per il dialogo con il modulo cellulare, la comunicazione seriale avviene mediante le linee RE0 ed RA5, sfruttate rispettivamente come TX ed RX; la AN3, assegnata all'A/D converter interno, ci serve per rilevare la presenza dei 5 volt e quindi l'inserzione del connettore nel computer.
Il convertitore USB/TTL che suggeriamo è l'FT782M, già usato in altri progetti (si può acquistare in versione già montata e collaudata presso Futura Elettronica, *www.futurashop.it*). Per la connessione con il PC è previsto un software di configurazione in ambiente Windows, che descriveremo nella prossima puntata.
I dati di configurazione non vengono salvati nella EEPROM del microcontrollore, ma in una memoria esterna di tipo 24FC256-SN; si tratta di una EEPROM CMOS da 256 kbit ad interfaccia I²C-Bus. Per dialogare con essa, il micro usa RD4 ed RD5, inizializzate per funzionare rispettivamente come SDA (linea dati) ed SCL (linea di clock). L'aver spostato i dati di configurazione del telecontrollo in una memoria esterna ci permette di riservare la EEPROM interna del microcontrollore ai dati di funzionamento, consentendoci di memorizzare più dati di configurazione o di aggiungere funzionalità, che

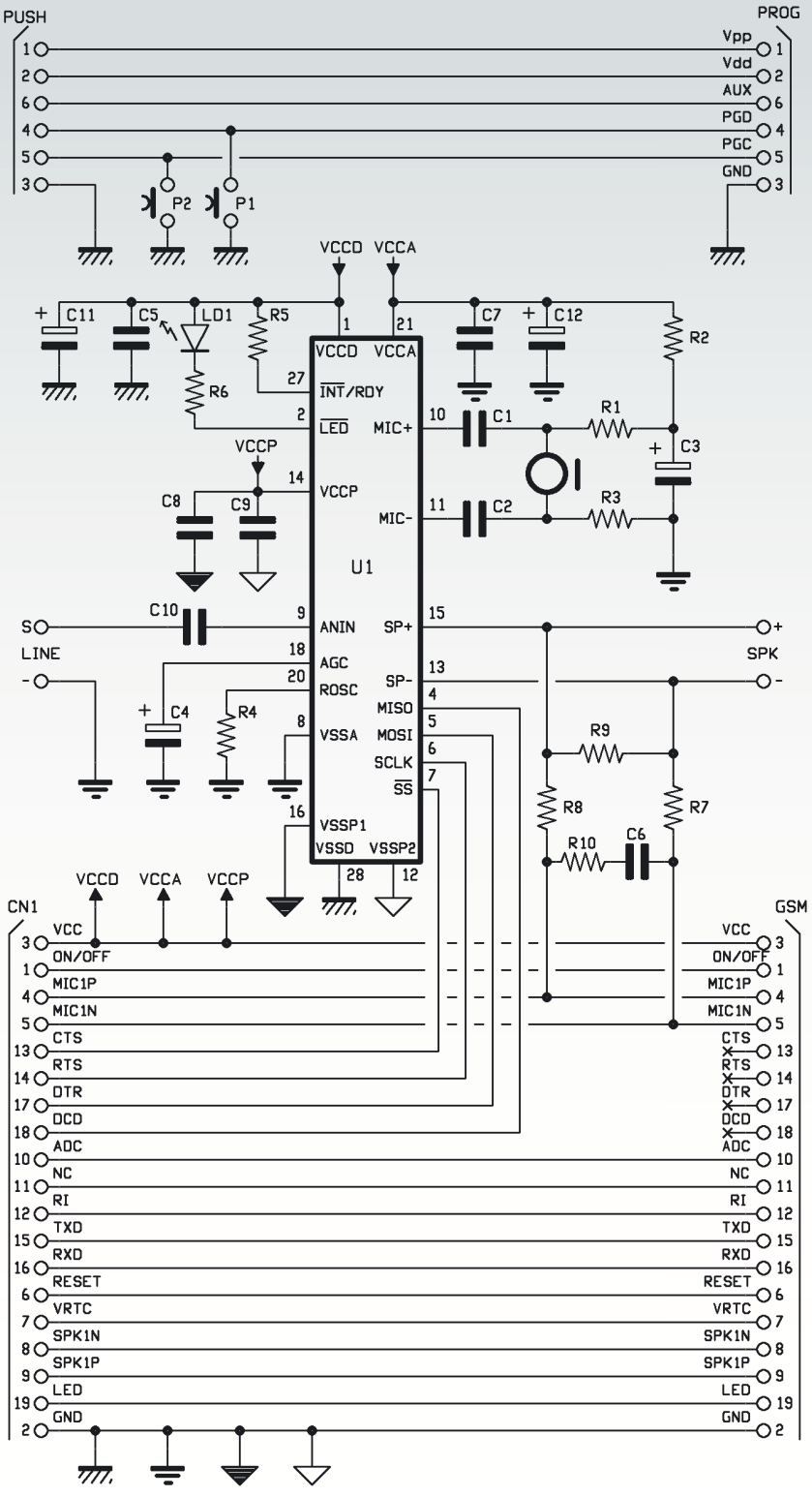


Almeno uno degli input va collegato all'uscita di allarme dell'antifurto; le uscite possono comandare attuatori locali come le sirene.

inevitabilmente renderebbero il firmware più pesante.
Per caricare il firmware nel micro abbiamo predisposto un'interfaccia ICSP, facente capo alle linee /MCLR (piedino 18) PGD (piedino 17) e PGC (pin 16) dell'U6 e del connettore ICSP; a quest'ultimo giunge anche il collegamento di massa della scheda.
La connessione ICSP da noi prevista è compatibile con il PicKit2.

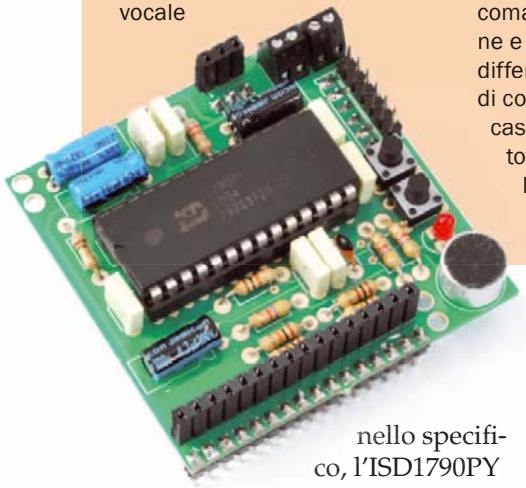
SINTESI VOCALE

Bene, ora possiamo spendere qualche parola sulla sintesi vocale, realizzata dall'integrato ISD1790PY: si tratta di un vero e proprio registratore allo stato solido composto da un circuito di campionamento del segnale in ingresso, una particolare memoria analogica (detta MLS=Multi Level Storage) ed un blocco di uscita che in riproduzione ricostruisce l'audio memorizzato. Diversamente da quanto ci si potrebbe aspettare, questo integrato utilizza la tecnica DAST sviluppata e brevettata dalla ISD (Information Storage Devices) che consiste nel memorizzare direttamente dei campioni di audio, anziché dati digitali ottenuti convertendo con un ADC i campioni stessi; ciò permette innanzitutto di ottenere una maggiore densità di memorizzazione, giacché una locazione memorizza il livello di tensione istantaneo corrispondente ad un campione e non un bit facente parte di un byte che descrive in digitale quel campione analogico.
In altre parole, mentre registrando in digitale, un campione (cioè un frammento temporale) del segnale da registrare occupa un certo numero di bit (in base alla risoluzione che si desidera ottenere) e perciò di celle di memoria, con la tecnica DAST richiede una singola cella. Inoltre la fedeltà audio è migliore, in quanto per ogni campione si memorizza il livello reale, non una sua ricostruzione digitale, la quale essendo composta da valori discreti, quando viene riprodotta non è più uguale.
L'integrato ISD1790PY appartiene alla famiglia ISD1700 della Nuvoton, che consta di diversi dispositivi caratterizzati ognuno da una durata di registrazione;

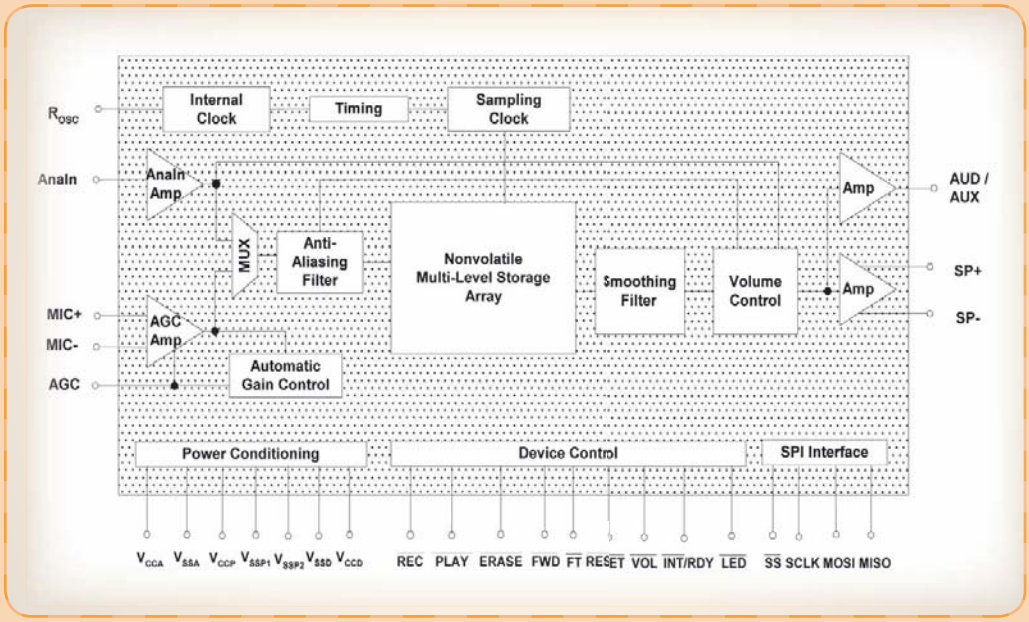


La **sintesi vocale**

I messaggi audio che il nostro sistema di teleallarme trasmette, sono riprodotti da un integrato (incapsulato in un contenitore DIP standard a 14+14 pin) per sintesi vocale appartenente alla serie ISD1700 della Nuvoton (*www.nuvoton.com*), che consta di integrati in grado di registrare e riprodurre da 20 secondi ad otto minuti di parlato o musica; il campionatore di questi componenti può lavorare a frequenze di clock comprese tra 4 e 12 kHz, come è consuetudine negli integrati per sintesi vocale



nello specifico, l'ISD1790PY ha una capienza di 90 secondi. Il componente dispone di un ingresso microfonico con controllo automatico del guadagno (AGC), un'uscita per altoparlante da 8 ohm e un'altra uscita analogica separata che può essere configurata per pilotare un amplificatore esterno; dispone delle funzioni di controllo del volume, vAlert (voiceAlert) che può essere utilizzata per indicare che è presente in memoria un nuovo messaggio, filtro anti-alias e smoothing filter. Prevede inoltre quattro effetti sonori speciali per confermare acusticamente le operazioni di inizio registrazione, arresto registrazione, cancellazione, avanti e cancellazione totale.
Sebbene l'ISD1790PY possa essere comandato direttamente tramite livelli logici e pulsanti, della ISD. Per ogni componente è prevista la possibilità di scegliere la frequenza di campionamento in modo da ottenere il giusto compromesso tra fedeltà e durata di registrazione; l'impostazione si effettua variando il valore del resistore che il costruttore prevede sia collegato tra il piedino Rosc (20) e massa.
La serie ISD1700 offre due possibilità di controllo: tramite un'interfaccia SPI (ideale per gestire i chip di sintesi vocale mediante un microcontrollore) o attraverso i soliti piedini di comando della registrazione e della riproduzione. La differenza tra i due metodi di controllo è che nel primo caso è possibile gestire totalmente l'integrato e lo spazio di memoria disponibile, partizionando la EEPROM in settori indirizzabili secondo un preciso protocollo e quindi registrando e riproducendo molti messaggi, mentre con il comando dai piedini di controllo REC (comando di registrazione a livello: il chip registra finché tale pin è mantenuto fisso a livello basso), PLAY (riproduzione: basta dargli un impulso a zero logico ed il chip riproduce l'intero contenuto della memoria) ed ERASE (cancellazione messaggio) si può agire solo sulla totalità della memoria disponibile. Per la riproduzione di messaggi preventivamente registrati partizionando la memoria (con il comando da bus SPI) è possibile usare il piedino FWD, che consente di scorrere tra i messaggi da riprodurre.
Il chip può registrare più messaggi: se si registra per meno della durata della memoria, la successiva registrazione parte dalla fine del messaggio precedente.
La serie ISD1700 dispone di ingressi audio separati per microfono (MIC, differenziale) e linea (AnaIn, sbilanciata) e di due uscite distinte, una single-ended a basso livello per amplificatori (piedino AUD/AUX) ed una a ponte, amplificata, in grado di pilotare direttamente un piccolo altoparlante.
Come tutti gli integrati della serie ISD, anche gli ISD1700 sono facili da usare e per nulla critici; l'unica raccomandazione per limitare il rumore di fondo è rispettare la separazione delle masse della sezione digitale da quelle della parte analogica, come in questo caso lo gestiamo col microcontrollore, sfruttando l'interfaccia SPI di cui dispone; ecco perché P1 e P2, che pure si trovano nel modulo vocale per registrare i messaggi, in realtà non sono collegati all'integrato ma raggiungono, tramite i connettori del modulo stesso, il microcontrollore.
Infatti i pulsanti effettuano la richiesta al microcontrollore, che risponde comandando l'ISD; invece il LED di segnalazione di stato, che indica le fasi di inizio e fine registrazione, ma anche la fine del messaggio in riproduzione, viene comandato direttamente dall'apposita uscita (piedino 2) dell'U1.
Vediamo dunque le connessioni del circuito, che fanno capo a quattro connettori: PUSH, PROG, CN1 e GSM; i primi due sono semplicemente collegamenti passanti che portano alla connessione di programmazione del microcontrollore sulla scheda base e sulle loro linee 4 (PGD) e 5 (PGC) abbiamo applicato i pulsanti di controllo della registrazione e riproduzione, P1 e P2. Tali pulsanti vengono letti dal microcontrollore della scheda base per decidere le azioni da intraprendere.
CN1 e GSM sono passanti, eccetto che per le quattro linee CTS, RTS, DTR e DCD: questo perché il bus SPI che il micro usa per dialogare con l'ISD1790PY fa capo alle linee CTS, RTS, DTR e DCD del connettore che porta alla scheda base che in origine è stato previsto per interfacciarsi con il cellulare GSM; per l'esattezza, le predette linee sarebbero quelle dello zoccolo di espansione ESP della scheda base, che in questo caso non vengono usate dal cellulare e quindi montando il modulo vocale servono ad interfacciare l'SPI del chip ISD.
Dunque, CTS corrisponde ad /SS (identificazione del dispositivo Slave del bus) RTS coincide con SCLK (clock del bus) DTR

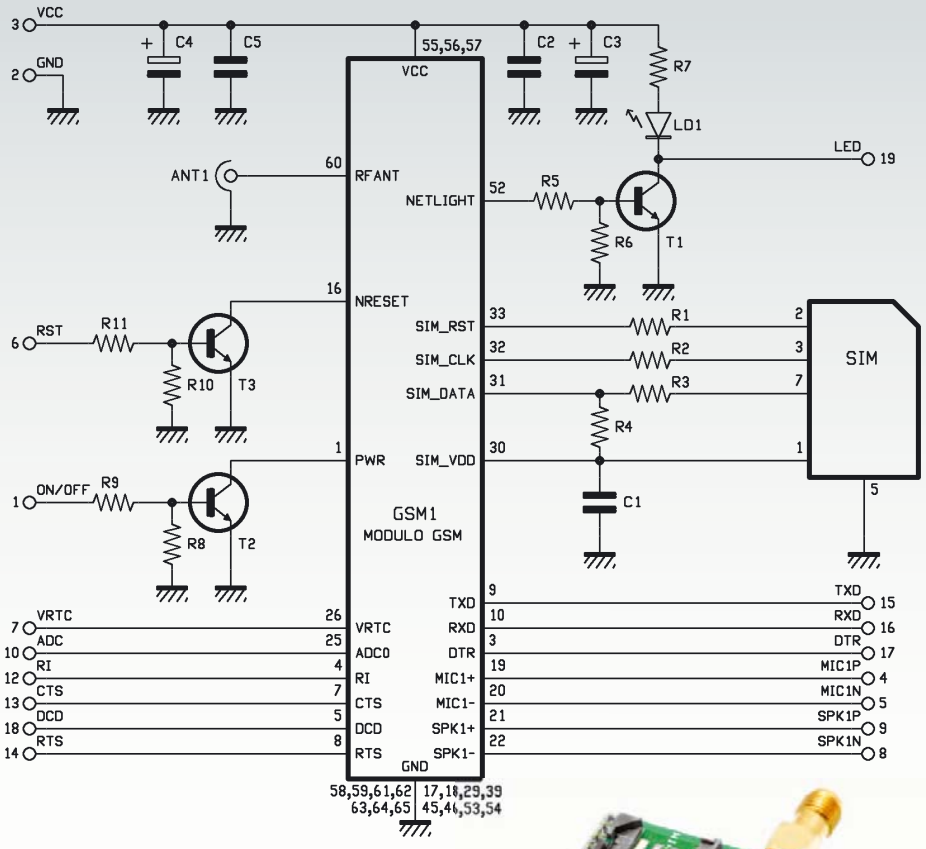


prescritto dal costruttore e specificato nel data-sheet. La separazione si effettua molto semplicemente facendo arrivare ai piedini di massa ed alimentazione analogica una pista che porta direttamente al negativo di alimentazione del circuito; piste distinte devono arrivare alle sezioni digitali. Nei punti da cui partono tali piste, si deve porre una coppia di condensatori di filtro (uno elettrolitico, ad esempio da 100 microfarad, ed uno ceramico da 100 nF); questo accorgimento fa in modo che se un disturbo si propaga sulla linea digitale non riesce ad arrivare ai pin dell'alimentazione della parte analogica. In verità l'ISD1700 prevede la separazione anche dell'alimentazione dello stadio di potenza, ossia l'amplificatore per l'altoparlante, quindi ha tre coppie di pin di alimentazione: Vcca e Vssa (rispettivamente positivo e negativo dello stadio analogico, cioè input microfonico e di linea, campionatori) Vccd e Vssd (per la sezione digitale, vale a dire l'interfaccia SPI e gli stadi di controllo della logica che fa capo ai pin di comando /REC, /PLAY, /ERASE ecc.) e, infine, Vccp e Vssp (rispettivamente positivo e negativo di alimentazione dell'amplificatore audio per l'altoparlante). Ognuna deve essere distinta dalle altre.
In questo riquadro vedete lo schema a blocchi interno dell'ISD1700.

veicola il MOSI (Master Output Slave Input, ossia l'ingresso dati dell'ISD1790PY) e il DCD trasporta i dati dal MISO (Master Input Slave Output) dell'ISD al microcontrollore. La connessione SPI è quella completa bidirezionale e permette al microcontrollore della scheda base di impartire i comandi e di leggere i dati in arrivo dall'ISD1790PY.
La gestione della memoria di registrazione dei messaggi è affidata in toto al microcontrollore, che la ripartisce in due zone principali: la prima, della durata di 40 secondi, contiene il menu dei comandi e delle opzioni proposto dal circuito quando lo si chiama; questa porzione non può essere modificata. Invece nella seconda possiamo registrare i messaggi personalizzati relativi alle risposte che il sistema darà sullo stato degli ingressi e sulla condizione delle uscite; ciò perché, mentre il menu è fisso, gli avvisi vocali devono essere adattati alle esigenze dell'installazione e comunque è giusto che ognuno possa personalizzarli nel modo che ritiene più efficace. L'unico limite è la durata assegnata ad ogni messaggio, che è limitata a 5 secondi.
L'audio in uscita dall'integrato viene reso disponibile tra i contatti SPK, dai quali può pilotare direttamente un piccolo altoparlante da usare come monitor durante la registrazione dei messaggi (per sentire come sono venuti); siccome lo stadio di potenza audio dell'ISD1790PY è a ponte, quindi i piedini SP+ ed SP- sono sollevati dalla massa, per trasferirne il segnale all'ingresso BF del modulo cellulare abbiamo dovuto realizzare la rete formata da R7÷R10, oltre che da C6 (quest'ultima filtra i segnali ad alta frequenza e previene le oscillazioni che il disadattamento delle impedenze possono causare). L'accoppiamento all'input audio del SIM900 è indispensabile per far ascoltare i messaggi durante le chiamate d'allarme ed il menu vocale. La registrazione dei messaggi si effettua direttamente con una capsula microfonica (MIC) collegata all'apposito ingresso differenziale dell'U1 (MIC+ e MIC-) e polarizzata dalla rete R2, R1, R3, C3.
L'ingresso audio di linea (LINE) non viene usato dal sistema, però può essere utile se intendete registrare messaggi o musica provenienti da una fonte esterna; ricordate che l'input di linea ha una sensibilità di poche centinaia di millivolt ed alta impedenza.
La durata di registrazione è stabilita dalla frequenza di clock dell'oscillatore interno all'ISD1790PY, che scandisce il campionamento; tale frequenza si definisce con il valore della resistenza R4 (collegata tra Rosc e massa).

SCHEDA GSM

Il c.s. che ospita il cellulare SIM900 si monta sopra tutti, ossia



Elenco Componenti:

C1: 220 nF multistrato (0805)
C2: 100 nF multistrato (0805)
C3: 470 μF 6,3 VL tantalio (CASE-D)
C4: 470 μF 6,3 VL tantalio (CASE-D)
C5: 100 nF multistrato (0805)
LD1: LED verde (0805)
R1: 15 ohm (0805)
R2: 15 ohm (0805)
R3: 15 ohm (0805)
R4: 10 kohm (0805)
R5: 4,7 kohm (0805)
R6: 10 kohm (0805)
R7: 330 ohm (0805)
R8: 10 kohm (0805)
R9: 4,7 kohm (0805)
R10: 10 kohm (0805)
R11: 4,7 kohm (0805)
T1, T2, T3: BC817
GSM1: Modulo GSM SIM900
SIM: Slot per SIM-CARD

Varie:
- Connettore antenna MMCX 90°
- Strip maschio 2x10 poli 2mm
- Circuito stampato



per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono facilmente reperibili in commercio mentre i master dei CS utilizzati sono scaricabili gratuitamente dal sito della rivista (www.elettronicain.it). Il progetto del combinatore (cod. TDG135) è anche disponibile, comprensivo di antenna, montato e collaudato al prezzo di 128,00 Euro. Sono disponibili separatamente l'apposito contenitore plastico già forato (BOXTDG135, Euro 16,90) e l'interfaccia USB per la programmazione tramite PC (FT782M, Euro 15,00). Non è compreso l'alimentatore. 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



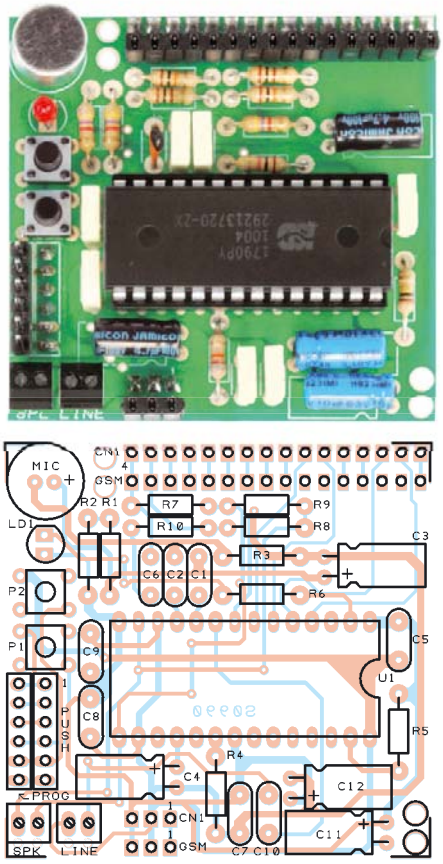
sulla scheda della sintesi vocale; allo scopo, dispone di due pin-strip per le connessioni con il circuito base del telecontrollo (che passano dal modulo vocale) uno da 3 pin ed uno da 16 pin: il primo trasporta positivo e negativo dell'alimentazione digitale, oltre alla linea di controllo dell'accensione (PWR), mentre il secondo contiene tutti i segnali e le linee di comunicazione da e verso il modulo GSM, ma anche la massa della sezione analogica dei cellulari (AGND, contatto 11). La linea PWR serve al microcontrollore della scheda base per comandare l'accensione e lo spegnimento del modulo GSM, che comunque rimane costantemente sotto tensione, fornita dalla linea Vcc ai piedini 55, 56, 57, ma viene acceso o spento mediante il livello logico applicato alla sua linea PWR (piedino 1 del SIM900). Quest'ultima linea è provvista di una resistenza di pull-up interna ed è attiva a zero logico, quindi il microcontrollore, per accendere il modulo cellulare, pone a livello logico alto PWR (contatto 1 del pin-strip) e manda in saturazione il transistor T2, il quale pone a livello basso la linea PWR del modulo SIMCom. Il reset del cellulare avviene facendo inviare dal microcontrollore l'1 logico sulla linea RST, allorché il transistor T3 va in saturazione e pone a livello basso la linea NRESET del SIM900.

In questa applicazione non abbiamo bisogno di gestire il reset, quindi la rete di T3 può non essere montata.
Procediamo ora con le linee di controllo dell'UART, ossia TXD, RXD, DTR, RTS, CTS, DCD, che vanno all'esterno mediante i contatti, rispettivamente, 15, 16, 14, 14 e 18 del pin-strip; lo stesso dicasi per VRTC (contatto 7) e ADC0 (10). Notate che in questa applicazione, della seriale usiamo solo i canali dati (TXD, RXD) e non i segnali di controllo RTS, CTS, DTR, DCD.
L'audio, facente capo a due contatti per il microfono (ad ingresso differenziale) ed altrettanti per l'altoparlante, transita dai contatti 4, 5, 8, 9, che corrispondono rispettivamente a MIC1P e MIC1N (positivo e negativo del microfono) ed SPK1N e SPK1P (rispettivamente negativo e positivo dell'altoparlante). Il segnale di RI (arrivo chiamata) esce dal contatto 12 del pin-strip.
L'antenna del modulo GSM, in questo caso si collega direttamente sullo stampato del cellulare: ciò evita tutti quei problemi che possono manifestarsi portando le connessioni d'antenna dallo stampato del cellulare a quello base (ad esempio disadattamenti di impedenza). La presa d'antenna è al piedino 60.
Il transistor T1 serve per comandare localmente il LED di campo del cellulare: la sua base viene polarizzata dal livello logico presente sul piedino 52 (NETLIGHT) del GSM1 e dal collettore parte la linea che porta al contatto 19 del pin-strip e che fa capo alla linea LED, con la quale il microcontrollore si informa sull'eventuale presenza della rete GSM e sullo stato di connessione del modulo (assenza segnale di rete, presenza rete ecc.).
Concludiamo l'analisi del circuito occupandoci delle linee di gestione della SIM, che si interfaccia tramite il bus formato da SIM_CLK (clock) SIM_RST (reset) e SIM_DATA (canale dati); la linea SIM_VDD viene usata per accendere e spegnere la SIM e viene gestita dal modulo GSM. Notate la presenza del quadruplo Zener D1, i cui diodi servono a proteggere la SIM-Card da eventuali picchi di tensione dovuti a disturbi, che possono arrivare dalla linea di alimentazione fino alle linee del bus di comunicazione tra SIM e cellulare; D1 può comunque non essere montato.

Elenco Componenti:

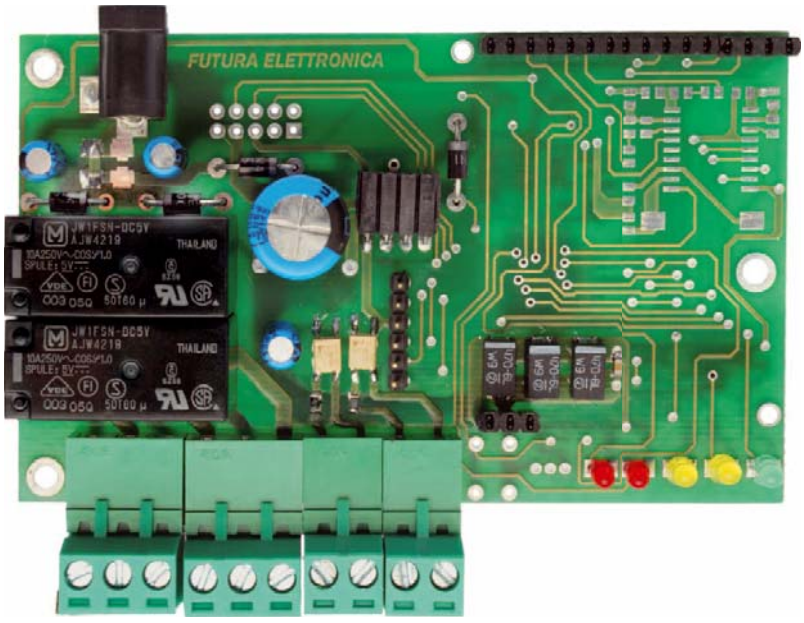
R1÷R3: 4,7 kohm
R4: 82 kohm
R5: 100 kohm
R6: 1 kohm
R7, R8: 10 kohm
R9, R10: 100 ohm
C1, C2: 100 nF 63 VL poliestere
C3, C4: 4,7 μF 100 VL elettrolitico
C5: 100 nF 63 VL poliestere
C6: 4,7 pF ceramico
C7÷C10: 100 nF 63 VL poliestere
C11, C12: 100 μF 63 VL elettrolitico
LD1: LED 3 mm rosso
U1: ISD1790PY
P1, P2: Microswitch
MIC: Microfono preamplificato

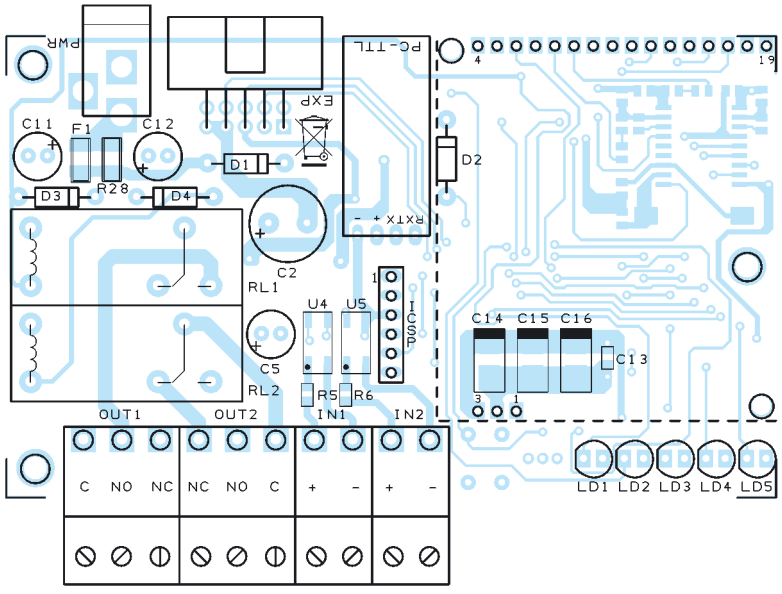
Varie:
- Morsetto 2 poli passo 2,54 mm (2 pz.)
- Zoccolo 14+14
- Strip maschio 3 poli
- Strip maschio 6 poli
- Strip maschio 16 poli
- Strip femmina 3 poli
- Strip femmina 6 poli
- Strip femmina 16 poli
- Circuito stampato



REALIZZAZIONE PRATICA

Il sistema consta di tre circuiti stampati, due dei quali realizzati in buona parte da componentistica SMD, quindi realizzarlo richiede un minimo di esperienza e una certa attrezzatura base, la quale consiste semplicemente in un saldatore o stazione saldante a punta finissima, da 20 W di potenza, oltre a del flussante, a una pinzetta e del filo di lega saldante da 0,5 mm di diametro. Non guasta avere a disposizione anche una lente d'ingrandimento, per ispezionare le saldature dei componenti più piccoli.
Dopo aver ottenuto le singole basette (trovate le tracce nel nostro sito web *www.elettronicain.it*) montate i componenti in ordine di altezza dando la precedenza agli SMD; per il montaggio del microcontrollore della scheda base, consigliamo innanzitutto di posizionare il componente ben centrato sulle piazzole corrispondenti e di fermarlo stagnandone leggermente un elettrodo per





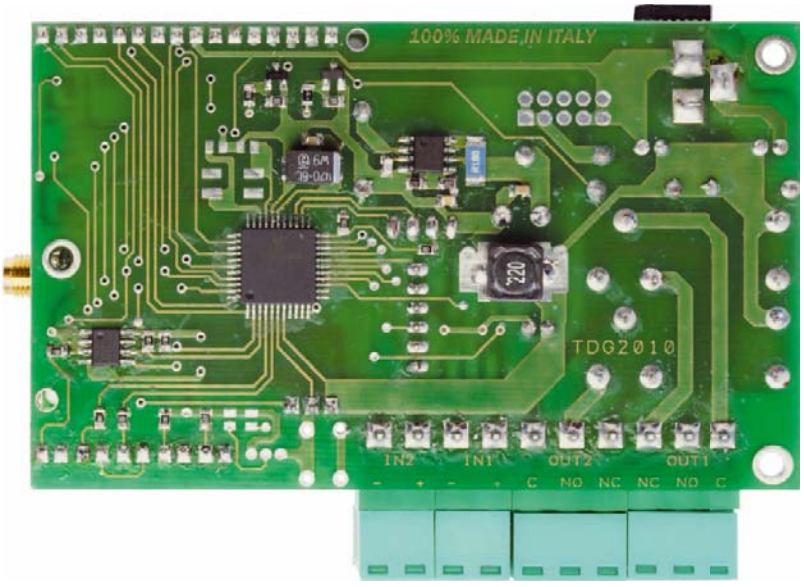
Elenco Componenti:

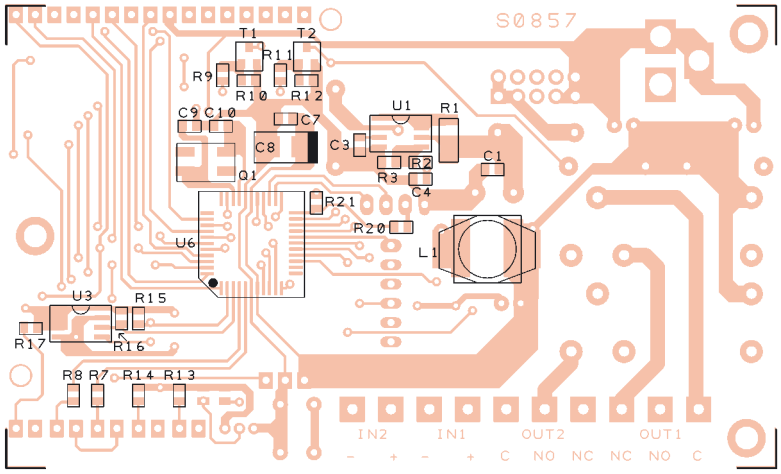
R1: 0,1 ohm 1W (1206)
R2: 2,2 kohm (0805)
R3: 1,2 kohm (0805)
R4: -
R5: 4,7 kohm (0805)
R6: 4,7 kohm (0805)
R7: 330 ohm (0805)
R8: 330 ohm (0805)
R9: 4,7 kohm (0805)
R10: 10 kohm (0805)
R11: 4,7 kohm (0805)
R12: 10 kohm (0805)
R13: 330 ohm (0805)
R14: 330 ohm (0805)
R15: 1,5 kohm (0805)
R16: 1,5 kohm (0805)
R17: 330 ohm (0805)
R18: -
R19: -
R20: 1 kohm (0805)
R21: 2,2 kohm (0805)
R22: -
R23: -
R24: -
R25: -
R26: -
R27: -
C1: 100 nF multistrato (0805)
C2: 1000 µF 35 VL elettrolitico
C3: 100 pF ceramico (0805)
C4: 100 nF multistrato (0805)
C5: 100 µF 16 VL elettrolitico
C6: -
C7: 100 nF multistrato (0805)
C8: 470 µF 6,3 VL tantalio (CASE-D)
C9: -
C10: -
C11: 100 µF 16 VL elettrolitico
C12: 100 µF 16 VL elettrolitico
C13: 100 nF multistrato (0805)
C14: 470 µF 6,3 VL tantalio (CASE-D)

lato; nel fare le saldature, raccomandiamo di prestare molta attenzione a non cortocircuitare, con la lega saldante, piedini vicini (ciò non è affatto da escludere in quanto il passo dei pin del micro è estremamente ridotto).
Va meglio con il modulo di sintesi vocale, giacché è interamente realizzato con componenti a montaggio tradizionale e quindi si monta assai semplicemente anche senza disporre di una particolare attrezzatura.
Il modulo per sintesi vocale richiede dei connettori S.I.L. femmina a passo 2,54 mm dotati di pin lunghi, per realizzare l'interconnessione tra la scheda base e il modulo cellulare; la scheda base necessita invece di semplici connettori S.I.L. da 2,54 mm per ospitare il blocco GSM/sintesi vocale, mentre il circuito del GSM ha bisogno di semplici pin-strip. Il modulo GSM è tutto a montaggio superficiale ed il suo componente più critico è indubbiamente il SIM900; quest'ultimo va posizionato sulle proprie piazzole e poi saldato facendo sciogliere dello stagno negli angoli formati dai suoi contatti con le piazzole del circuito stampato. La saldatura che si otterrà sarà più che sufficiente e duratura,

C15: 470 µF 6,3 VL tantalio (CASE-D)
C16: 470 µF 6,3 VL tantalio (CASE-D)
C17: -
C18: -
C19: -
C20: -
Q1: -
Q2: -
U1: MC34063AD
U2: -
U3: 24FC256-SN
U4: TLP181
U5: TLP181
U6: PIC18F46K20-I/PT (MF857)
U7: -
D1: 1N4007
D2: 1N5819
D3: 1N4007
D4: 1N4007
T1: BC817
T2: BC817
LD1: LED 3 mm rosso
LD2: LED 3 mm rosso
LD3: LED 3 mm giallo
LD4: LED 3 mm giallo
LD5: LED 3 mm verde
L1: Bobina 20 µH
RL1: Relé 5 V 1 scambio
RL2: Relé 5 V 1 scambio
P1: -
F1: Fusibile rapido 2 A (1206)

Varie:
- Morsettiera 2 poli (2 pz.)
- Morsettiera 3 poli (2 pz.)
- plug alimentazione
- Strip maschio 6 poli
- Strip femmina 3 poli
- Strip femmina 16 poli
- Strip femmina 4 poli 90°
- Circuito stampato





anche se lo stagno non penetra sotto i pad. Chi non vuole far da sè può acquistare il modulo col GSM già montato e collaudato: si trova da Futura Elettronica ed è siglato TDG_GSM900.
Completate le saldature di tutti i circuiti e accertato che non vi siano componenti polarizzati invertiti o piazzole unite accidentalmente, si può assemblare il sistema, disponendo il modulo di sintesi vocale sopra la scheda base e facendolo entrare a fondo nei connettori previsti allo scopo; montato tale modulo, inseritegli quello GSM, con analoga manovra. È fondamentale rispettare tale ordine di montaggio, altrimenti i collegamenti non si realizzano correttamente, in quanto il modulo per sintesi vocale può portare i segnali al GSM ma non è possibile viceversa. L'antenna del modulo si collega direttamente su di esso, mediante un cavetto terminante con un connettore SMA. Per alimentare il telecontrollo occorre un alimentatore o batteria capace di erogare da 5 a 30 volt e una corrente dell'ordine dei 500 mA.
Bene, per ora abbiamo concluso; nella prossima puntata vedremo come configurare ed usare il combinatore.

LA PRIMA RIVISTA ITALIANA PER LA COMUNITÀ LABVIEW

LabVIEW World



IN QUESTO NUMERO:

- Novità dal mondo LabVIEW
- Tutorial
- Tips & Techniques
- Interviste
- Applicazioni
- Scuola di LabVIEW
- Eventi, seminari...

e molto altro ancora

abbonati subito!

Rivista a cadenza trimestrale. Costo abbonamento annuale € 20,00.

Abbonamenti, numeri arretrati, download su *www.labviewworld.it*



Analizziamo il comportamento degli integrati di Linear Technology facendo ricorso alla versione del simulatore SPICE appositamente personalizzato da LT. In particolare, ecco un esempio con il dispositivo di protezione della tensione di alimentazione LTC4365.

LTSPICE IV TUTORIAL

dell'Ing. VINCENZO MENDOLA

Quanti di voi si sono cimentati nella progettazione di dispositivi elettronici analogici oppure hanno sentito la necessità di modificare circuiti esistenti per aggiungere o personalizzare alcune funzionalità, avranno quasi sicuramente utilizzato, almeno una volta, un simulatore circuitale. Questa tipologia di software permette, con grande precisione e velocità, di analizzare il funzionamento di un circuito elettronico e verificare come varia il suo comportamento al variare dei parametri circuitali, eliminando la lunga fase di sperimentazione che viene generalmente intrapresa attraverso l'uso di breadboard e millefori nel mondo hobbistico e di prototipi da parte dei professionisti. Nei sistemi più complessi, per quanto utile ed affidabile, il simulatore analogico non sempre può sostituirsi all'analisi strumentale ed alla realizzazione fisica della scheda elettronica, in quanto le caratteristiche del circuito stampato e la disposizione dei componenti sono parametri che non vengono considerati da questa tipologia di simulazioni; per studiare e analizzare questi effetti si utilizzano i simulatori elettromagnetici, solitamente non necessari per le applicazioni di cui ci occupiamo sulla rivista.
I programmi che consideriamo hanno come progenitore comune SPICE (Simulation Program with Integrated Circuit Emphasis), sviluppato all'università di Berkley, California, alla fine degli anni '60. Nel corso degli anni la complessità dei dispositivi elettronici è cresciuta e conseguentemente anche



Fig. 1



Fig. 2

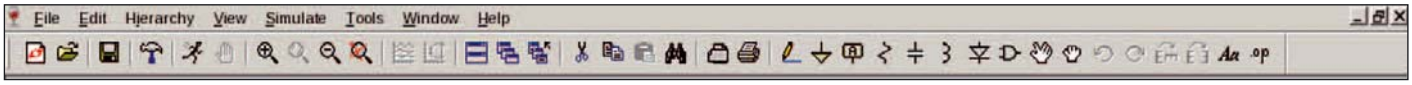


Fig. 3



Fig. 4

l'esigenza di disporre di simulazioni più veloci ed accurate; ciò ha determinato un continuo perfezionamento degli algoritmi, l'ampliamento delle librerie e l'aumento delle capacità complessive del codice che è avvenuto di pari passo con l'incremento delle prestazioni dei calcolatori elettronici. L'introduzione dell'interfaccia grafica ha semplificato notevolmente l'utilizzo di questi programmi, rendendo più immediato e intuitivo l'inserimento del circuito da studiare, eliminando al contempo la necessità di introdurre direttamente le istruzioni con cui si rappresentano gli elementi circuitali e le connessioni. Per completezza aggiungiamo che l'uso diretto di comandi SPICE è una feature che è stata mantenuta nella maggior parte delle attuali gradevolissime GUI (Graphical User Interface) per via delle potenzialità e della flessibilità offerte dai comandi diretti.

A partire dal codice di SPICE sono stati realizzati un'infinità di programmi derivati; il più famoso ed il primo ad essere in grado di essere eseguito sui personal computer è sicuramente PSPICE, nato negli anni '80.

Dopo questa breve introduzione, ci dedichiamo alla versione realizzata dalla Linear Technology: LTSPICE IV. Data la disponibilità di un cosi' gran numero di software c'è da chiedersi come mai LT abbia ritenuto necessario realizzare una nuova versione di SPICE piuttosto che servirsi di una delle tante già in commercio; la ragione sta nel fatto che i prodotti disponibili sono stati considerati troppo lenti per l'analisi dei nuovi alimentatori switching, caratterizzati da frequenze di commutazione notevolmente superiori rispetto a quelle utilizzate agli albori di questa tecnologia. Questo miglioramento del codice è stato ottenuto ottimizzando gli algoritmi di calcolo; un altro vantaggio introdotto è stato quello di permettere l'inserimento dei parametri parassiti direttamente nelle specifiche dei singoli componenti, come ad esempio la resistenza serie di un'induttanza, riducendo di conseguenza il numero di nodi e velocizzando al contempo la risposta, senza perdere in accuratezza.

Questo programma, ottimizzato

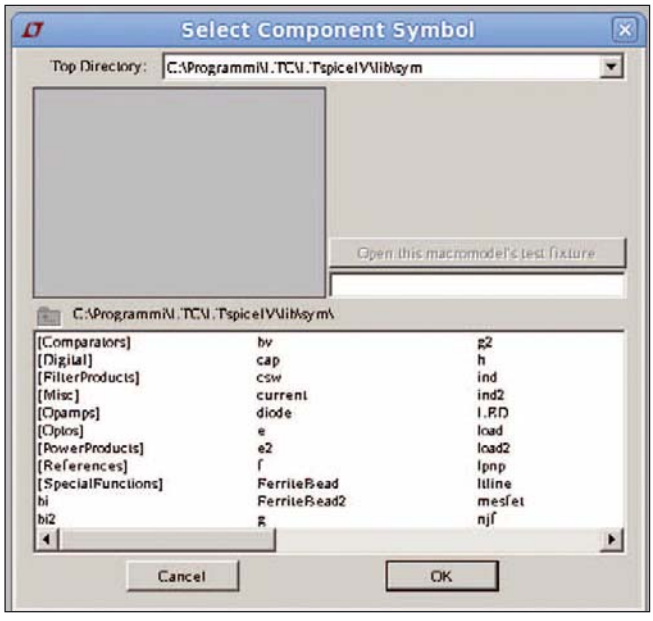


Fig. 5

per gli SMPS (Switching Mode Power System), può essere utilizzato per simulare qualunque circuito analogico, anche facendo ricorso a componenti generici o di altri produttori; ad ogni modo, per sfruttare a pieno le potenzialità offerte, è conveniente servirsi dei semiconduttori LT, per i quali sono disponibili modelli aggiornati frequentemente.
Veniamo ora a LTSPICE IV: innanzitutto scarichiamo gratuitamente dal link *http://ltspice.linear.com/software/LTspiceIV.exe* il file necessario all'installazione, che può essere eseguita sia su sistemi operativi Windows sia su Linux mediante WINE. Una volta installato, eseguiamo il programma: si aprirà l'interfaccia grafica (**Fig. 1**). Clicchiamo su "File" e "New Schematic" per realizzare un nuovo circuito. La stessa operazione può essere eseguita direttamente tramite l'icona "New Schematic" in alto a sinistra (**Fig. 2**).
Noteremo a questo punto che gran parte delle icone sotto la barra del menu, prima non selezionabili, diventano ora completamente visibili (**Fig. 3**).
Scegliamo "Component" rappresentato dal simbolo di una porta AND (**Fig. 4**): si aprirà la finestra "Select Component Symbol" (**Fig. 5**) dalla quale potremo scegliere i componenti che vogliamo inserire nella simulazione. Per introdurre direttamente una parte della LT di cui conosciamo il codice del produttore, è sufficiente digitarne la parte numerica. Supponiamo di voler utilizzare il circuito integrato "LTC4365" e di non ricordarci esattamente il suo "prefisso" (la LT generalmente nomina la gran parte dei suoi semiconduttori "LTXXXX" o "LTCXXXX"): inseriamo "4365" nel campo di ricerca; automaticamente verrà selezionato il componente desiderato (**Fig. 6**). Confermiamo con "OK": in questo modo avremo agganciato al cursore del mouse la parte scelta. Clicchiamo con il tasto sinistro del mouse e premiamo "ESC". Proseguiamo nella realizzazione dello schema aggiungendo gli altri elementi circuitali.
Un modo rapido per introdurre i componenti discreti è quello di servirsi delle icone disposte sulla sinistra di "Component" (**Fig. 7**): queste ci permettono di aggiungere diodi, induttanze, capacità e resistenze. L'icona (**Fig. 8**) serve per aggiungere il riferimento di massa mentre per collegare gli elementi circuitali ci serviamo di "Wire", rappresentato da una matita che traccia una linea (**Fig. 9**).
Come già evidenziato, il principale vantaggio di utilizzare LTSPICE IV con i semiconduttori LT consiste nella disponibilità di un gran numero di librerie e di circuiti "preconfezionati". Vediamo come: scegliamo, ad esempio, il circuito integrato LTC4365 precedentemente inserito nell'area di lavoro; spostiamo il cursore sul simbolo circuitale e selezioniamo, dopo aver cliccato con il tasto destro, "Open this macromodel's test fixture" (**Fig. 10**) operazione che determina l'apertura immediata del circuito di test, lo stesso che viene presentato nel datasheet relativo all'integrato scelto (**Fig. 11**). In questo caso tutti i parametri relativi alla simulazione sono già correttamente impostati per cui possiamo solo preoccuparci di analizzare i risultati. La stessa operazione può essere effettuata direttamente dalla finestra "Select Component Symbol" selezionando "Open this macromodel's test fixture". Eseguiamo la simulazione vera e propria tramite l'icona "Run" identificabile facilmente dal disegno stilizzato di una persona che corre (in inglese "run" significa appunto correre,

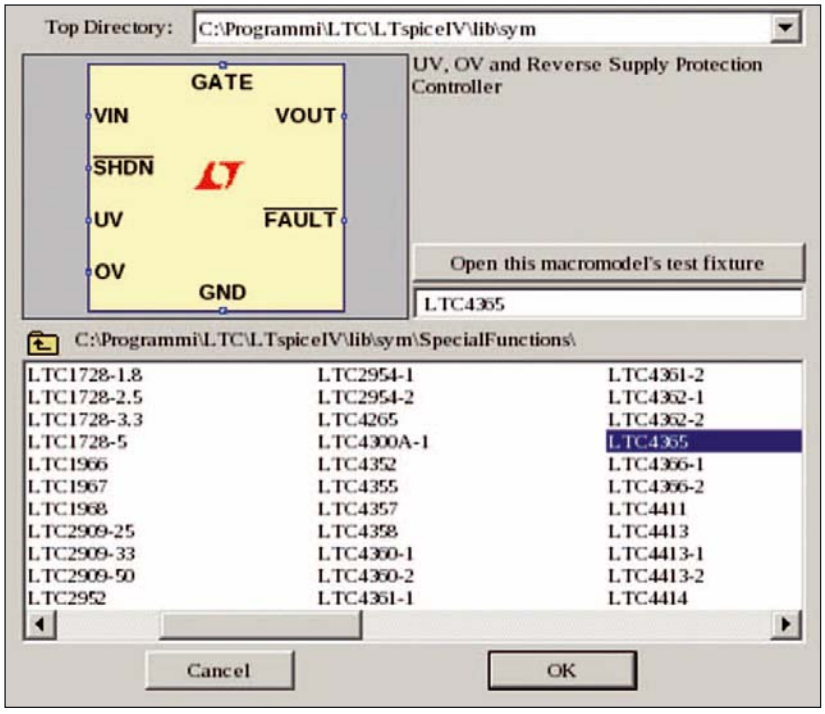


Fig. 6

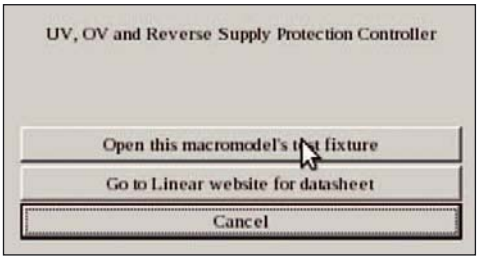


Fig. 10

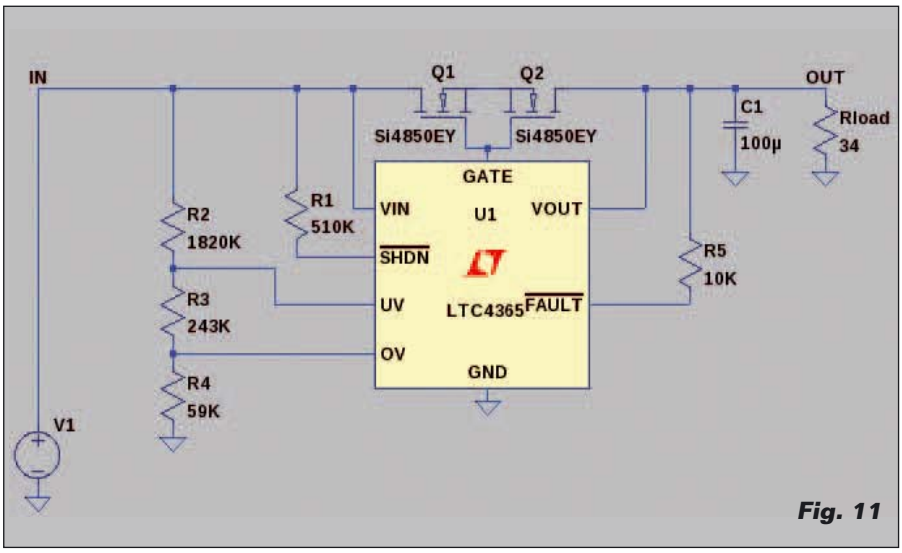


Fig. 11

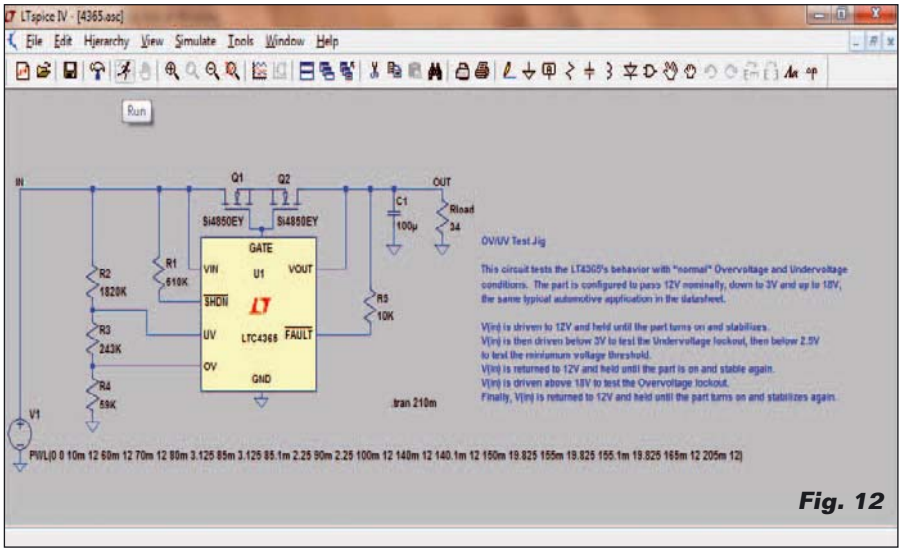


Fig. 12

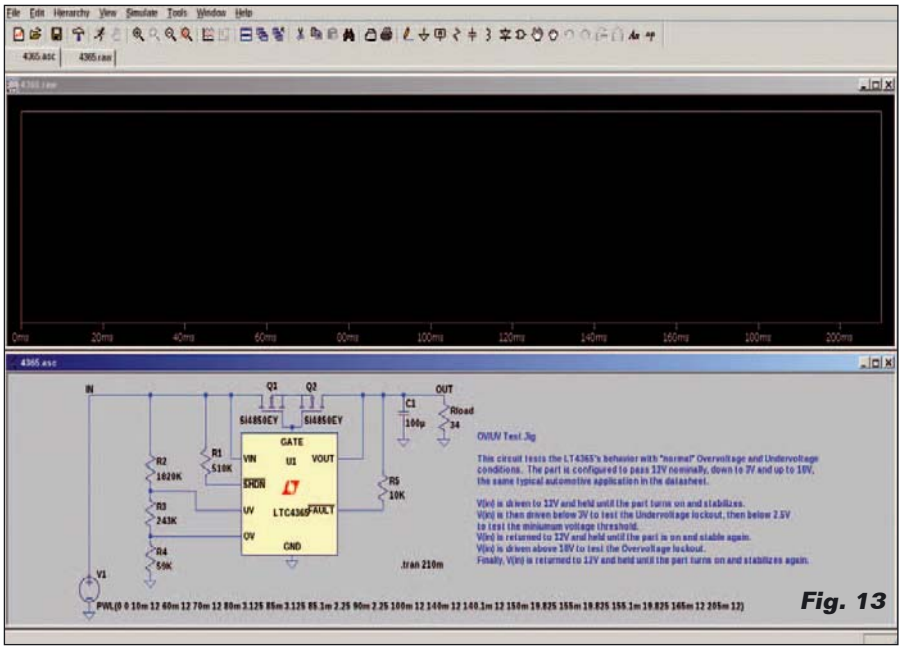
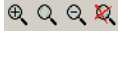


Fig. 13

eseguire) (**Fig. 12**). Si aprirà una nuova finestra con l'asse delle ascisse graduato in millisecondi e l'asse delle ordinate al momento senza alcuna grandezza (**Fig. 13**). Torniamo alla finestra con lo schema elettrico e avviciniamo il cursore all'etichetta "OUT"; il cursore si trasforma nel simbolo di un puntale di un multimetro. Clicchiamo e vedremo che sul grafico verrà rappresentato l'andamento della tensione d'uscita Vout in funzione del tempo (**Fig. 14**). Qualora volessimo analizzare l'andamento della corrente che attraversa il resistore di carico, spostiamo il cursore al di sopra del componente in oggetto: il cursore assumerà le sembianze di un amperometro e sul grafico verrà aggiunta anche tale grandezza (**Fig. 15**). Per comodità di visualizzazione separiamo le due grandezze tensione e corrente cliccando con il tasto destro in un area qualunque del grafico e selezionando "Add Plot Pane" (**Fig. 16** e **17**). Trasciniamo ora l'etichetta I(Rload) all'interno del nuovo grafico: automaticamente LTSPICE si occuperà di visualizzare separatamente le due grandezze da analizzare (**Fig. 18**). Come tutti i software di questo tipo è possibile ingrandire e selezionare aree a piacimento in modo da osservare nel dettaglio le forme d'onda  (**Fig. 19**).

Il simulatore della LT permette anche di mostrare, in maniera semplice e chiara, la potenza dissipata dagli elementi circuitali presenti nello schema; a tal proposito proviamo ad esempio ad aggiungere il grafico della potenza dissipata dal resistore "Rload": come nel caso precedente poniamo il cursore sul simbolo che rappresenta il nostro resistore e teniamo premuto il tasto "ALT" in modo da trasfor-

mare il cursore in un termometro. Confermiamo con il tasto sinistro, ottenendo quanto riportato nella **Fig. 20**.

Abbiamo mostrato come eseguire l'analisi della risposta al transitorio di un circuito (TRANSIENT) ma LTSPICE IV è in grado di effettuare tutte le più comuni tipologie di simulazioni analogiche tra cui AC SWEEP (analisi della risposta in frequenza), DC SWEEP (analisi del comportamento di un circuito al variare del valore DC di un generatore di tensione o corrente), NOISE (analisi del rumore).

Come potete intuire questa è solo una sintesi delle potenzialità offerte da LTSPICE IV che sono davvero tante e ben illustrate sia dalla documentazione tecnica ufficiale [1] [2] a cui consigliamo caldamente di fare riferimento per impadronirsi delle principali funzionalità del simulatore sia dall'help in linea accessibile direttamente dall'icona "Help" posta sulla barra dei menù.

Passiamo ora ad una possibile realizzazione pratica del circuito studiato precedentemente facente uso dell'integrato LTC4365 (**Fig. 21**): a questo è stato essenzialmente aggiunto un transistor PNP e un led rosso per segnalare l'intervento della protezione. Si sarebbe potuto utilizzare anche un diodo led per segnalare il regolare funzionamento del dispositivo oltrechè l'applicazione di una tensione al di fuori della finestra impostata ma non è stato ritenuto necessario in quanto per certe applicazioni è consigliabile ridurre al minimo i consumi; sono state invece aggiunte tutte le connessioni utili ad agevolare i collegamenti con il circuito da proteggere: partendo da sinistra della serigrafia al di sotto del connettore a pettine singolo in linea abbiamo "VIN", "SHDN"

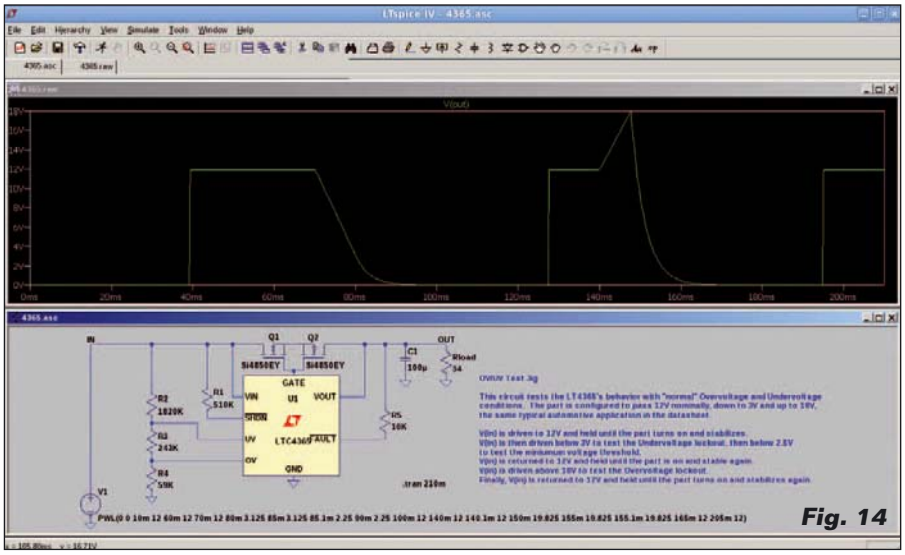


Fig. 14

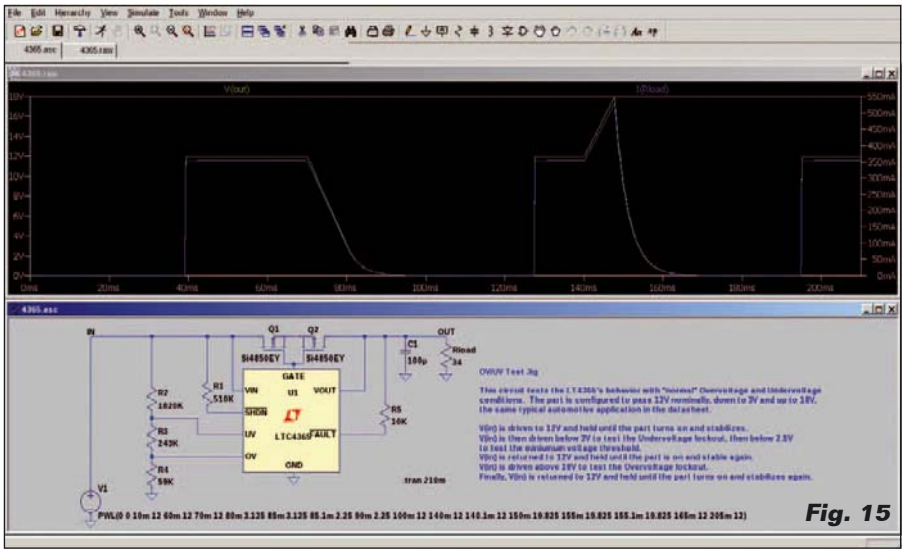


Fig. 15

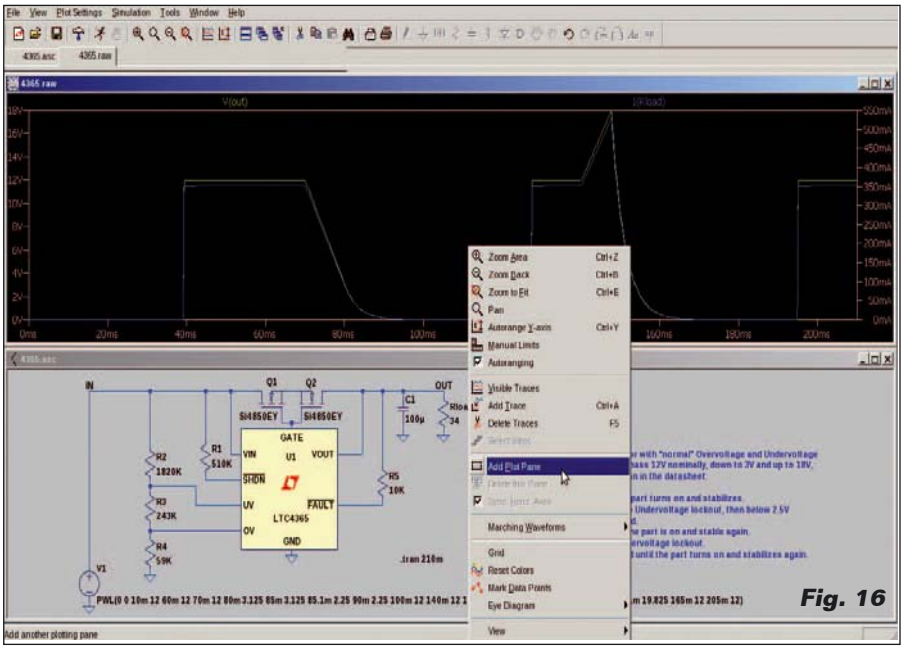


Fig. 16

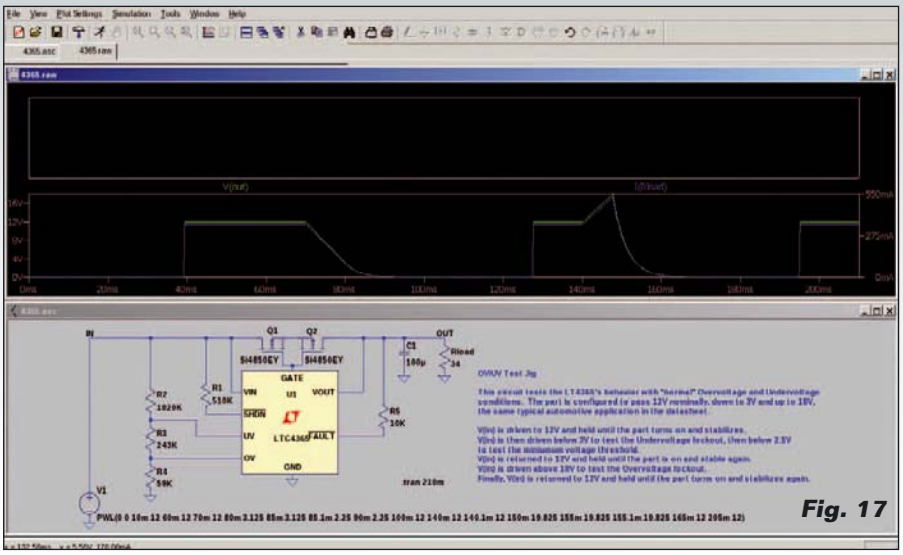


Fig. 17



Fig. 18

deriva dal fatto che generalmente sono più facilmente reperibili per l'hobbista, permettendo in maniera semplice di selezionare i valori necessari per impostare le soglie di protezione volute. Per chi volesse cimentarsi nella produzione di un proprio pcb, magari a singola faccia ed esclusivamente con componenti SMD, ricordiamo solo di prestare molta attenzione nella saldatura del LTC4365, le cui dimensioni sono estremamente ridotte (è un package TSOT-23 a 8 piedini) e richiedono l'utilizzo di un saldatore con la punta sottile, possibilmente conica, adatta a saldare componenti a montaggio superficiale di piccole dimensioni e soprattutto una buona manualità ed esperienza con questa tipologia di saldature.
Come si può osservare dall'analisi dello schema a blocchi del circuito integrato riportato nel datasheet [3], questo è costruito attorno a due comparatori con isteresi (**Fig. 22**), le cui soglie d'intervento sono fissate dalle tre resistenze collegate tra VIN, UV, OV e GND (indicate con R2,

e "MICRO" che permettono di selezionare se abilitare la protezione utilizzando un segnale esterno come quello proveniente da un micro o tenere sempre in funzione l'LTC4365; proseguendo abbiamo "GND", "FAULT", "OUT" e "VIN" il cui significato non necessita di ulteriori spiegazioni. Si è scelto di avere una connessione al pin "FAULT" dell'integrato per rendere disponibile anche all'esterno del circuito tale indicazione, per chi avesse la necessità di farne uso con un microcontrollore.
Il PCB è molto compatto e a doppia faccia, per ridurre ulteriormente le dimensioni. L'utilizzo di resistenze a foro passante

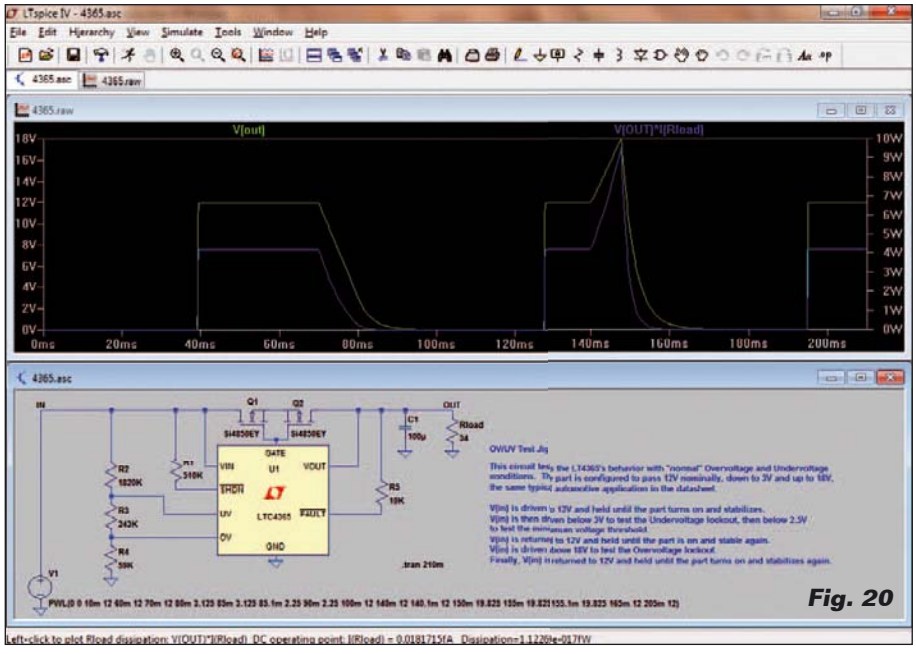


Fig. 20

Schema a blocchi LTC4365

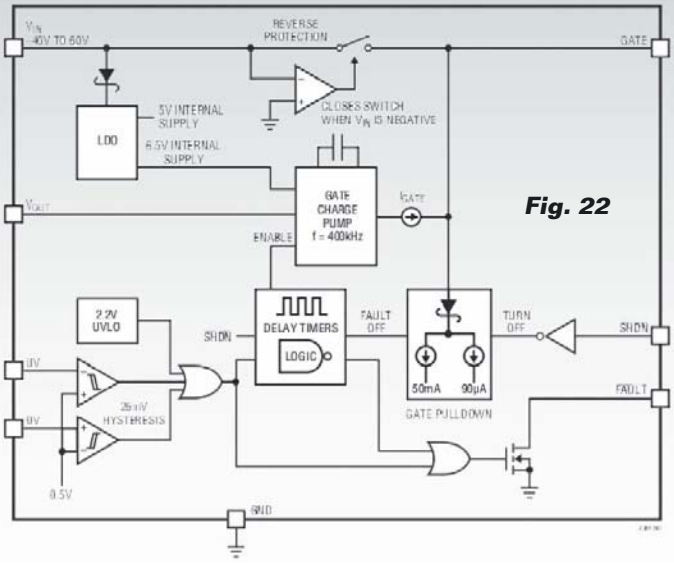


Fig. 22

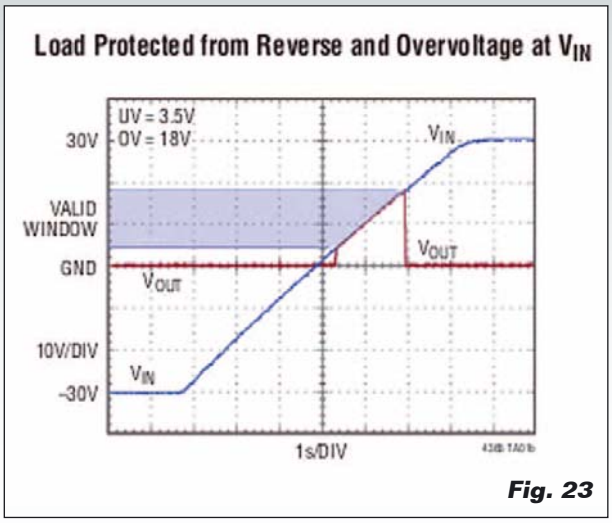


Fig. 23

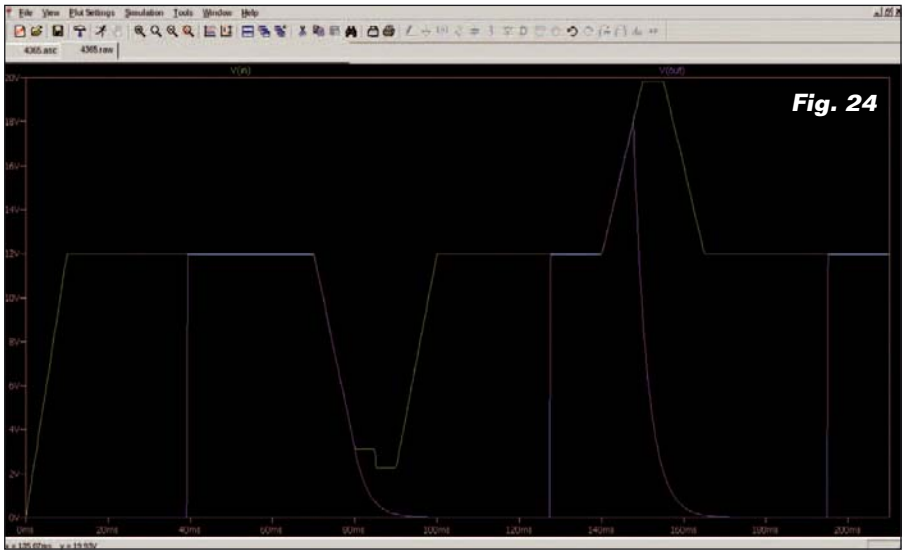


Fig. 24

R3, R4 in LTSPICE e R1, R2 ed R3 nel datasheet); i loro valori determinano la tensione sopra la quale (OV=OVERVOLTAGE) o al di sotto della quale (UV= UNDERVOLTAGE) il terminale "GATE" dell'integrato è a livello basso e quindi il mosfet in uscita è interdetto, come spiegato in dettaglio nella sezione "Application Information" del foglio dati del costruttore.
Il doppio MOSFET a canale N in configurazione back to back viene usato come protezione dalle inversioni di polarità, come evidenziato nel datasheet di cui è visibile un estratto che mostra un oscillogramma molto chiaro di come lavora il circuito (**Fig. 23**). È pertanto molto semplice ottenere, con una manciata di componenti, un sistema completo di protezione dalle sovratensioni, sottotensioni (particolarmente critiche per il corretto funzionamento di sistemi a microcontrollore) e inversioni di polarità, caratterizzato da tempi di intervento molto rapidi, come ben visibile dalla simulazione di **Fig. 24** in cui vengono confrontate su una scala temporale di 20ms a divisione, la tensione di uscita in funzione di quella d'ingresso, evidenziando le soglie d'intervento.
Una volta determinato il dispositivo da proteggere, LTSPICE può essere d'aiuto per verificare la correttezza dei valori scelti per i tre resistori posti sui due comparatori in modo da ottenere, in maniera semplice e compatta, un dispositivo che, realizzato a componenti discreti, sarebbe molto complesso e ingombrante.



Fig. 21

RIFERIMENTI

[1] *http://ltspice.linear.com/software/scad3.pdf*
[2] http://ltspice.linear.com/software/LTspiceGettingStartedGuide.pdf
[3] *cds.linear.com/docs/Datasheet/4365f.pdf*

made in Italy

Conosci bene ARDUINO?

Vuoi realizzare un telecontrollo GSM per gestire e monitorare a distanza impianti di riscaldamento/condizionamento o apparecchiature elettriche di vario genere?

La soluzione è...

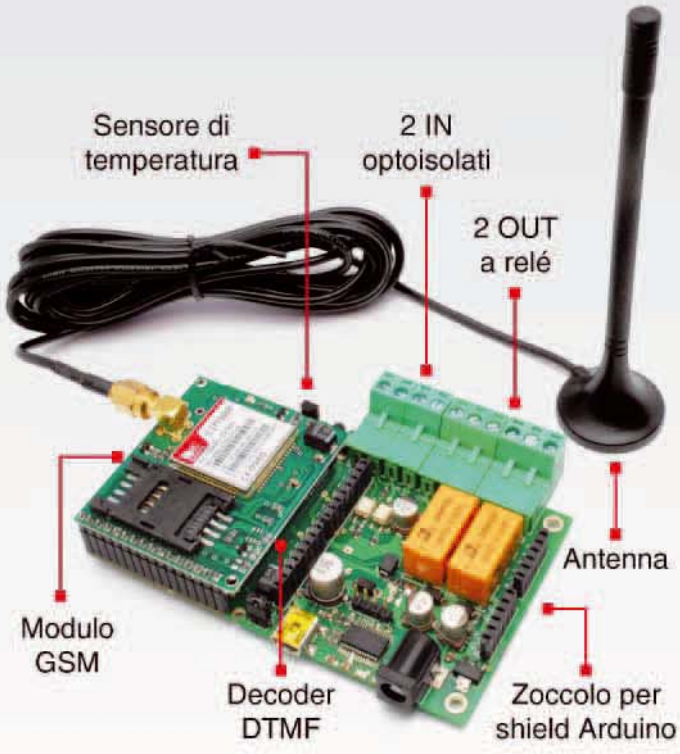
TiDiGino

la piattaforma hardware open-source, flessibile, semplice da utilizzare e alla portata di tutti.

Sviluppa il tuo controllo remoto basato su ARDUINO!

Con TiDiGino potrai ad esempio monitorare e impostare da remoto la temperatura ambiente di casa tua, gestendo l'impianto di riscaldamento/condizionamento, controllare pompe ed impianti di irrigazione, antifurti per immobili civili ed industriali, l'apertura di porte o cancelli, e molto altro ancora.

Lo zoccolo disponibile sulla scheda TiDiGino ti permette di utilizzare shield per Arduino progettati per svolgere specifiche funzioni utili ad ampliare le risorse hardware e le potenzialità applicative dell'unità.



Il telecontrollo GSM dispone di:

- ✓ Modulo GSM con SIM900 completo di antenna.
- ✓ Decoder DTMF.
- ✓ 2 uscite a relé.
- ✓ 2 ingressi optoisolati.
- ✓ Sensore di temperatura.
- ✓ Zoccolo per il montaggio di shield ARDUINO.
- ✓ Porta USB per programmazione (IDE ARDUINO).

Grazie alla disponibilità di librerie per la gestione della sezione GSM, ed alcuni esempi per testare le funzionalità hardware, potrai realizzare il tuo progetto ancora più facilmente.

Il telecontrollo GSM TiDiGino (cod. FT924M) viene fornito programmato con bootloader e sketch in grado di far svolgere al dispositivo tutte le funzioni disponibili. Informazioni tecniche dettagliate disponibili su ***www.futurashop.it****.*

€ 138,00 *cod.* **FT924M**

FUTURA ELETTRONICA

Via Adige, 11 • 21013 GALLARATE (VA)
Tel. 0331/799775 • Fax 0331/792287
www.futurashop.it

Prezzi IVA inclusa.

Disponibile presso i migliori negozi di elettronica o nel nostro punto vendita di Gallarate (VA). Caratteristiche tecniche e vendita on-line: *www.futurashop.it*

SHIELD UNIVERSALE PER MODULI GSM/GPRS/GPS



Appositamente studiato per i moduli della SIMCom, si adatta a diversi modelli grazie ad uno zoccolo polivalente e librerie specifiche. Ultima puntata.

di MARCO MARTINES

Una delle funzioni proficuamente implementabili in Arduino è la connettività GSM/GPRS, grazie alla quale si possono realizzare applicazioni come il controllo a distanza da cellulare, la connessione diretta tra due modem, il telecontrollo, l'accesso al web ed altro ancora. Convinti di questo, abbiamo presentato già due progetti di shield (ossia schede con specifiche funzionalità applicabili ad Arduino) che consentono la connessione telefonica cellulare; l'ultima l'avete vista nel fascicolo di aprile scorso, dove abbiamo presentato lo shield universale per moduli GSM/GPRS/GPS, accennando alla gestione ed al firmware. Ora è giunto il momento di concludere l'argomento descrivendo la libreria e la gestione del modulo SIM908 della SIMCom, equipaggiato con un ricevitore GPS integrato. La particolarità dello shield che vi proponiamo è che dispone di vari connettori per ospitare più moduli GSM/GPRS prodotti dalla SIMCom e montati su schede adattatrici come quelle che avete già visto in occasione della pubblicazione dei progetti dei telecontrolli della serie TDG1xx e nei

Tabella 1

char attachGPS()	Esegue la procedura per l'attivazione del GPS: attiva l'alimentazione ed imposta la modalità "hot start".	gps.attachGPS()
char deattachGPS()	Esegue la disattivazione, disabilitando l'alimentazione in modo da risparmiare energia.	gps.deattachGPS()

char getStat()	Controlla lo stato del GPS, i possibili output sono:
	0: GPS non avviato.
	1: Fix non eseguito (può essere causato da un segnale di scarsa qualità).
	2: Fix 2D eseguito.
	3: Fix 3D eseguito.
		char stat;
		stat=gps.getStat();
		if(stat==3)
		Serial.println("FIXED");

char getPar(**char*** str_long, **char*** str_lat, char* str_alt, **char*** str_time, **char*** speed);	Salva all'interno delle variabili passate i valori, rispettivamente, di longitudine, latitudine, altitudine, orario e velocità.	char lon[10]; char lat[10]; char alt[10]; char time[15]; char vel[10]; gps.getPar(lon,lat,alt,time,vel);

circuiti dei localizzatori basati su GPS o rete cellulare GSM. Oltre al popolare SIM900, il nostro nuovo shield per Arduino supporta il recentissimo SIM908, che ne è un'evoluzione e si propone di conquistare il mercato dei moduli GSM/GPRS quad-band fornendo svariate funzionalità aggiuntive che lo rendono unico nel suo genere, in special modo nel settore dei prodotti low-cost.
Il SIM908 si distingue per l'implementazione di un localizzatore GPS a 42 canali, caratterizzato da un'ottima accuratezza (arriva a raggiungere una precisione inferiore ai 3 metri) e da un ridottissimo tempo richiesto per effettuare il primo fix (1 secondo in modalità "hot start" e 30 secondi per quanto riguarda la modalità "cold start"). Una volta stabilita la connessione, il tempo di refresh, ossia di aggiornamento dei dati relativi alla posizione, è di 1 Hz, cioè un aggiornamento ogni secondo.
Questo gioiello, inoltre, è progettato per essere utilizzato alimentato con una batteria al litio, di cui è in grado di gestirne, in modo autonomo, il ciclo di carica, semplificando notevolmente tale processo che richiederebbe hardware e software dedicati.
Il SIM908 si gestisce assai facilmente, soprattutto se abbiamo una certa dimestichezza con il SIM900, in quanto pur integrando il ricevitore GPS, risulta pin-to-pin compatibile; le uniche differenze riguardano la gestione e l'interfacciamento, giacché nel SIM908 le due seriali vengono usate una per il GSM e l'altra per il GPS. Più esattamente, la prima seriale è provvista di un UART che fa capo alle linee TXD, RXD, DTR, che vanno all'esterno mediante i contatti, rispettivamente, 12, 14, 10 del connettore; per il GPS, invece, la seriale fa capo a GPSTXD (contatto 4) e GPSRXD (contatto 5). La prima seriale è in realtà destinata al controllo globale del SIM908, quindi da essa possiamo anche configurare il ricevitore GPS e domandargli di fornirci i dati sulla localizzazione, sul numero di satelliti agganciati e via di seguito, ovvero ricevere i dati richiesti e farli acquisire al microcontrollore della scheda base. Dalla seconda seriale (GPSTXD/GPSRXD) invece, escono continuamente le stringhe in formato NMEA standard del sistema GPS.
Se volete sapere di più sull'hardware, potete riferirvi alla prima puntata di questo articolo; nelle prossime righe troverete invece ciò che manca da spiegare sulla gestione e sul controllo mediante Arduino.

LA LIBRERIA PER IL SIM908

Prevedendo di usare anche il SIM908, la libreria per la gestione di tale modulo nello shield ha subito delle modifiche per fornire un rapido accesso a tutte le nuove funzioni messe a disposizione; la nuova libreria è derivata da quella utilizzata per il modulo SIM900 ed è reperibile in Internet al sito *http://code.google.com/p/gsm-shield-arduino/*. Per distinguerla dalla versione dedicata al SIM900, quella per il SIM908 è identificata con la sigla *GSM_GPRS_GPS*.
Notate che è possibile utilizzare la nuova libreria anche per gestire cellulari SIM900, a patto di non chiamare funzioni dedicate a questo nuovo modulo. Invece gli sketch destinati alla versione senza GPS sono compatibili con quella basata sul SIM908.
Vediamo qualche novità introdotta: innanzitutto, per quanto riguarda l'accensione è stata aggiunta la funzione f**orceON();** destinata a verificare lo stato del modulo e a forzarne l'accensione; questo perché con il supporto per la carica delle batterie a litio (vedi pagine seguenti), il modulo può essere avviato per eseguire la sola funzione di ricarica, escludendo



***Fig. 1** - Il ponte su J1 dell'FT971 per abilitare l'alimentazione all'antenna GPS.*

la gestione della rete GSM ma rispondendo ai principali comandi AT (un po' come quando colleghiamo il nostro cellulare spento alla presa di corrente: si attiva parzialmente solo al fine di gestire la carica). Se vogliamo scongiurare questa modalità ed accertarci che il modulo sia realmente acceso è dunque necessario chiamare la funzione **gsm.forceON();** la quale è destinata all'utilizzo del GPS (e della batteria).
È stata realizzata una classe contenuta all'interno dei file *gsm.cpp* e *gsm.h*, di cui è possibile istanziare un oggetto tramite **GPSGSM gps**, dopo aver incluso il relativo file *#include "gps.h"*, per poter richiamare le relative funzioni anteponendo il prefisso "gps." alla funzione desiderata.
Questa suddivisone in file differenti è stata realizzata per limitare al massimo l'utilizzo di RAM: difatti, ad esempio, tutte le variabili impiegate dalla classe relativa al GPS non saranno allocate in memoria se non verrà incluso il relativo file tramite **#include "gps.h"**. Ciò permette di scegliere, in base all'utilizzo dello sketch, quali variabili utilizzare. Come già accennato, anche per la gestione della batteria sono presenti delle funzioni che consentono di misurare la tensione e la temperatura delle celle; per motivi pratici, occupando poca memoria queste sono state incluse nella classe del GPS.
Dunque, per utilizzarle (come visto in precedenza per il GPS) dopo aver incluso il file *#include "gps.h"* bisogna istanziare l'oggetto corrispondente usando **GPSGSM gps**.
Nei prossimi paragrafi verranno mostrate le funzioni per il controllo del GPS e della batteria.

GESTIONE DEL GPS

Prima di utilizzare il GPS, è necessario effettuare un piccolo setup: per prima cosa, con una goccia di stagno andiamo ad effettuare un ponte sul jumper J1 presente sul modulo FT971 (vedere **Fig. 1**). Questo serve per portare l'alimentazione all'antenna del GPS. Successivamente, carichiamo lo sketch di esempio (all'interno della cartella examples) chiamato *GSM_GPRS_GPS_Library_AT* (o anche *GSM_GPRSLibrary_AT*) e una volta lanciato e completata l'inizializzazione impartiamo i seguenti comandi:

AT+CGPSPWR=1
AT+CGSPRST=0

Attendiamo un minuto e a questo punto il GPS dovrebbe essere funzionante; per verificarlo continuiamo impartendo il comando:

AT+CGPSINF=0

Se ci vengono restituite le coordinate, vuol dire che tutto è funzionante e possiamo procedere con l'utilizzo standard tramite le funzioni implementate, di cui la **Tabella 1** fornisce una breve panoramica.
Ora procediamo con un semplice esempio che ci permette di capire come ottenere le coordinate del sistema GPS dal modulo SIM908 montato sullo shield; il firmware corrispondente è descritto nel **Listato 1**.

Utilizzare lo shield con Arduino Mega

Per i problemi legati alla RAM, oppure semplicemente per progetti che richiedono un maggior numero di input/output, la pur ottima Arduino Uno potrebbe non risultare più sufficiente, spingendoci all'utilizzo di un prodotto più evoluto come Arduino Mega; sfortunatamente quest'ultima ha un po' di problemi con la seriale software che non è pienamente implementata. Ma non tutto è perduto, in quanto la Mega dispone di ben quattro seriali hardware, quindi possiamo sfruttare una di queste al posto della seriale software per comunicare con lo shield.
Con l'ultimo rilascio, la libreria è stata predisposta per essere utilizzata completamente anche con Arduino Mega; per poterne usufruire, è necessario aprire il *file GSM.h* e selezionare la scheda utilizzata commentando opportunamente le linee di codice.
Utilizzando lo shield con Arduino Mega dovremo commentare nel seguente modo:

```
//#define UNO
#define MEGA
```

viceversa, se volessimo utilizzare Arduino uno:

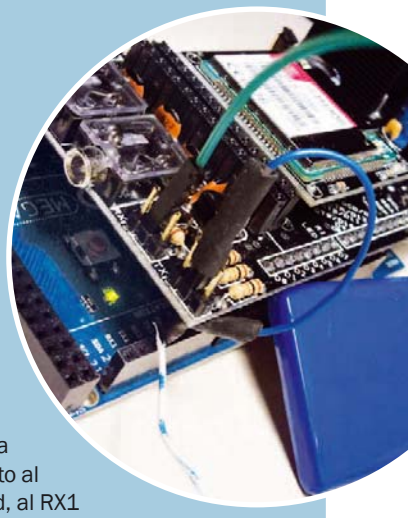
```
#define UNO
//#define MEGA
```

Analogamente, anche il file *HWSerial.h*, dovrà essere configurato. Come in precedenza, vediamo prima l'esempio per Arduino Mega:

```
#define MEGA
```

e per Arduino Uno:

```
//#define MEGA
```



All'interno del file *HWSerial.h* non è necessario definire l'eventuale utilizzo con Arduino Uno, in quanto la classe implementata in esso viene utilizzata solo dalla seriale hardware.
La libreria utilizza la seriale Serial0 (RX 0 TX 1) per le comunicazioni esterne mentre la seriale Serial1 (RX 18 TX 19) per comunicare con i SIM900 e SIM908.
Nulla vieta di sostituire ogni ricorrenza di **Serial1** con **Serial2** o quella che si preferisce.
Si tenga presente che per utilizzare la seriale hardware bisogna creare il collegamento tra shield e Arduino Mega utilizzando un ponticello: il pin TX dello shield va collegato al TX1 18 di Arduino Mega ed il pin RX, sempre dello shield, al RX1 19 (vedere l'immagine in questo riquadro).

Listato 1

```
#include “SIM900.h”
#include <SoftwareSerial.h>

#include “gps.h” //includiamo il file per la gestione di GPS e batteria

GPSGSM gps;               //istanziamo un oggetto relativo alla classe GPS

char lon[10];             //longitudine
char lat[10];             //latitudine
char alt[10];             //altitudine
char time[15];            //tempo in formato yyyymmddHHMMSS
char vel[10];             //velocità
char stat;                //variabile per memorizzare lo stato del GPS
boolean started=false;

void setup()
{
  //Serial connection.
  Serial.begin(9600);
  Serial.println(“GSM GPRS GPS Shield”);
  if (gsm.begin(2400)){
    Serial.println(“\nstatus=READY”);
    gsm.forceON();        //assicura che il SIM908 non sia in modalità rica-
rica
    started=true;
  }
  else Serial.println(“\nstatus=IDLE”);

  if(started){
    if (gps.attachGPS())  //avvia il GPS
      Serial.println(“status=GPSON”);
    else Serial.println(“status=ERROR”);

    delay(20000);         //tempo richiesto per agganciare i satelliti
    stat=gps.getStat();   //salva lo stato del GPS
        if(stat==1)
                Serial.println(“NOT FIXED”);
        else if(stat==0)
                Serial.println(“GPS OFF”);
        else if(stat==2)
                Serial.println(“2D FIXED”);
        else if(stat==3)
                Serial.println(“3D FIXED”);
        delay(5000);
                //salviamo tutti i dati relativi al GPS per stamparli
                //successivamente
        gps.getPar(lon,lat,alt,time,vel);
        Serial.println(lon);
        Serial.println(lat);
        Serial.println(alt);
        Serial.println(time);
        Serial.println(vel);
  }
};

void loop()
{
};
```

USARE E GESTIRE LA BATTERIA

Se vogliamo utilizzare la batteria al litio come fonte di alimentazione per il nostro modulo FT971 che ospita il SIM908 (prestate attenzione al fatto che il SIM900 non è in grado di gestire la carica, dunque è sconsigliato l'utilizzo della batteria con esso, in quanto non sareste in grado di ricaricarla) è sufficiente chiudere il jumper presente sullo shield denominato CHRG e, ovviamente, impostare nella posizione VEXT il secondo ponticello vicino al connettore della batteria. Una volta terminato il ciclo di carica, potrete riaprire il ponticello CHRG e settare nuovamente lo shield su BAT.
Notate che alcune applicazioni potrebbero richiedere l'alimentazione solamente per aggiornare il contatore destinato alla misura del tempo; in tal caso l'impiego di una batteria al litio potrebbe non essere necessario.
Comunque ricordate che sia il SIM900, sia il SIM908, possono essere dotati di un circuito che sfrutta la carica accumulata in un condensatore (di dimensione adeguata) per aggiornare l'RTC (Real Time Clock). Tramite le due funzioni di libreria è possibile ottenere la percentuale di carica residua, la tensione a cui viene erogata la corrente e la tensione letta dal sensore di temperatura. In caso di applicazioni poco ventilate, con tempi di lavoro prolungati e in condizioni climatiche non proprio ottime, è consigliato monitorare tale valore per assicurarsi che la batteria lavori entro i limiti che configurano un corretto funzionamento.
La temperatura può essere calcolata in base alla relazione tensione/temperatura del sensore, considerando che il legame è inversamente proporzionale, intendendo con ciò che a temperature maggiori corrisponderanno tensioni minori.
Inoltre è possibile settare il modulo in modo che si spenga automaticamente se la batteria sta lavorando al di fuori dei range consentiti; per attivare questa modalità è sufficiente inviare (la cosa si può fare tramite lo sketch di esempio contenuto all'interno della cartella *examples*, dedicato alla comunicazione tramite comandi AT) il comando seguente:

AT+CMTE=1

Viceversa, per disabilitarlo bisogna impartire il comando:

AT+CMTE=0

Mentre per conoscere in che modalità è configurato il modulo occorre impartire il comando:

AT+CMTE?

Per conoscere la corretta sintassi e i relativi return delle funzioni fate riferimento alla **Tabella 2**. Anche in questo caso vediamo come implementare in uno sketch queste funzioni, riferendoci al **Listato 2** che contiene il codice corrispondente.

Tabella 2

char getBattInf(**char*** str_perc, **char*** str_vol);	Salva, all'interno delle variabili passate, la batteria residua, in termini percentuali, e il voltaggio di batteria.	char str_perc[5]; char str_vol[6]; gps.getBattInf(str_perc,str_vol); Serial.print("CARICA: "); Serial.print(str_perc); Serial.println("%");
char getBattTVol(**char*** str_vol);	Salva all'interno delle variabili passate i valori, rispettivamente, di longitudine, latitudine, altitudine, orario e velocità.	char lon[10]; char lat[10]; char alt[10]; char time[15]; char vel[10]; gps.getPar(lon,lat,alt,time,vel);

MODALITÀ DEBUG

Durante l'utilizzo degli shield, a volte capita di non riuscire ad ottenere i risultati voluti senza comprenderne il motivo; per aiutarvi, le librerie (sia quella destinata al SIM900 che quella per il SIM908) sono configurabili per fornire in output alcuni messaggi di debug durante l'esecuzione delle funzioni chiamate.

All'interno del file *GSM.h* infatti è presente la seguente riga:

//#define DEBUG_ON

Decommentandola, si andrà ad

Listato 2

```
#include "SIM900.h"
#include <SoftwareSerial.h>
#include "inetGSM.h"
#include "gps.h" //includiamo il file per la gestione di GPS e batteria

GPSGSM gps;      //istanziamo un oggetto per utilizzarne le funzioni

char perc[5];    //stringa in cui salviamo la percentuale (tre cifre più due caratteri di 'a capo': "\n\r")
char volt[6];    //stringa in cui salviamo il voltaggio di batteria (quattro cifre più due caratteri di 'a capo': "\n\r")
char tvolt[6];   //stringa in cui salviamo il voltaggio del sensore di temperatura (quattro cifre più due caratteri di 'a capo': "\n\r")

long prevmillis=millis(); //variabile per la temporizzazione
int interval=10000;       //intervallo per l'esecuzione della routine

void setup()
{
  Serial.begin(9600);
  Serial.println("GSM GPRS GPS Shield.");
  if (gsm.begin(4800)){
    Serial.println("\nstatus=READY");
    gsm.forceON();        //assicura che il SIM908 non sia in modalità ricarica
  }
  else Serial.println("\nstatus=IDLE");
};

void loop()
{
  if(millis()-prevmillis>interval){        //controlla la temporizzazione relativa alla routine
    gps.getBattInf(perc,volt);             //acquisisce percentuale e tensione di batteria
    gps.getBattTVol(tvolt);                //acquisisce tensione del sensore di temperatura
    //carica batteria
    Serial.print("Carica batteria: ");
    Serial.print(perc);
    Serial.println("%");
    //tensione batteria
    Serial.print("Tensione batteria: ");
    Serial.print(volt);
    Serial.println(" mV");
    //tensione sensore temperatura
    Serial.print("Tensione sensore temperatura: ");
    Serial.print(tvolt);
    Serial.println(" mV");
    Serial.println("");
    prevmillis=millis();
  }
}
```

Carica delle batterie al litio

Le batterie al litio si suddividono in due categorie: quelle realizzate con ioni di litio (Li-Ion) e quelle formate dai polimeri di litio (Li-Po). Le prime si caratterizzano per avere un buon ciclo di vita, basso effetto memoria e soprattutto un'alta densità energetica; proprio per questo risultano molto diffuse nei dispositivi mobili come laptop, smartphone e cellulari, compreso anche il SIM908. La carica è leggermente più complicata rispetto a quella della classica batteria al piombo: la corrente di carica, infatti, deve essere inferiore alla massima supportata, indicata nelle specifiche tecniche, altrimenti le prestazioni della batteria in fatto di capacità potrebbero essere pregiudicate irrimediabilmente.
Analogamente, la tensione deve essere limitata (dinamicamente), secondo quanto indicato nelle specifiche, altrimenti oltre ad avere un degrado della capacità, si potrebbe andare incontro al surriscaldamento o, in alcuni casi, all'esplosione.
Risulta dunque ovvio che il caricabatterie dovrà essere in grado di controllare la corrente massima e al contempo regolare la tensione di alimentazione.
Nelle batterie al litio, la prima fase della carica si effettua a corrente costante (il limite corrisponde alla massima corrente di carica) fin quando la tensione non sale oltre i 3,7 volt, poi si procede a tensione costante, tendendo al valore di 4,2 V e limitando la corrente. Man mano che ci si avvicina alla carica completa, la corrente erogata dal caricabatterie viene diminuita, fino a diventare nulla.
Tutto questo è stato implementato nel modulo SIM908, il quale è in grado di gestire (quasi) autonomamente il ciclo di carica di un accumulatore a ioni di litio. Nel data-sheet del SIM908 sono elencate le caratteristiche consigliate per il pacco batterie:

- massima tensione di ricarica del pacco batterie di 4,2 V e capacità consigliata di 1.100 mAh (batterie con capacità superiore richiederanno tempi di carica più lunghi);
- il pacco batterie deve essere dotato di un circuito di protezione da ricariche con correnti e/o tensioni sopra il limite;
- la resistenza interna del pacco batterie, includendo il circuito di sicurezza sopracitato, dovrebbe essere il più possibile bassa (valori inferiori ai 150 mohm);
- la batteria dovrebbe essere protetta dall'inversione di polarità.

abilitare la modalità di output dei messaggi; viceversa, commentandola nessun messaggio di diagnostica verrà mostrato in output sulla seriale.
Va comunque notato che, a meno di non poterne fare a meno, è preferibile non abilitare tali messaggi, perché non farlo permette di risparmiare spazio in RAM.
Nelle prossime versioni della libreria software, questi messaggi saranno spostati nella EEPROM, così da non consumare le risorse di RAM; allora non dovremo più preoccuparci di escluderli.

UTILIZZO DELLA RAM IN ARDUINO UNO

Il problema principale in cui ci si imbatte durante l'utilizzo dello shield con Arduino Uno risiede nella saturazione della memoria RAM a disposizione. Infatti, quest'ultima è destinata a contenere tutte le variabili utilizzate dal nostro sketch, siano queste interi, numeri a virgola mobile, stringhe e quant'altro; occuparla inutilmente limita l'esecuzione dello sketch. Una piccola parentesi: non confondete la RAM con la memoria disponibile segnalataci dall'IDE durante la compilazione. Ad esempio se vediamo:

Binary sketch size: 2464 bytes (of a 32256 byte maximum)

il massimo numero di byte si riferisce alla dimensione massima del programma, inteso come numero di istruzioni impiegate nello sketch e non come spazio allocato per l'utilizzo delle variabili. Notate che i comandi:

```
int buf1[9999];
int buf2[9999];
int buf3[9999];
int buf4[9999];
int buf5[9999];
```

sebbene siano poche e semplici dichiarazioni di variabili ed occupino pochi byte, in realtà saturano completamente la memoria dell'Arduino Uno, limitata a 2 kB. Purtroppo non vi è alcun LED o messaggio che ci informa dell'esaurimento della memoria, condizione che si manifesta (fastidiosamente) con comportamenti anomali quali riavvii, nessun segno di vita o strani caratteri in output sulla seriale.
Una prima soluzione al problema è spostare tutte le stringhe, il cui contenuto è costante, all'interno della memoria EEPROM facilmente accessibile in lettura dallo sketch, meno in scrittura.
La seconda soluzione invece è utilizzare la scheda Arduino Mega, che è dotata di ben 8 kB di RAM. ■



per il MATERIALE

Lo shield descritto in queste pagine è disponibile in scatola di montaggio al prezzo di 19,50 Euro (codice 8190-GSMGPRSSHIELDV2). I moduli GSM, GPRS e GPS utilizzati con questa base sono invece disponibili già montati e collaudati.
a) modulo cellulare standard GSM/GPRS con SIM900 (cod. 8100-TDGG-SM_900), 51,00 Euro compresa antenna;
b) modulo cellulare GSM/GPRS miniatura (cod. 7100-FT900M) 49,00 Euro, antenna esclusa;
c) modulo miniatura GSM/GPRS/GPS con SIM908 (cod. 7100-FT971) 65,00 Euro, antenne escluse.
Questo shield va montato sulla board Arduino UNO che costa 24,50 Euro (cod. 7300-ARDUINOUNOREV3). 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

www.koellekunter.de

HAM RADIO
FRIEDRICHSHAFEN

Il n°1 d'Europa!

37. Esposizione Internazionale per Radioamatori

22.–24.6.2012

Messe Friedrichshafen/Germania

- 63° meeting DARC sul Lago di Costanza
- Il più grande mercato delle pulci HAM d'Europa
- Il meglio offerto da radiotecnica, elettronica e CB
- Incontro top dei radioamatori d'Europa

www.hamradio-friedrichshafen.de

MESSE FRIEDRICHSHAFEN

DARC
Deutscher Amateur-Radio-Club e.V.



SOLUZIONI PER LA TUA AUTO...

Mantieni sempre in perfetta efficienza il tuo veicolo verificando le condizioni dei principali componenti e sistemi con questi validi ed affidabili strumenti.

... PER LA DIAGNOSI

TESTER EOBD
Sistema completo, in valigetta

composto da:

- Interfaccia hardware montata
- Software di gestione Unicarscan-Scanmaster
- Cavo di connessione USB

cod. OBDSYSTEM
€ 149,00



Prezzi IVA inclusa.

... PER LA MESSA A PUNTO

TACHIMETRO DIGITALE CON PUNTATORE LASER

Permette di misurare, senza contatto, la velocità angolare di alberi motore, ruote dentate e pulegge fino ad un massimo di 99999 rpm, con accuratezza dello 0,05%. Dimensioni 160 x 58 x 39mm.



cod. DTO6234N
€ 89,00

cod. DVM205AM
€ 61,00



MULTIMETRO DIGITALE A 4 CIFRE PER AUTOVEICOLI

È in grado di misurare il numero di giri/minuto di un motore, l'angolo Dwell, tensione/corrente alternata e continua, frequenza, resistenza, capacità, diodi e continuità. Completo di batteria, puntali, cavetti con coccodrilli e termocoppia.



... PER L'ISPEZIONE

SNAKE CAMERA CON MONITOR LCD

Sistema per videoispezione composto da una telecamera CMOS a colori con tubo orientabile da 84,5cm e monitor LCD TFT a colori da 2,36". Angolo di ripresa 50°; illuminatore a LED con intensità regolabile; dimensioni telecamera Ø17x79mm.



cod. CAMCOLI3
€ 120,00

FUTURA ELETTRONICA

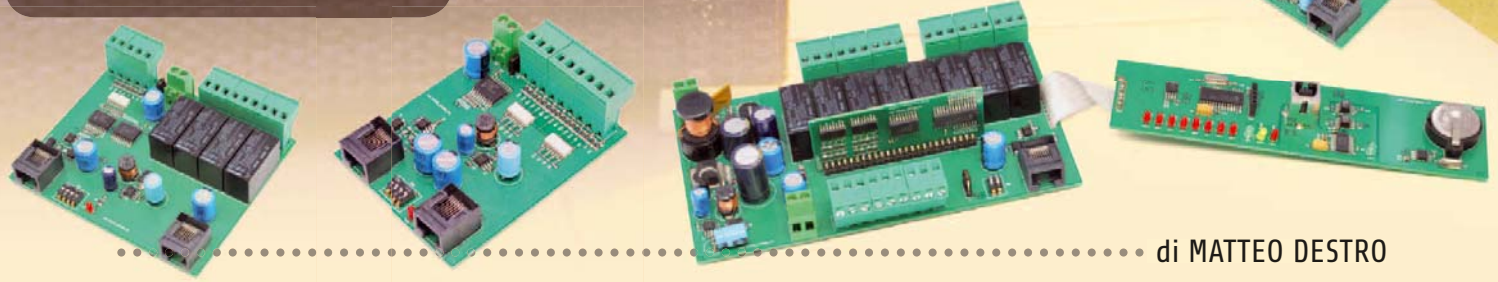
www.futurashop.it

Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Maggiori informazioni su questi prodotti e acquisti on-line su www.futurashop.it.

MINIBUS, UN BUS PER L'AUTOMAZIONE

Un sistema flessibile per la automazione domestica, di apparecchiature ed impianti con un modulo centrale di controllo e moduli periferici specializzati collegati con il protocollo I²C a livello di segnale potenziato per poter coprire lunghe distanze. Prima puntata



di MATTEO DESTRO

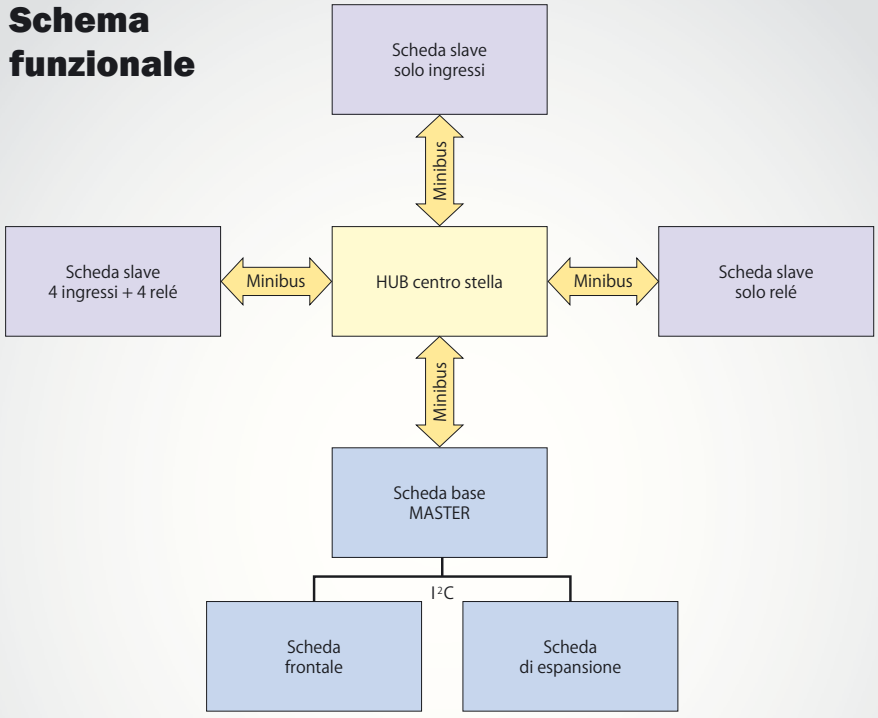
La Domotica è la disciplina che si occupa delle tecnologie atte a migliorare la qualità della vita in casa e svolge un ruolo importante nel rendere "*intelligenti*" apparecchiature, impianti e sistemi.
Ad esempio un impianto elettrico intelligente può regolare la gestione dell'irrigazione del prato di casa facendolo irrigare solo in determinate ore della giornata e per un periodo di tempo prestabilito. Si può quindi affermare che con "*casa intelligente*" si indica un ambiente domestico dove apparecchiature e sistemi sono in grado di svolgere funzioni parzialmente autonome o programmate dall'utente, che può interagire utilizzando opportune interfacce come pulsanti, telecomandi ecc. Il sistema di controllo centralizzato provvede a svolgere i comandi impartiti dall'utente (ad esempio l'accensione di una luce) oppure gestisce in modo autonomo una serie di regolazioni.
Il sistema proposto in queste pagine ha come obiettivo la gestione di una serie di ingressi e uscite programmate in modo da implementare una serie di automatismi utili alla gestione della casa, come ad esempio, l'irrigazione del prato.
Inizialmente ideato per gestire l'irrigazione del prato di casa dell'autore, si è evoluto in un sistema più generale con uscite digitali che rispondono ai comandi impartiti mediante la programmazione degli stati degli ingressi digitali.
Utilizzando un software sviluppato appositamente per questa applicazione, sarà possibile configurare fino a 32 zone di intervento automatiche le quali agiranno su una delle possibili uscite a disposizione. Ognuna delle 32 zone può pilotare l'uscita ad essa associata in base all'ora e al giorno della settimana, in più per ogni giorno si possono definire fino a sette punti di intervento. Un punto di intervento è l'informazione codificata di quando l'uscita deve essere attivata, nell'arco delle 24 ore, e quando deve

essere disattivata.
In aggiunta è possibile forzare in quali mesi dell'anno la zona in esame può pilotare l'uscita ad essa associata. **Ad esempio, nel caso in cui la funzione da assolvere sia l'irrigazione del prato, si può decidere che questa sia attiva dal mese di maggio al mese di settembre e che sia disattivata nei restanti mesi**.
La **Fig. 1** mostra un esempio di quanto appena esposto; come si può osservare, si è configurata la zona 1 affinché agisca sull'uscita 1 (punto) del modulo di indirizzo 1. Tale zona è attiva solo nei mesi da Maggio a Settembre. Inoltre l'uscita viene attivata solo e soltanto nei giorni Lunedì, Mercoledì, Venerdì e Domenica dalle ore 6:30 alle ore 7:00. Per ogni giorno della settimana si è configurato un solo punto di intervento.
Oltre alla configurazione delle zone il software permette di impostare se le uscite devono lavorare in configurazione ***NC*** o ***NO*** (***NC*** = Normal Close [normalmente chiuso] e ***NO*** = Normal Open [normalmente aperto]) e se gli ingressi sono attivi sul fronte ***RE*** o ***FE*** (***RE*** = Rising Edge [fronte di salita] e ***FE*** = Falling Edge [fronte di discesa]).
Infine per ognuna delle uscite a disposizione il software può associare ad esse uno o più ingressi di quelli disponibili. Per ognuno degli ingressi si potrà decidere l'azione logica che dovrà compiere sull'uscita ovvero ***SET***, ***RESET*** o ***TOGGLE***.
Tutte queste funzioni, insieme a tutte le altre non ancora menzionate, verranno trattate in dettaglio nelle prossime puntate.
Proponiamo quindi un sistema composto da una serie di moduli di ingresso/uscita (moduli di tipo ***SLAVE***) pilotati da una unità centrale (definita ***MASTER***) che contiene tutta l'elettronica necessaria a gestire le schede SLAVE e a interfacciarsi al PC per la programmazione della memoria EEPROM, la quale conterrà le informazioni necessarie alla gestione di tutti gli ingressi/uscite in esame.
Il sistema mette quindi a disposizione:

- un modulo di tipo ***MASTER***, comprendente 8 ingressi optoisolati e 8 uscite a relé indipendenti. Un Real Time Clock (RTC) per la gestione della data e dell'ora, una interfaccia USB optoisolata per interfacciarsi al PC e un microcontrollore 18F2620 per la gestione di tutta l'elettronica. Il modulo. ***MASTER*** non è indirizzabile dall'utente, il suo indirizzo logico è 1 (indirizzo fisico 0);
- fino a sette moduli di tipo SLAVE, i quali possono essere del tipo: solo ingressi, solo uscite o ingressi e uscite. Gli ingressi sono sempre optoisolati mentre le uscite sono esclusivamente a relé. Ad ogni modulo deve essere assegnato un indirizzo univoco a partire da 2 fino a 8, l'assegnazione dell'indirizzo viene fatto via hardware utilizzando un dip-switch. L'indirizzo fisico corrisponde all'indirizzo logico meno uno. In pratica se assegniamo alla scheda SLAVE indirizzo logico 8 in realtà si imposterà con il dip-switch, su questa scheda, l'indirizzo 7.

Quanto detto si può riassumere con uno schema a blocchi di massima che evidenzia il collegamento tra i diversi moduli ***SLAVE*** e il

Schema funzionale



Nello schema è evidenziata la logica di collegamento tra i diversi moduli del sistema. L'unità di controllo è composta dalla scheda base MASTER, dalla scheda frontale e dalla scheda d'espansione collegate tra di loro col protocollo I2C. L'unità di controllo si collega alla scheda SLAVE mediante Minibus utilizzando la scheda SLAVE centro stella. Le diverse schede SLAVE sono poi collegate tra di loro, e quindi alla scheda MASTER, utilizzando la scheda HUB centro stella oppure in cascata tra di loro fino ad un massimo di sette schede SLAVE in qualsiasi combinazione tra i diversi tipi disponibili: schede SLAVE solo relé, schede SLAVE solo ingressi e schede SLAVE 4 ingressi più 4 relé.



Fig. 1

modulo ***MASTER*** (**Fig. 2**).
Come si può osservare, il collegamento fisico tra i moduli viene fatto sfruttando il bus I²C adeguatamente trattato per coprire distanze significative.
In seguito tratteremo in dettaglio come viene gestito il collegamento fisico tra le varie schede.

DESCRIZIONE HARDWARE

In questa prima puntata parleremo delle schede base e frontale del modulo ***MASTER,*** lasciando la descrizione della scheda di espansione alla puntata successiva**.** Il modulo ***MASTER***, diversamente dagli altri, è l'unico che può funzionare in modo totalmente indipendente. Infatti l'hardware in esso contenuto permette di gestire fino a 8 ingressi optoisolati e 8 uscite a relé indipendenti sufficienti per soddisfare le esigenze delle applicazioni più semplici.
Osservando lo schema a blocchi dell'hardware (**Fig. 3**) si possono notare tutti i blocchi funzionali che vanno a comporre il modulo. Quasi tutti i blocchi sono collegati tra loro utilizzando il bus di comunicazione I²C tranne per l'interfaccia USB verso il PC che utilizza un collegamento RS-232. Il cuore della scheda è un PIC18F2620 opportunamente programmato il quale deve gestire tutte le comunicazioni su BUS I²C a seconda della configurazione utente salvata nella memoria EEPROM esterna da 1 Mbit.
Il microcontrollore deve comunicare su Bus I²C con i seguenti elementi:

- Memoria EEPROM esterna da 1 Mbit contenente il programma utente per la gestione delle uscite;
- Real Time Clock (RTC) per la gestione della data e dell'ora (programmabile da PC);
- MCP23017 per la gestione della sezione di ingresso optoisolata (8 ingressi) e della sezione di uscita a relé (8 uscite);
- traslatore di livello P82B96 per il collegamento delle schede di espansione.

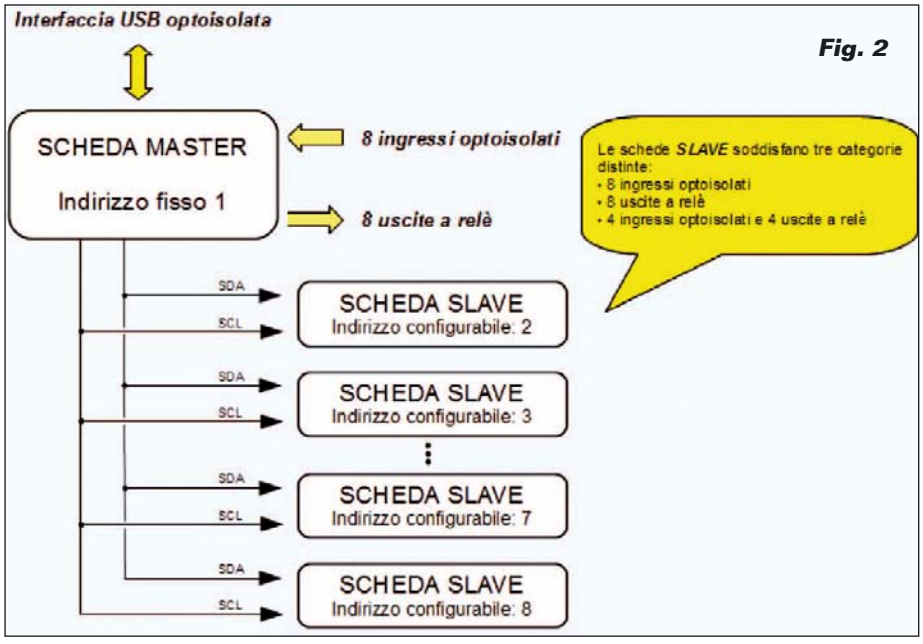
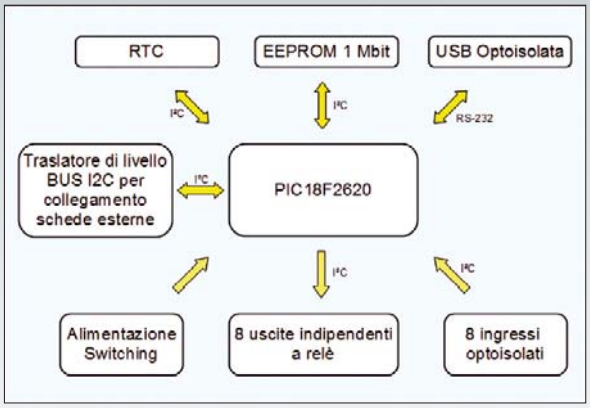


Fig. 2

Nel caso in cui ci siano dei moduli di espansione collegati al modulo ***MASTER***, il microcontrollore dovrà comunicare con ognuno di essi ovvero, dovrà interfacciarsi con l'integrato MCP23008 il quale, a seconda della sua configurazione, dovrà gestire 8 ingressi optoisolati oppure 8 uscite indipendenti a relé. Nel caso invece di modulo misto; 4 ingressi optoisolati e 4 uscite a relé indipendenti.
Come si evince dall'introduzione, il sistema di comunicazione è incentrato sul BUS I²C e sugli integrati di espansione degli I/O, in particolare MCP23017 e MCP23008. L'elettronica presente sul modulo ***MASTER*** è stata suddivisa in tre schede distinte, che sono: scheda base, scheda frontale e scheda di espansione. Le prime due possono essere inserite in un contenitore modu-

Fig. 3



lare da 9 moduli per barra DIN; la scheda di espansione, saldata sulla scheda base, contiene l'elettronica di gestione degli ingressi optoisolati e delle uscite a relé. Gli otto ingressi e le otto uscite sono gestiti attraverso l'integrato MCP23017.

SCHEDA BASE

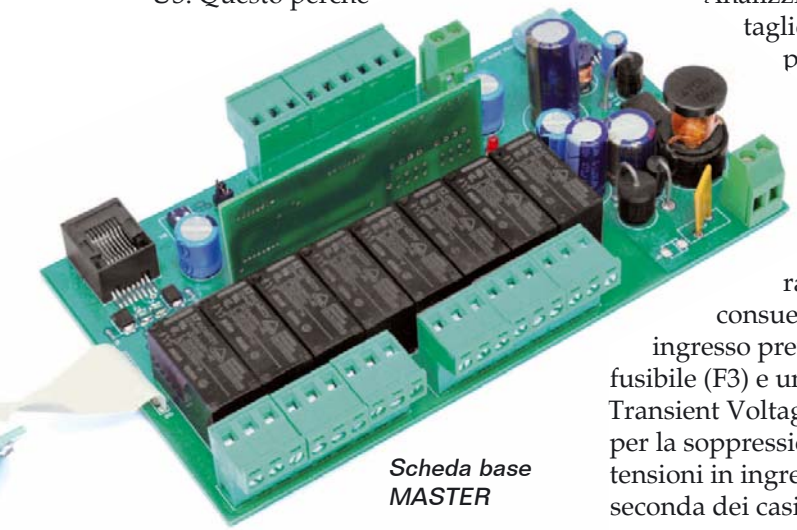
Cominciamo la descrizione degli schemi elettrici partendo dalla scheda base e in particolare dalla sezione di alimentazione: osservando lo schema, si può notare che non è stato predisposto un trasformatore di rete e che quindi non è possibile collegare la scheda direttamente alla tensione di rete.

Dovete quindi scegliere un opportuno trasformatore modulare da barra DIN da abbinare alla scheda, oppure un alimentatore stabilizzato, sempre per barra DIN. In entrambi i casi si deve prestare attenzione alle tensioni di uscita, che devono essere compatibili con la sezione di alimentazione della scheda.

Se utilizzate un'alimentazione esterna stabilizzata a +12Vdc/5A dovete chiudere il ponticello JA e aprire il JB; in questo modo non serve montare C7, C8, C9, C10, C11, D4, D6, L2 e U3. Questo perché la tensione stabile di alimentazione +12Vdc viene direttamente fornita dall'alimentatore esterno. È importante che l'alimentatore sia in grado di fornire almeno 3A di corrente, necessari nel caso in cui si utilizzino più schede di espansione. Il caso che genera il carico maggiore, dal punto di vista dell'assorbimento, si ha quando vengono installati sette moduli di espansione (***SLAVE***) con solo uscite a relè.

Ognuna di esse mette a disposizione otto relè della serie FTR-H1CA012V per un totale di 64 relè (7x8 + 8 del modulo ***MASTER***). La bobina di ciascun relè assorbe 44 mA, quindi se tutti e 64 i relè disponibili fossero eccitati l'assorbimento salirebbe fino a circa 2,82 A più l'assorbimento dovuto al resto dell'elettronica di ciascuna scheda.

Se invece si utilizza un'alimentazione esterna non stabilizzata da 12 a 24 Vca, si deve chiudere JB e lasciare aperto JA; in questo caso non va montato il solo D7. Con questa configurazione è anche possibile alimentare il circuito con una tensione stabilizzata da 15 a 24 Vcc.

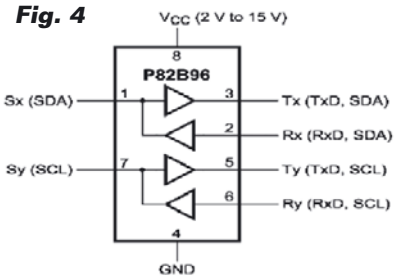


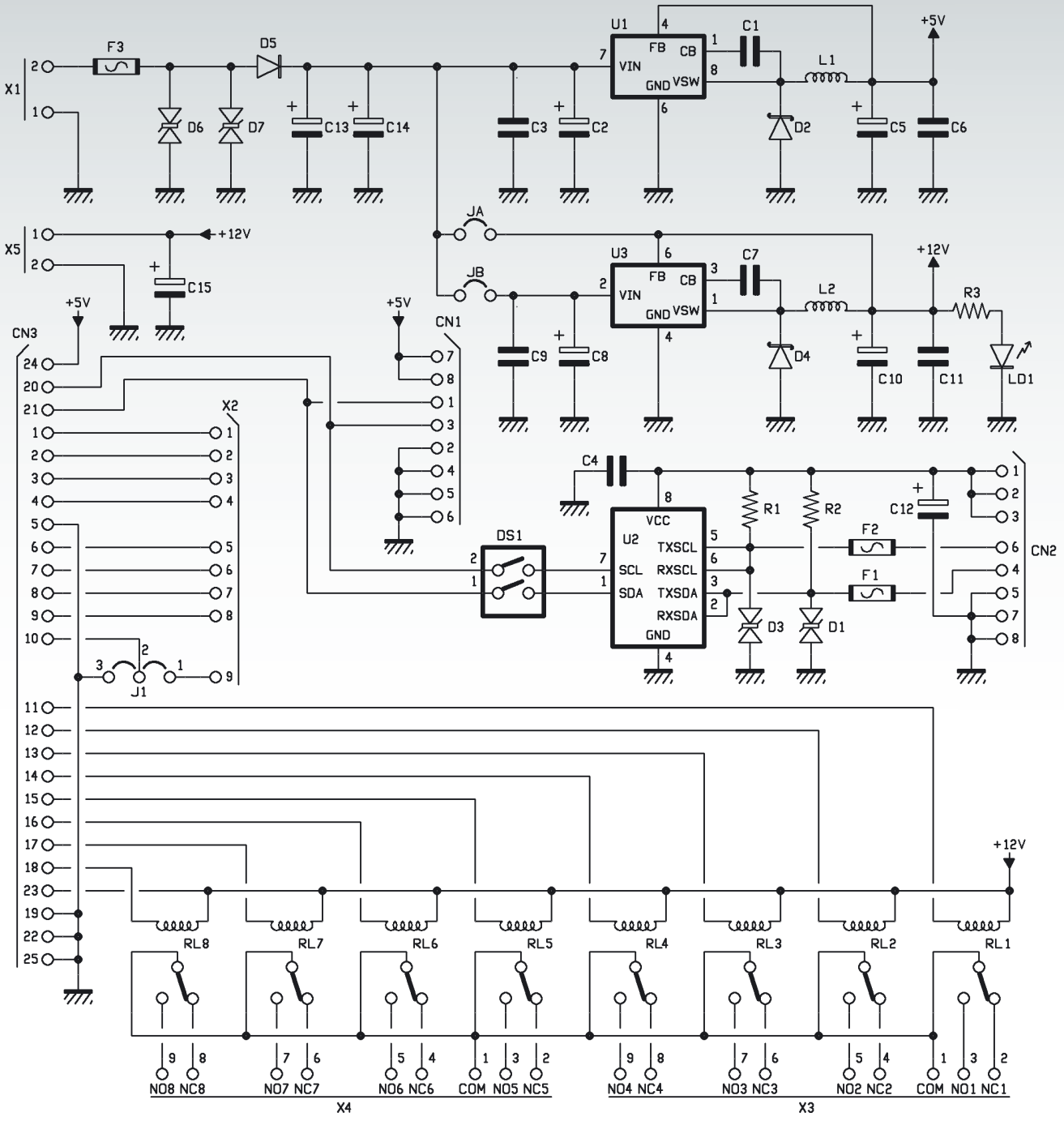
Scheda base MASTER

Analizziamo ora in dettaglio come è stata progettata tutta la sezione di alimentazione a partire dalla morsettiera X1. Come di consueto, la sezione di ingresso prevede un termofusibile (F3) e un TVS (TVS = Transient Voltage Suppressor) per la soppressione delle sovratensioni in ingresso (D6 o D7 a seconda dei casi). Il diodo D6 interviene per tensioni superiori a 33 V, mentre D7 interviene per tensioni superiori a 12V. In entrambi i casi la sovratensione manda in corto il TVS generando un assorbimento anomalo di corrente che fa intervenire il termofusibile F3, il quale, aprendosi, permette di proteggere tutta l'elettronica a valle. Il diodo D5 è un semplice rettificatore.

Seguono due stadi di alimentazione di tipo switching per generare le tensioni +5 V e +12 Vcc; la prima si ottiene da un integrato LM2674M-5.0 (U1) in configurazione step-down già utilizzato in precedenti progetti. La corrente massima che può erogare è pari a 500 mA con un'efficienza massima pari al 92 %. La tensione +5 V viene portata a tutti gli integrati presenti sia sulla scheda base che sulla scheda frontale, nonché sulla scheda di espansione. Il secondo stadio di alimentazione serve per generare 12 Vcc stabilizzati, necessari ad alimentare l'integrato U2 e tutti i relè presenti sulla scheda base, nonché tutti i moduli ***SLAVE*** che verranno collegati al modulo ***MASTER*** utilizzando il connettore CN2. Il connettore

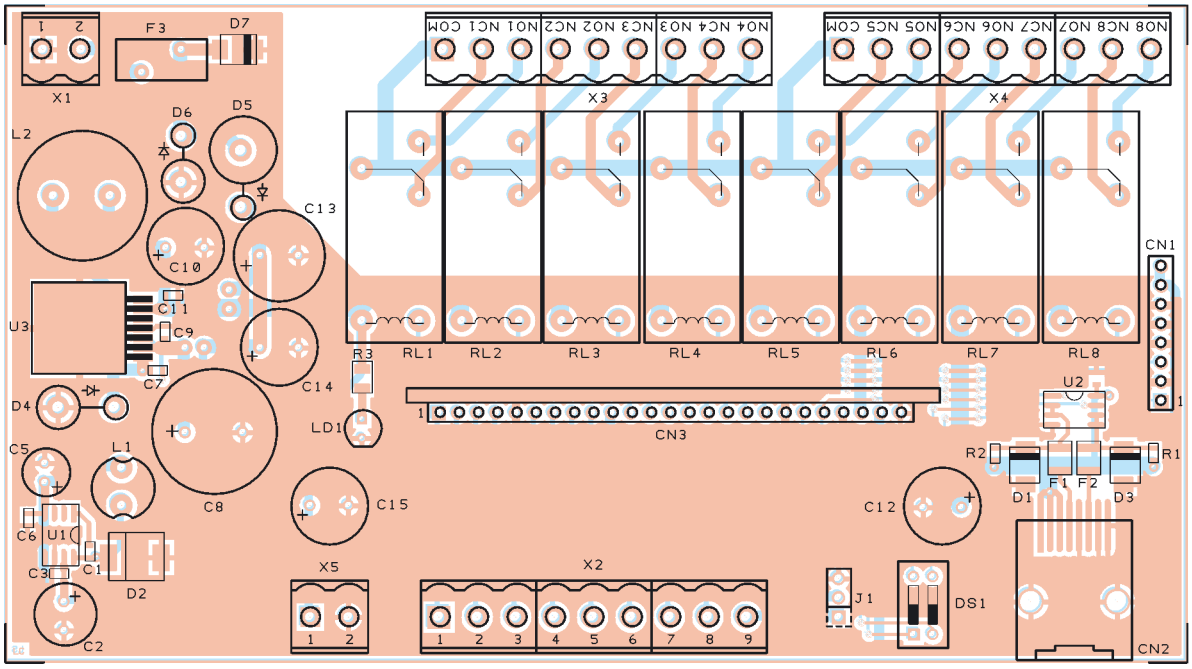
Fig. 4

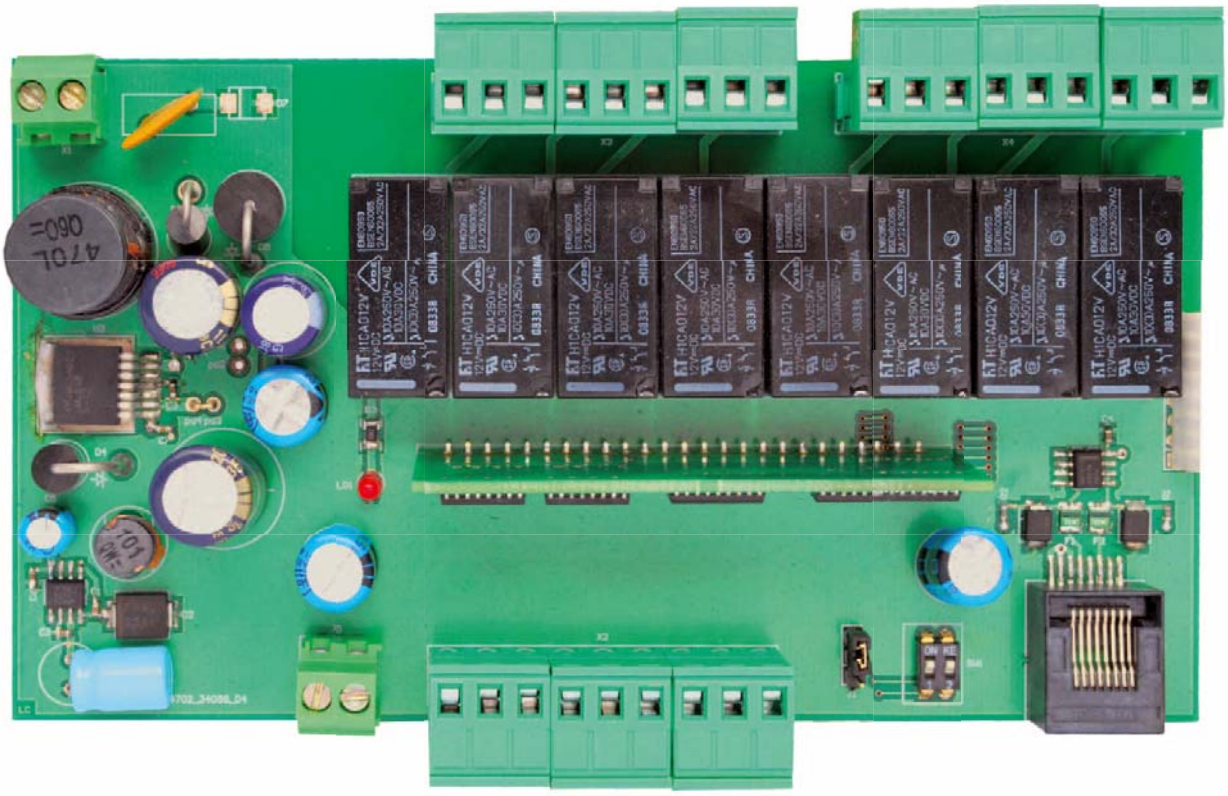




CN2 è un connettore di rete RJ45, quindi si capisce che per collegare i moduli ***SLAVE*** si dovrà usare un cavo di rete twistato opportunamente crimpato per l'applicazione in esame. La tensione +12 V viene anche portata alla morsettiera X5, nel caso in cui si vogliano alimentare gli interruttori da collegare agli ingressi della morsettiera X2. In tal caso il jumper J1 deve essere chiuso tra i pin 2 e 3, in modo che il riferimento di massa per la sezione degli ingressi sia lo stesso della scheda base. Se invece si vuole alimentare gli interruttori con una tensione esterna, il jumper J1 deve essere chiuso tra i pin 1 e 2, in modo da disaccoppiare la massa degli interruttori da quella della scheda base.

L'integrato utilizzato per ricavare i +12 Vdc è un LM2678S-12.0 (U3) il quale, come il precedente, viene fatto lavorare in configurazione step-down. La corrente massima che può erogare è di 5 A, più che sufficiente per l'applicazione, e la sua efficienza





massima è pari al 92 %.
Il numero di componenti necessario per ognuno dei due regolatori switching è molto ridotto, ma bisogna prestare attenzione alla scelta del valore dell'induttanza (componente chiave nei regolatori switching) e del tipo di diodo di ricircolo, nonché del valore capacitivo per la sezione di ingresso e di uscita.
I relè utilizzati sono della serie FTR-H1CA012V con bobina da 12 V e vengono pilotati dall'integrato ULN2803ADW montato sulla scheda di espansione (CN3).
Il collegamento verso tutti i moduli ***SLAVE*** viene fatto sfrut-

Elenco Componenti:

R1: 2,2 kohm 1% 0603
R2: 2,2 kohm 1% 0603
R3: 4,7 kohm 1% 0603
C1: 10 nF 50 VL ceramico 0603
C2: 100 μF 50 VL elettrolitico
C3: 100 nF 50 VL ceramico 0603
C4: 100 nF 50 VL ceramico 0603
C5: 100 μF 25 VL elettrolitico
C6: 100 nF 50 VL ceramico 0603
C7: 10 nF 50 VL ceramico 0603
C8: 820 μF 35 VL elettrolitico
C9: 100 nF 50 VL ceramico 0603
C10: 560 μF 25 VL elettrolitico
C11: 100 nF 50 VL ceramico 0603
C12: 220 μF 50 VL elettrolitico
C13: 470 μF 50 VL elettrolitico
C14: 220 μF 50 VL elettrolitico
C15: 220 μF 50 VL elettrolitico
U1: LM2674M-5.0
U2: P82B96
U3: LM2678S-12.0
D1: SMBJ13CA
D2: SS34
D3: SMBJ13CA
D4: 80SQ045
D5: GI752-E3
D6: 1.5KE33CA
D7: SMBJ13CA
LD1: LED 3 mm rosso
F1: Termofusibile autoripristinante 50 mA
F2: Termofusibile autoripristinante 50 mA
F3: Termofusibile autoripristinante 2500 mA
L1: Induttanza 100 μH 800 mA
L2: Induttanza 47 μH 4300 mA
RL1÷RL8: Relé 12V JW1FSN
DS1: Dip-Switch 2 poli

Varie:
- Connettore RJ45 da CS Molex 95501-6889
- Morsetto 2 poli ad inserzione (2 pz.)
- Morsetto 9 poli ad inserzione (3 pz.)
- Strip maschio 3 poli
- Strip maschio 8 poli
- Scheda a inserzione Clock_Madre_InOut
- Jumper
- Circuito stampato

tando il bus I²C; e per questo, su ognuno dei moduli ***SLAVE***, è presente un integrato MCP23008 (I/O expander), scelto con l'interfaccia di comunicazione I²C (ne esiste anche una versione con interfaccia SPI). La distanza tra il modulo ***MASTER*** e i moduli **SLAVE** può raggiungere anche le decine di metri, di molto superiore alle distanze tipiche del protocollo, quindi si rende necessario ricorrere ad un accorgimento, che consiste nel traslare i livelli ad un valore di tensione più alto del livello originale di 5 volt. Tale funzione viene svolta dall'integrato P82B96, che trasforma gli impulsi 0/5V in impulsi corrispondenti a 0/12 V, in modo da rendere i dati meno esposti ai disturbi captati dai fili del bus.

La **Fig. 4** mostra lo schema a blocchi interno dell'integrato P82B96: come si può notare i pin siglati Sx e Sy vengono utilizzati per collegare, rispettivamente, i segnali SDA e SCL al livello di tensione normale, mentre i pin Tx, Rx e Ty, Ry sono gli omonimi per la parte di segnale traslata di livello a 12 V. Quindi in uscita ritroviamo due pin per ogni segnale di ingresso, perché è necessario garantire la bidirezionalità del segnale. In pratica le uscite Tx e Rx vengono collegate assieme e con una resistenza di pull-up connessa ai +12 V. Similarmente per le uscite Ty e Ry.

Come protezione per eventuali sovratensioni o assorbimenti anomali di corrente vengono introdotti per ogni linea un TVS e relativo termofusibile (si sfrutta il concetto esposto per la sezione di alimentazione).

Su tutti i moduli ***SLAVE*** è presente un P82B96 che riporta il segnale al livello di tensione e corrente originale. Il cavo utilizzato per i collegamenti tra i moduli, oltre a trasmettere le informazioni, porta anche l'alimentazione (+12 V) utilizzata dai moduli ***SLAVE*** per ricavare la tensione interna di alimentazione di +5 V.

La **Fig. 5** mette in evidenza un possibile collegamento tra il modulo ***MASTER*** e gli eventuali moduli ***SLAVE***. È importante notare che l'integrato P82B96 può essere collegato a uno o più integrati P82B96 solo dal lato a più alto livello di tensione, ovvero sfruttando le coppie di pin Tx, Rx e Ty, Ry ma, non è possibile collegare tra loro più integrati P82B96 sfruttando i pin Sx e Sy.

La comunicazione verso la scheda frontale viene fatta utilizzando lo strip di collegamento CN1; i segnali da riportare sulla scheda frontale riguardano le sole linee SCL e SDA più la massa e l'alimentazione stabilizzata +5Vdc.

SCHEDA FRONTALE

Analizziamo lo schema elettrico della scheda frontale, la quale ospita il cuore del sistema, ossia il microcontrollore PIC18F2620 (U4). Ad esso vengono collegati, sfruttando il bus di comunicazione I²C, gli integrati 24FC1025 (U6 memoria EEPROM da 1Mbit) e l'integrato DS1307 (U5 RTC =

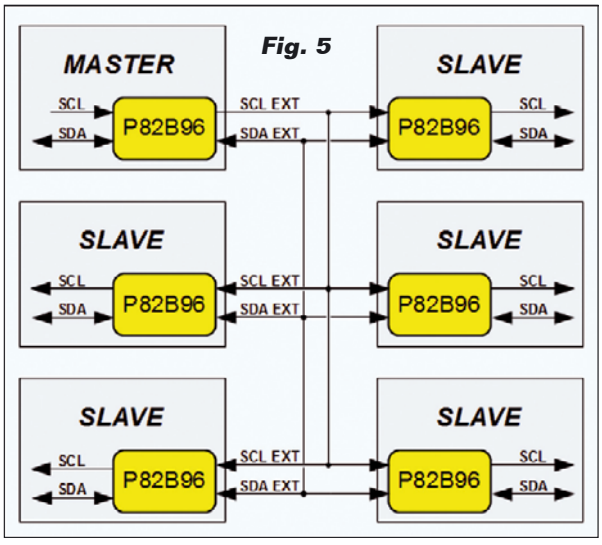
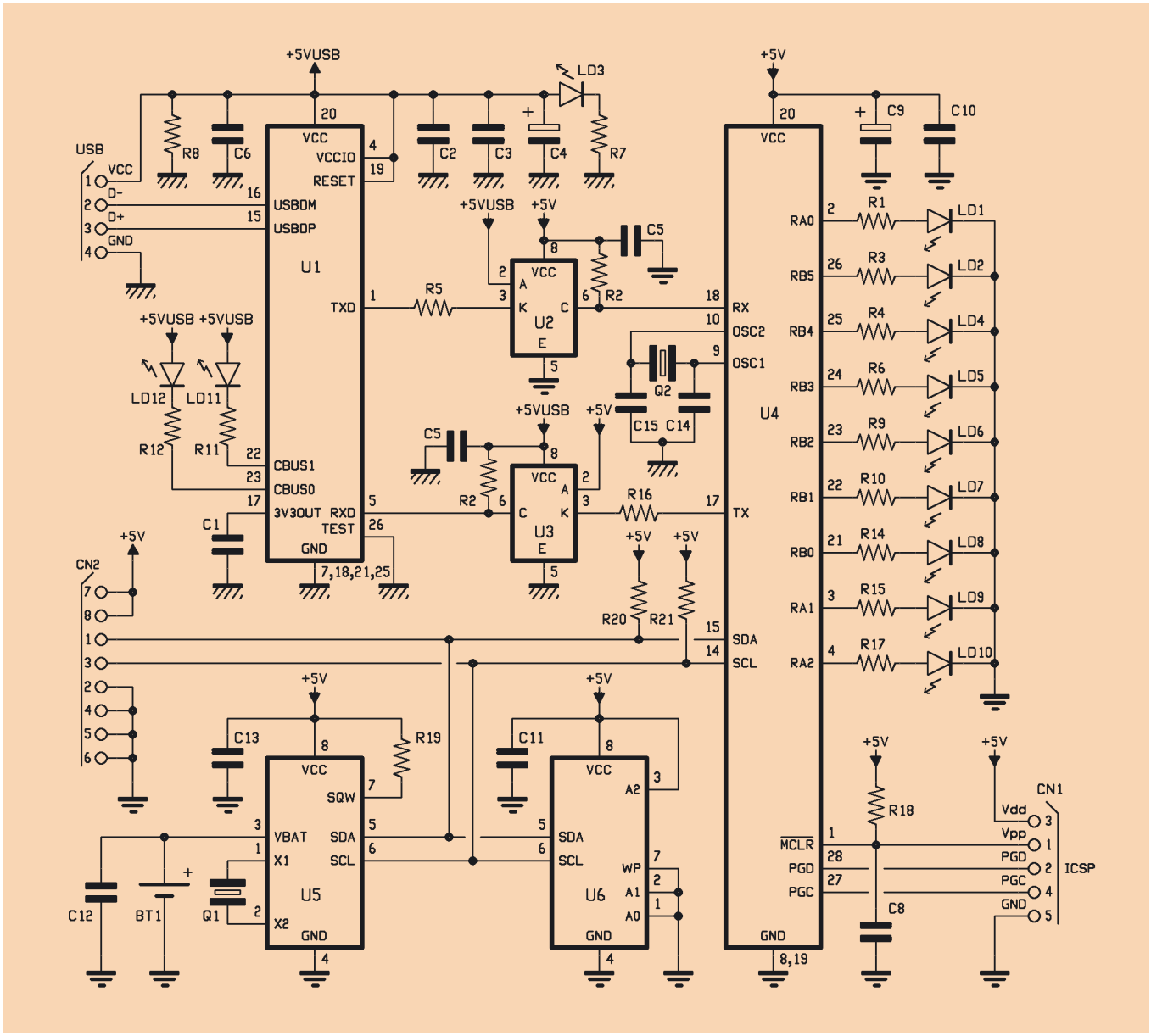


Fig. 5

Tabella 1

Address	Description
0x00100 ÷ 0x001FF	Zona 1
0x00100 ÷ 0x0011F	Lunedì Zona 1
0x00100 ÷ 0x00103	Punto di intervento 1 (Start HH:MM) → (End HH:MM)
0x00104 ÷ 0x00107	Punto di intervento 2 (Start HH:MM) → (End HH:MM)
0x00108 ÷ 0x0010B	Punto di intervento 3 (Start HH:MM) → (End HH:MM)
0x0010C ÷ 0x0010F	Punto di intervento 4 (Start HH:MM) → (End HH:MM)
0x00110 ÷ 0x00113	Punto di intervento 5 (Start HH:MM) → (End HH:MM)
0x00114 ÷ 0x00117	Punto di intervento 6 (Start HH:MM) → (End HH:MM)
0x00118 ÷ 0x0011B	Punto di intervento 7 (Start HH:MM) → (End HH:MM)
0x0011C	Flag abilitazione punto di intervento. "1" punto di intervento attivo, "0" punto di intervento non attivo D7 D6 D5 D4 D3 D2 D1 D0 x P7 P6 P5 P4 P3 P2 P1
0x0011D ÷ 0x0011F	Byte liberi
0x00120 ÷ 0x0013F	Martedì Zona 1
0x00140 ÷ 0x0015F	MercoLEDì Zona 1
0x00160 ÷ 0x0017F	Giovedì Zona 1
0x00180 ÷ 0x0019F	Venerdì Zona 1
0x001A0 ÷ 0x001BF	Sabato Zona 1
0x001C0 ÷ 0x001DF	Domenica Zona 1
0x001E0 ÷ 0x001FF	Flag, Indirizzo e Punto
0x001E0 ÷ 0x001E3	**4 Byte per gestione Flag** 12 Bit Gestione mese associato alla zona (M1 – M12; M = "1" mese abilitato, M = "0" mese non abilitato). 1 Bit Gestione zona attiva, (Z = "1" zona attiva, Z = "0" zona non attiva) 1 Bit Selezionato modulo solo uscite (U = "1" modulo solo Out) 1 Bit Selezionato modulo sia ingressi che uscite (IU = "1" modulo In/Out M12 M11 M10 M9 M8 M7 M6 M5 M4 M3 M2 M1 x x x x x x x x x x x x x x x x x IU U Z
0x001E4	Indirizzo associato alla zona
0x001E5	Punto associato alla zona
0x001E6 ÷ 0x001FF	Free

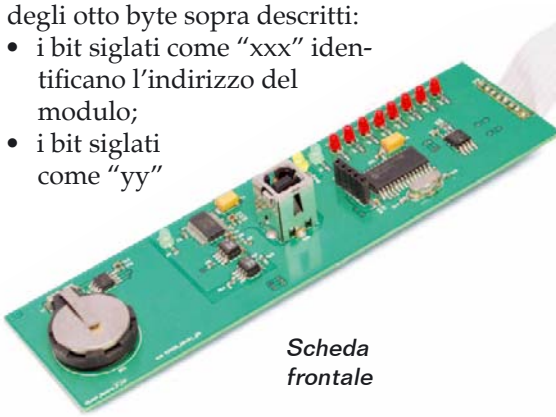
Real Time Clock).
Nella memoria EEPROM, attualmente, vengono memorizzati solo i dati inerenti alla gestione delle 32 fasce orarie, occupando 8.192 byte (per ogni fascia oraria sono stati allocati 256 byte). I dati inerenti alle fasce orarie partono dalla locazione 0x00100 perché i primi 256 byte vengono lasciati liberi per eventuali sviluppi futuri, come la memorizzazione di flag o costanti. La **Tabella 1** mostra come viene allocato lo spazio in memoria per la gestione di una zona (zona 1).
Per ogni giorno della settimana si possono definire fino a sette punti di intervento, ognuno dei quali richiede quattro byte per la memorizzazione (ad esempio indirizzi da 0x00100 a 0x00103). In questi quattro byte vengono memorizzati l'ora e i minuti di attivazione dell'uscita corrispondente e l'ora e i minuti di disattivazione. Per sapere se il punto di intervento n-esimo è attivo, è necessario memorizzarne lo stato in un flag. Essendo sette i punti di intervento, è sufficiente un solo byte come evidenziato dalla tabella (ad esempio locazione 0x0011C). Gli ultimi tre byte dei 32 allocati sono liberi e forzati a 0xFF. Quanto descritto è valido per i restanti giorni della settimana inerenti la zona 1. Ovviamente il discorso si espande e si ripete per tutte le restanti 31 fasce orarie. In coda a tutte le informazioni appena descritte trovano posto quattro byte per la memorizzazione dei flag generici inerenti la zona n-esima più due byte per la memorizzazione dell'indirizzo e del punto dell'uscita associata alla zone in esame.
Per quanto riguarda i flag va detto che:



- il bit "Z" evidenzia se la zona è attiva oppure no;
- il bit "U" evidenzia se l'indirizzo e il punto assegnati fanno parte di un modulo di sole uscite;
- il bit "IU" evidenzia se l'indirizzo e il punto assegnati fanno parte di un modulo misto di ingressi/uscite;
- tutti i bit siglati "x" non sono utilizzati;
- iI flag da "M1" a "M12" evidenziano in quali mesi la zona configurata è attiva (M1 corrisponde a Gennaio, mentre M12 corrisponde a Dicembre).

Oltre alla mappatura appena descritta, ne esiste una seconda inerente alla memoria EEPROM interna al PIC18F2620; che trovate nella **Tabella 2**.
In testa alla memoria, nei primi 8 byte (da 0x0000 fino a 0x0007) vengono allocate le informazioni sui moduli collegati al Bus.
All'indirizzo 0x0000 trovano posto le informazioni funzionali del modulo ***MASTER*** all'indirizzo 0x0001 le informazioni funzionali del modulo ***SLAVE*** di indirizzo logico 2 (indirizzo fisico 1) e così via fino all'indirizzo 0x0007, dove sono memorizzate le informazioni funzionali del modulo ***SLAVE*** di indirizzo logico 8 (indirizzo fisico 7).
Analizzando in dettaglio uno degli otto byte sopra descritti:

- i bit siglati come "xxx" identificano l'indirizzo del modulo;
- i bit siglati come "yy"



Scheda frontale

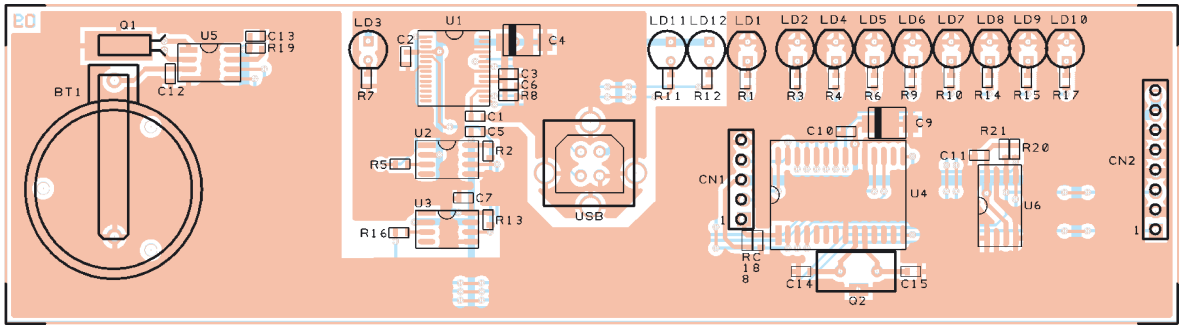
Tabella 2

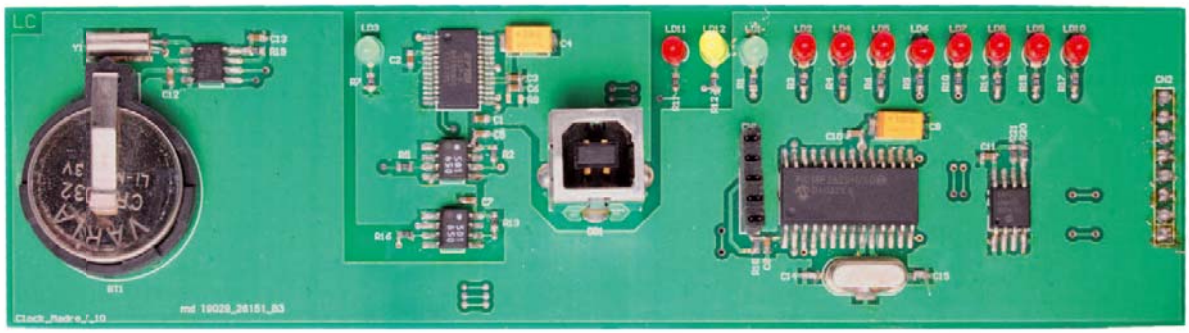
Address	Description							
0x0000 ÷ 0x0007	Indirizzi e tipologia modulo associato							
0x0000	Modulo 0. Indirizzo fisso e non programmabile. Sempre 8 IN e 8 OUT							
0x0001	Modulo 1. Programmabile come IN, OUT o IN/OUT xxx = Indirizzo modulo yy = Tipologia modulo "00" Output, "01" In/Out, "11" Input z = "1" Indirizzo modulo assegnato, "0" indirizzo non assegnato							
	D7	D6	D5	D4	D3	D2	D1	D0
	z	Free	Free	y	y	x	x	x
0x0002 ÷ 0x0007	Modulo 2, 3, 4, 5, 6 e 7							
0x0008 ÷ 0x000F	Maschera sugli ingressi							
0x0008	Modulo 0. Maschera sugli ingressi. "0" ingresso attivo basso, "1" ingresso attivo alto. Default attivo alto							
	D7	D6	D5	D4	D3	D2	D1	D0
	x	x	x	x	x	x	x	x
0x0009 ÷ 0x000F	Modulo 1, 2, 3, 4, 5, 6 e 7							
0x0010 ÷ 0x0017	Maschera sulle uscite							
0x0010	Modulo 0. Maschera sulle uscite. "1" contatto normalmente aperto, "0" contatto normalmente chiuso. Default contatto normalmente aperto (NC)							
	D7	D6	D5	D4	D3	D2	D1	D0
	x	x	x	x	x	x	x	x
0x0011 ÷ 0x0017	Modulo 1, 2, 3, 4, 5, 6 e 7							
0x0018 ÷ 0x001F	Free							
0x0020 ÷ 0x005F	Modulo di uscita Indirizzo 1. Ingressi associati per ogni punto di uscita							
0x0020 ÷ 0x0027	Indirizzo 1 di uscita, Punto 1							
0x0020	Tipo di comando (2 Bit)	Indirizzo Modulo di ingresso (3 Bit)	Punto Modulo di ingresso (3 Bit)					
0x0021	Tipo di comando (2 Bit)	Indirizzo Modulo di ingresso (3 Bit)	Punto Modulo di ingresso (3 Bit)					
0x0022	Tipo di comando (2 Bit)	Indirizzo Modulo di ingresso (3 Bit)	Punto Modulo di ingresso (3 Bit)					
0x0023	Tipo di comando (2 Bit)	Indirizzo Modulo di ingresso (3 Bit)	Punto Modulo di ingresso (3 Bit)					
0x0024	Tipo di comando (2 Bit)	Indirizzo Modulo di ingresso (3 Bit)	Punto Modulo di ingresso (3 Bit)					
0x0025	Tipo di comando (2 Bit)	Indirizzo Modulo di ingresso (3 Bit)	Punto Modulo di ingresso (3 Bit)					
0x0026	Tipo di comando (2 Bit)	Indirizzo Modulo di ingresso (3 Bit)	Punto Modulo di ingresso (3 Bit)					
0x0027	Tipo di comando (2 Bit)	Indirizzo Modulo di ingresso (3 Bit)	Punto Modulo di ingresso (3 Bit)					
0x0028 ÷ 0x002F	Indirizzo 1 di uscita, Punto 2							
0x0030 ÷ 0x0037	Indirizzo 1 di uscita, Punto 3							
0x0038 ÷ 0x003F	Indirizzo 1 di uscita, Punto 4							
0x0040 ÷ 0x0047	Indirizzo 1 di uscita, Punto 5							
0x0048 ÷ 0x004F	Indirizzo 1 di uscita, Punto 6							
0x0050 ÷ 0x0057	Indirizzo 1 di uscita, Punto 7							
0x0058 ÷ 0x005F	Indirizzo 1 di uscita, Punto 8							
0x0060 ÷ 0x009F	Modulo di uscita Indirizzo 2. Ingressi associati per ogni punto di uscita							
0x00A0 ÷ 0x00DF	Modulo di uscita Indirizzo 3. Ingressi associati per ogni punto di uscita							
0x00E0 ÷ 0x011F	Modulo di uscita Indirizzo 4. Ingressi associati per ogni punto di uscita							
0x0120 ÷ 0x015F	Modulo di uscita Indirizzo 5. Ingressi associati per ogni punto di uscita							
0x0160 ÷ 0x019F	Modulo di uscita Indirizzo 6. Ingressi associati per ogni punto di uscita							
0x01A0 ÷ 0x01DF	Modulo di uscita Indirizzo 7. Ingressi associati per ogni punto di uscita							
0x01E0 ÷ 0x022F	Modulo di uscita Indirizzo 8. Ingressi associati per ogni punto di uscita							
0x0230 ÷ 0x03EF	Free							
0x03F0 ÷ 0x03FF	CheckSum Byte							

identificano la tipologia del modulo associato all'indirizzo di cui sopra ("00" = modulo solo uscite, "01" = modulo ingressi/uscite, "11" = modulo solo ingressi);

- il bit siglato come "z" indica se l'indirizzo è stato assegnato e quindi che il modulo MASTER deve aspettarsi che sul bus sia collegato un modulo SLAVE di indirizzo "xxx".

Le locazioni a partire dall'indirizzo 0x0008 (riferito a indirizzo logico 1) fino all'indirizzo 0x000F (riferito a indirizzo logico 8) contengono la maschera sugli ingressi. Ovvero indicano se l'ingresso n del modulo "***xxx***" è attivo sul fronte di salita "***1***" (***RE***) o attivo sul fronte di discesa "***0***" (***FE***).
Le locazioni a partire dall'indirizzo 0x0010 (riferito a indirizzo logico 1) fino all'indirizzo 0x0017 (riferito a indirizzo logico 8) contengono la maschera sulle uscite,





Elenco Componenti:

R1, R3, R4, R6, R7, R9, R10, R11, R12, R14, R15, R17: 470 ohm 1% 0603
R2, R13: 2,2 kohm 1% 0603
R5, R16: 330 ohm 1% 0603
R8, R19: 4,7 kohm 1% 0603
R18: 22 kohm 1% 0603
R20, R21: 2,2 kohm 1% 0603
C1÷C3, C5÷C7, C10÷ C13: 100 nF 50 VL ceramico 0603
C4, C9: 10 µF 25 VL tantalio
C8: 220 pF 50 VL ceramico 0603
C14, C15: 22 pF 50 VL ceramico
U1: FT232RL
U2, U3: HCPL0501
U4: PIC18F2620-I/SO
U5: DS1307Z
U6: 24FC1025
LD1, LD3: LED 3 mm verde
LD2, LD4÷LD11: LED 3 mm rosso
LD12: LED 3 mm giallo
Q1: Quarzo 32768 kHz
Q2: Quarzo 10 Mhz HC49/4H

Varie:
- Porta batteria CR2032
- Strip maschio 5 poli
- Strip maschio 8 poli
- Circuito stampato

ovvero indicano se l'uscita n del modulo "***xxx***" deve lavorare in configurazione normalmente chiuso "***0***" (***NC***) o normalmente aperto "***1***" (***NO***).
In coda a tutte queste informazioni, a partire dall'indirizzo 0x0020, vengono memorizzati i riferimenti dei possibili punti di ingresso rispetto all'uscita associata. Per ogni punto di uscita è possibile associare fino a otto punti di ingresso indipendenti (8 byte). Ogni punto di ingresso prevede tre bit per identificare il punto del modulo di ingresso, tre bit per identificare l'indirizzo del punto di ingresso e 2 bit per identificare l'azione che il punto di ingresso deve compiere sull'uscita, ovvero ***SET*** (setta l'uscita), ***RESET*** (resetta l'uscita) o ***TOGGLE*** (inverte lo stato dell'uscita). Come detto in precedenza oltre alla memoria EEPROM da 1 Mbit sul bus I²C è collegato l'integrato DS1307 (Real Time Clock) che può essere interfacciato al bus a una frequenza massima di 100Khz. Per il suo corretto funzionamento necessita di un quarzo da 32768Hz più una batteria tampone per assicurare che il RTC continui a funzionare anche in assenza di alimentazione. Le informazioni su data e ora sono mappati nella memoria interna del RTC come descritto dalla **Tabella 3**.
Affinché l'oscillatore del RTC sia attivo è necessario che il bit CH, all'indirizzo 0x00, sia settato a 0. Le informazioni di data e ora sono codificate in formato BCD come illustrato dalla tabella. È possibile decidere se l'integrato deve funzionare in modalità 12 ore oppure 24. Per questa impostazione si deve settare/resettare il bit 6 del registro di indirizzo 0x02. Se il bit 6 è settato, cioè 1, il bit 5 serve per indicare se l'ora letta riguarda la mattina (AM) o il pomeriggio (PM). Se invece la modalità di funzionamento è impostata sulle 24 ore allora il bit 5 assieme al bit 4 serve per definire il valore dell'ora in formato BCD.

Tabella 3

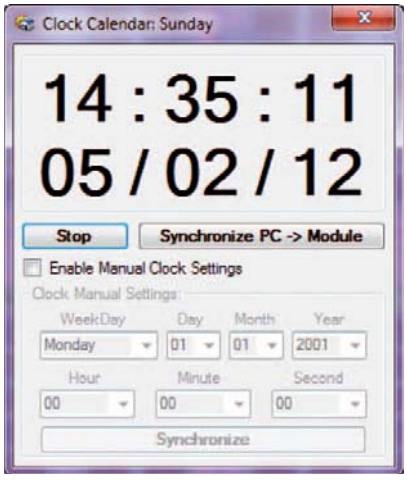
Add	Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	#
0x00	Seconds	CH	10 Seconds			Seconds				00 - 59
0x01	Minutes	0	10 Minutes			Minutes				00 - 59
0x02	Hours	0	12 24	10 HR A/P	10 HR	Hours				01 - 12 00 - 23
0x03	Day	0	0	0	0	0	Day			01 - 07
0x04	Date	0	0	10 Date		Date				01 - 28/29 01 - 30 01 - 31
0x05	Month	0	0	0	10 Month	Month				01 - 12
0x06	Year	10 Years				Years				00 - 99
0x07	Control	OUT	0	0	SQWE	0	0	RS1	RS0	#

La data e l'ora può essere impostata utilizzando il PC e l'apposito software. Quindi una volta abilitata la comunicazione seriale tra il PC e il modulo MASTER sarà necessario richiamare l'apposita finestra per la programmazione, vedi **Fig. 6**. È quindi possibile allineare il RTC presente sulla scheda con la data e l'ora del PC oppure configurare a mano la data e l'ora desiderati.
La comunicazione tra il PC e il PIC18F2620 viene realizzata sfruttando l'integrato FTDI FT232RL (U1) il quale fa da interfaccia tra la periferica RS232 del PIC e l'interfaccia USB del PC. Per opto-isolare il collegamento si utilizzano i foto-accoppiatori U2 e U3 interposti tra l'integrato FT232RL e il PIC. Quando il cavo USB viene collegato al modulo ***MASTER*** il LED LD3 si accende. Ricordiamo che il cavo USB oltre al segnale porta anche l'alimentazione, nella nostra applicazione la utilizziamo per alimentare l'integrato FT232RL (U1) e l'integrato opto-isolatore (U3) nonché i tre LED LD3, LD11 e LD12. I LED LD11 e LD12 servono per evidenziare che una comunicazione tra il PC e il modulo ***MASTER*** è in corso, ovvero per ogni pacchetto dati trasferito dal PC al PIC lampeggerà il LED LD12, nel caso in cui un pacchetto dati verrà trasferito dal PIC al PC lampeggerà il LED LD11.
Sulla scheda sono stati predisposti degli ulteriori LED per la diagnostica, in particolare:

- LD1 indica se la memoria EEPROM è programmata oppure no; se il LED lampeggia, la memoria non è programmata, mentre se è acceso fisso la memoria è programmata correttamente;
- LD2 fa riferimento al modulo di indirizzo logico 1 (modulo MASTER); il modulo MASTER esiste sempre, quindi il LED sarà spento se il modulo non è programmato sarà acceso fisso se il modulo è programmato;
- LD4 fa riferimento al modulo di indirizzo logico 2 (modulo SLAVE); se il modulo è previsto dalla programmazione e se il modulo è collegato al BUS il LED sarà acceso fisso, se invece il modulo non è collegato il LED lampeggerà a indicare la presenza di una anomalia; se invece il modulo non è previsto dalla programmazione, il LED sarà spento;
- LD5÷LD10 fanno riferimento, in sequenza, ai moduli dall'indirizzo logico 3 all'indirizzo logico 8 (moduli SLAVE) e il loro comportamento è identico a quello descritto nel punto precedente.

Tutti i LED appena descritti sono collegati direttamente al PIC18F2620, una parte è collegata alla porta B e una parte alla porta A. La frequenza di lavoro del microcontrollore è impostata a 10 MHz.
Nella prossima puntata parleremo della scheda di espansione del modulo ***MASTER*** e dei moduli ***SLAVE,*** ed entreremo nei dettagli, nel limite del possibile, per quanto riguarda il firmware di gestione di tutto il sistema.

Fig. 6

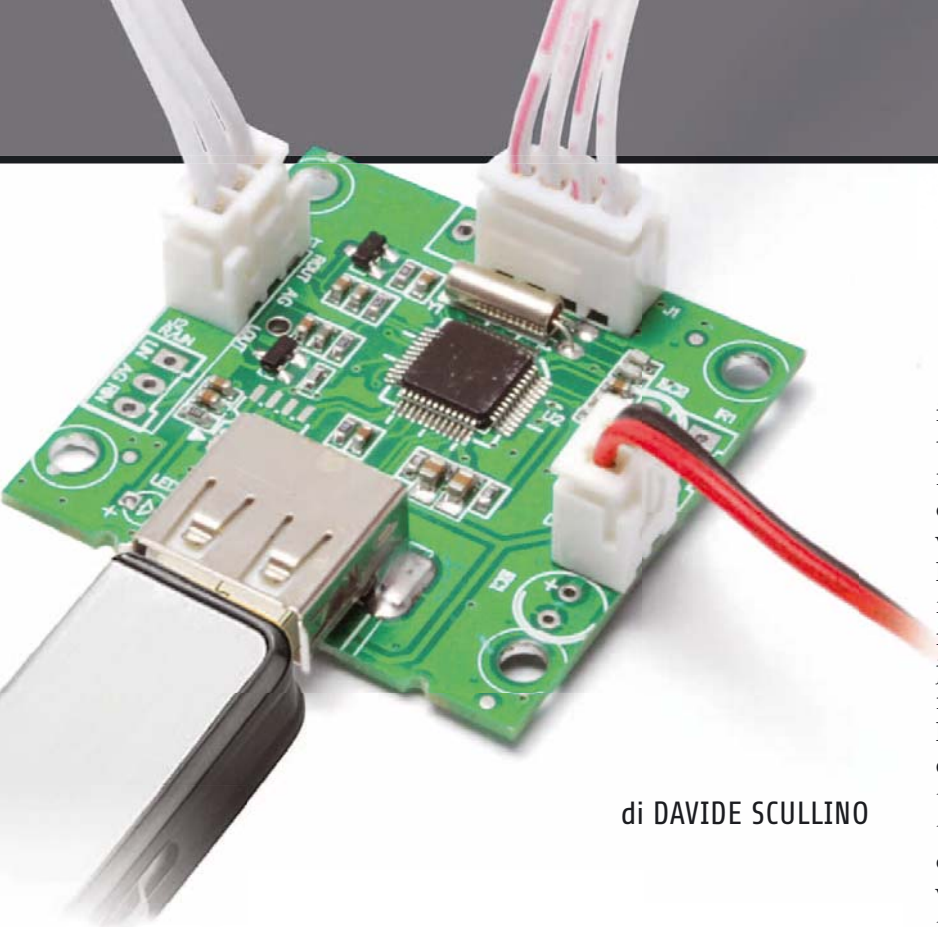




per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono di facile reperibilità. I master dei circuiti stampati possono essere scaricati dal sito della rivista così come il firmware utilizzato per programmare il microcontrollore PIC18F2620-I/SO. Quest'ultimo è disponibile non programmato al prezzo di 10,00 Euro, 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



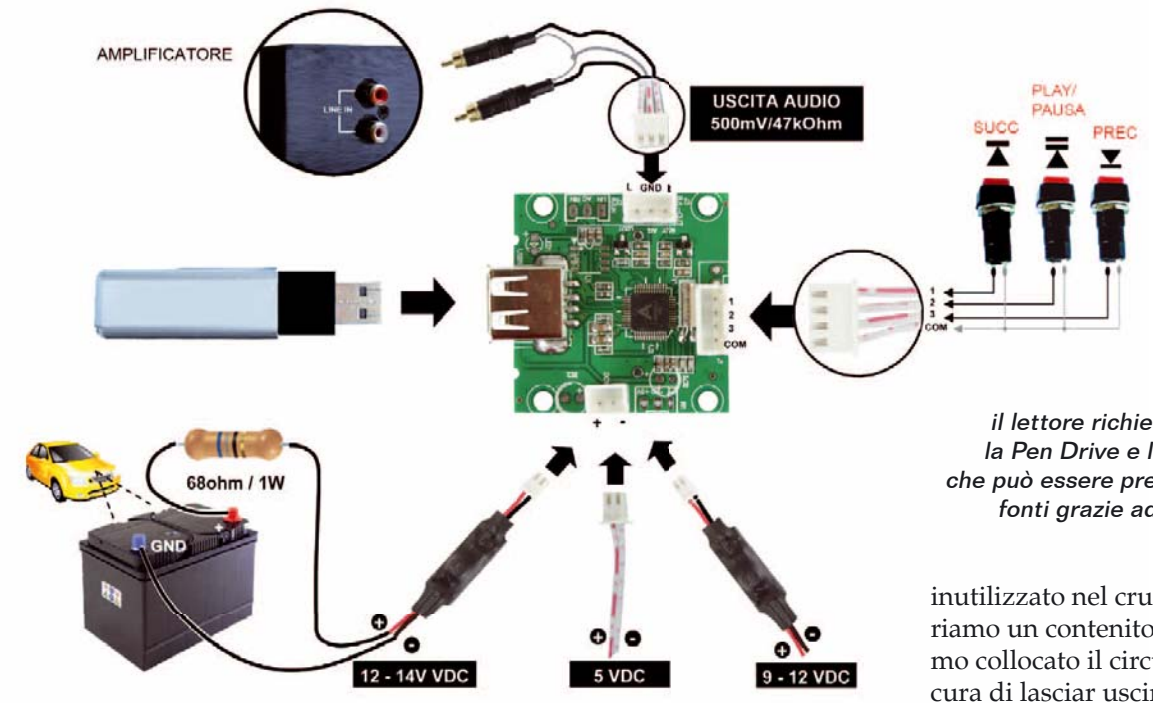
di DAVIDE SCULLINO

LETTORE MP3 JUKE BOX

Riproduce in sequenza i brani musicali registrati su Pen Drive USB della capacità massima di 16 GB; si comanda con tre pulsanti e fornisce in uscita un segnale per un piccolo amplificatore o un hi-fi.

Ormai da tempo, quello di MP3 è il lettore musicale più utilizzato e probabilmente l'unico preso in considerazione da chi voglia portare con sè la propria musica preferita, in viaggio verso il lavoro, durante il jogging, in bicicletta o nei momenti di relax; l'era dei Walkman e dei Compact Disc portatili è definitivamente tramontata e non c'è da sorprendersi, visto che gli MP3 non temono rivali. Infatti la densità di musica riproducibile a parità di peso e ingombro, ma anche di consumo (che inevitabilmente condiziona il peso dovuto alle batterie) che i lettori MP3 consentono, non può essere raggiunta neppure da un lettore di CD-ROM compatibile MP3. Se ancora non avete un lettore MP3 e volete realizzarne uno perché al mero acquisto preferite l'autocostruzione, troverete interessante un prodotto Velleman (distribuito in Italia da Futura Elettronica -*www.futurashop.it*-) nato modulare per poter essere adattato a tutte le situazioni: si tratta di un circuito





Per funzionare, il lettore richiede tre pulsanti, la Pen Drive e l'alimentazione, che può essere prelevata da varie fonti grazie ad un regolatore.

elettronico delle dimensioni di appena 40x40x8 mm (reso così piccolo dal fatto che è interamente realizzato con componenti SMD) fornito già montato e collaudato, contenente un decoder MP3 in grado di acquisire i file direttamente da una Pen Drive USB. Niente di più comodo, quindi, dal momento che ci risparmiamo la memoria on-board altrimenti necessaria e incorporata nei classici lettori MP3 per caricare i file musicali da un computer. Il tutto rende il lettore economico e di facile utilizzo anche in installazione fissa, dato che per riprodurre i brani non dobbiamo averli caricati in esso ma ci basta scriverli sulla memoria USB e poi inserirvi quest'ultima.

Il modulo è poliedrico, in quanto nasce per molteplici impieghi: ad esempio, con una piccola modifica possiamo incorporarlo in un impianto stereo compatto, prendendo da quest'ultimo l'alimentazione e collegando le uscite ad uno degli ingressi stereo disponibili; oppure possiamo alloggiarlo in un proprio contenitore che farà bella mostra di sè sul mobile accanto all'hi-fi.

Ancora, possiamo inserirlo in un piccolo contenitore che lo renda trasportabile e che disponga, allo scopo, di uno scomparto per la batteria di alimentazione e di un piccolo circuito amplificatore per cuffia realizzato con una coppia di transistor o un integrato ad hoc (ad esempio il classico TDA2822).

Un'ultima applicazione potrebbe essere in auto: troviamo un vano inutilizzato nel cruscotto, ci inseriamo un contenitore dove abbiamo collocato il circuito, avendo cura di lasciar uscire la presa USB e i pulsanti di comando, quindi colleghiamo le uscite audio agli spinotti RCA dell'ingresso AUX dell'autoradio; per l'alimentazione è disponibile un apposito modulo riduttore di tensione che dai 12 V della batteria ricava i 5 volt che occorrono al circuito. Ecco che abbiamo dotato la nostra autoradio sprovvista di lettore MP3 di un prezioso riproduttore che ci farà trascorrere anche i viaggi più lunghi in compagnia della nostra musica preferita, senza doverci mai preoccupare di cambiare CD, stazione FM o che altro.

Tutte queste applicazioni sono rese possibili dal fatto che il VM202, così si chiama il kit del lettore, oltre al lettore MP3 già montato e collaudato comprende tutto quello che serve ad alimentare quest'ultimo in diverse condizioni, ma anche a collegarlo con l'amplificatore o l'autoradio e con i pulsanti di controllo; a voi non resta altro che cercare il contenitore (sempre che non intendiate alloggiare il lettore in un apparecchio esistente) e fare i collegamenti del caso.

Più esattamente, il kit comprende il circuito del lettore MP3, due cavetti flat con intestati i connet-

CARATTERISTICHE TECNICHE

- **Formati supportati: MP3 (bit rate 128kb/s, 192kb/s, 256kb/s, 320kb/s)**
- **Livello di uscita audio: 2 x 500 mVeff**
- **Impedenza d'uscita audio: 47 kohm**
- **Tensione d'alimentazione: 5 Vcc (9÷14 Vcc tramite riduttore)**
- **Corrente assorbita: 50 mA max.**
- **Capacità Pen Drive: max. 16 GB**
- **Dimensioni: 45 x 40 x 8 mm**

tori che permettono di collegare l'uscita audio stereo e i pulsanti di comando (devono essere di tipo normalmente aperto), un modulo regolatore racchiuso in un tubo termorestringente e terminante da un lato con una trecciola rosso/nera e dall'altro con un cavo terminante con un connettore adatto a quello di alimentazione del circuito, ed una resistenza limitatrice (68 ohm, 1 W) da applicare in serie all'ingresso del regolatore laddove lo si alimenti con più di 12 Vcc (ad esempio in auto). Vediamo i tre casi applicativi uno ad uno.

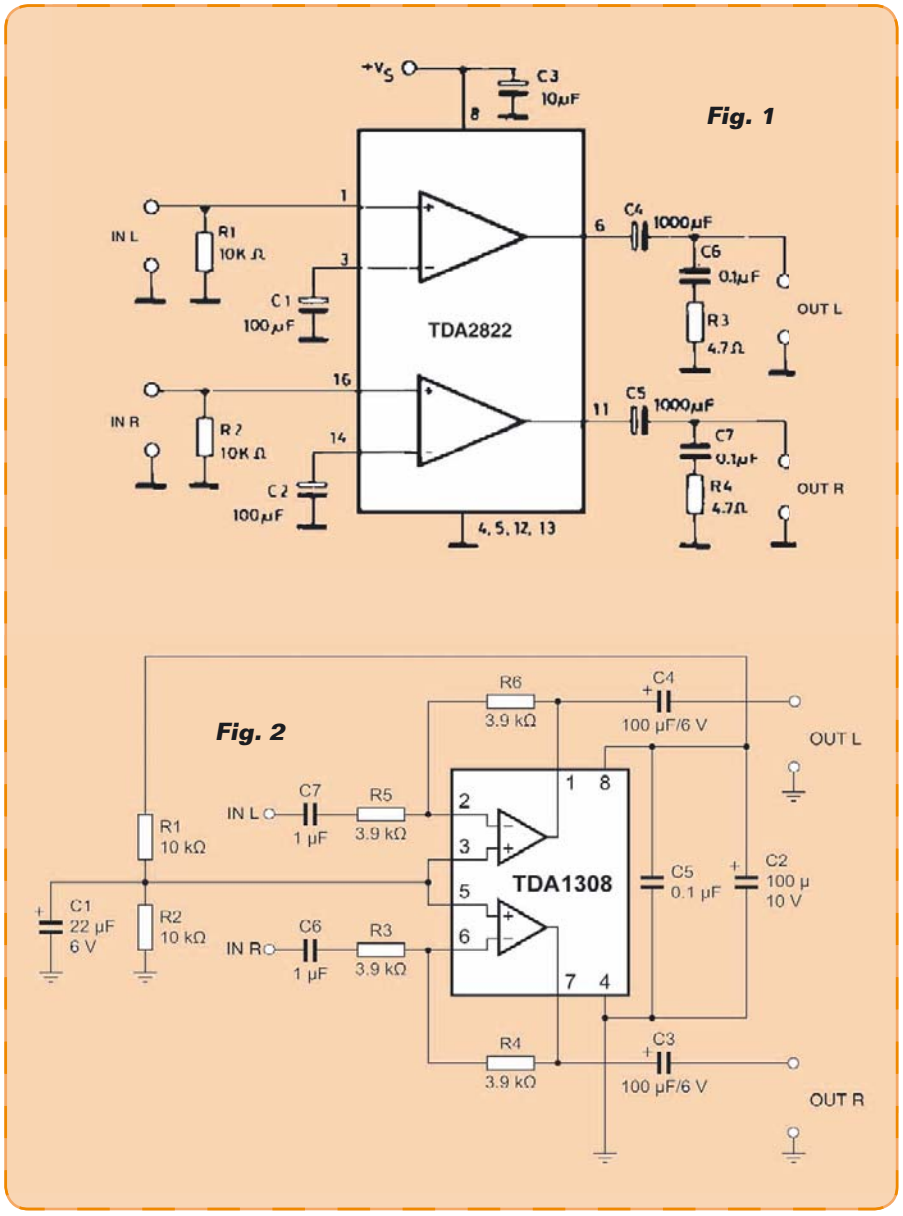


Fig. 1

Fig. 2

INSTALLAZIONE FISSA

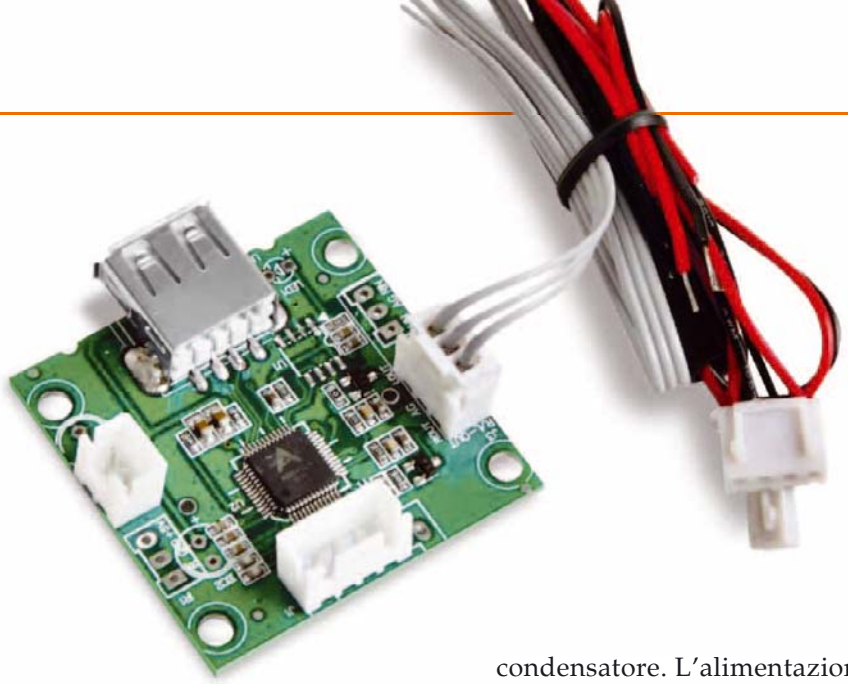
Sono previste due soluzioni: inserimento in un apparato audio esistente o realizzazione di un dispositivo a sè stante; in tutti e due i casi servono i cavi flat per portare fuori i segnali e collegare i pulsanti, ovvero comandare le linee corrispondenti a questi ultimi mediante segnali logici forniti ad esempio da un microcontrollore (per l'esattezza, i segnali logici dovrebbero comandare transistor open-collector che chiudono a massa gli input 1, 2, 3). L'uscita audio va connessa ad un ingresso stereo libero nel primo caso, mentre deve terminare su idonei connettori posizionati sul contenitore nel secondo: suggeriamo i classici RCA usati in tutti gli impianti hi-fi.

Per l'alimentazione, va tenuto presente che il circuito richiede 5 V stabilizzati ed una corrente massima di 50 milliampere, quindi se disponete di una fonte di tensione che soddisfi queste caratteristiche, con essa potete alimentare direttamente i contatti + e – del connettore DC1, altrimenti dovrete avvalervi del regolatore fornito con il kit; in questo caso dovete inserire il connettore del cavo di uscita del regolatore nel predetto DC1 della scheda e fornire alla trecciola rosso/nera (fate attenzione alla polarità: il filo rosso è il positivo ed il nero quello negativo) una tensione di valore compreso tra 9 e 12 V, ovviamente continua.

Se partite da un valore superiore e comunque non oltre i 15 volt, dovete inserire la resistenza da 68 ohm fornita nel kit in serie al cavo rosso della trecciola.

INSTALLAZIONE IN AUTO

In questo caso identificate un vano non utilizzato (di solito ce ne sono, a meno di non avere un'automobile full-optional) o in mancanza, prevedete un piccolo contenitore lavorato per rendere accessibile la presa USB; l'alimentazione va prelevata direttamente dalla batteria (previa interposizione di un diodo di protezione e di un'induttanza da 1 mH per proteggere dall'inversione di polarità e dai disturbi) in modo da limitare le interferenze causate dall'inserimento degli altri utilizzatori elettrici della vettura. Per "direttamente" s'intende non di collegare il con-



nettore DC1 ai 12 V della batteria, ma che il regolatore (in questo caso indispensabile) va collegato ai 12 V dell'automobile, interponendo il resistore da 68 ohm in dotazione; infatti l'impianto elettrico dell'auto, quando il motore è in moto, presenta una tensione anche superiore ai 14 volt.
L'uscita audio stereo va collegata ad un cavetto schermato (lo schermo deve essere connesso alla pista di massa, ovvero al - dell'alimentazione) a due vie terminante con due spinotti RCA, in modo da realizzare la connessione BF inserendo questi ultimi nelle prese volanti dell'ingresso AUX; le autoradio sono costruite in modo da accettare il segnale fornito dal lettore, in quanto i 500 mV sono più che sufficienti ad ottenere la massima potenza dalle casse. Anche l'impedenza è adatta, in quanto l'AUX delle autoradio va tipicamente da circa 20 a 50 kohm.

LETTORE MP3 PORTATILE

Oltre al circuito lettore vero e proprio, bisogna procurarsi un contenitore piccolo e tre pulsanti miniatura da applicare ad esso; per l'ascolto in cuffia è inoltre necessario un amplificatore di bassissima potenza, facilmente realizzabile con due transistor e qualche resistenza e condensatore. L'alimentazione del caso può essere prelevata direttamente da una batteria NiMH da 4,8 volt 300 mAh (nel senso che possiamo applicare la batteria direttamente al connettore DC1) oppure da un pacco Li-Ion da 2 celle, sempre da 300 mAh, ma in questo caso serve interporre il regolatore fornito nel kit. Come amplificatore BF potete utilizzare quello dello schema riportato in **Fig. 1**, basato sul TDA2822 della ST, oppure quello di **Fig. 2**, che usa il TDA1308 della NXP; il primo eroga fino a 2x0,5 W su carico da 8 ohm (125 mW su 32 ohm) e si alimenta da 3 a 9 volt, mentre il secondo sviluppa 2x80 mW su carico da 32 ohm e funziona da 3 a 7 volt. Scegliete voi.

LE FUNZIONI DEL LETTORE

I comandi si impartiscono mediante tre pulsanti, che consentono le seguenti funzioni:

- Play/Pausa (pulsante collegato al contatto 2);
- skip al brano successivo (pulsante collegato al contatto 1);
- skip al brano precedente (pulsante relativo al contatto 3).

In breve, il tasto che collegate al contatto 2 permette di avviare la riproduzione o metterla in pausa: la lettura dei file inizia dal primo e termina all'ultimo, in base all'ordine in cui i file MP3 sono memorizzati nella Pen Drive; premendo lo stesso pulsante mentre la riproduzione è in corso, la stessa si ferma e riprende alla successiva pressione, dal punto dove è stata sospesa.
Il pulsante di skip al brano seguente fa saltare la riproduzione dal brano corrente a quello che nell'ordine della Pen Drive viene dopo; il tasto collegato al contatto 3 del J1, infine, fa tornare indietro di un brano. Tutti i pulsanti devono avere come terminale comune il contatto COM del connettore J1 (quello riservato ai comandi) il quale corrisponde alla massa di riferimento.
Notate che per come funziona il circuito, i brani possono essere riprodotti da qualsiasi punto della struttura e del file system della Pen Drive, quindi sia se si trovano sulla root, ossia la directory principale, sia se sono contenuti in cartelle; infatti il circuito all'avvio effettua una scansione per cercare tutti i file MP3, quindi inizia la riproduzione automaticamente facendoci ascoltare i brani in base all'ordine in cui trova i relativi file nelle directory, dando la precedenza a quelli nella root.



per il MATERIALE

Il lettore MP3 descritto in queste pagine è disponibile già montato e collaudato al prezzo di Euro 19,90 IVA compresa (cod. VM202).

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

DAI VOLUME ALLA TUA MUSICA!



Diffusori passivi professionali in ABS

- 500W - 2 vie
- Woofer 15"
- Titanium tweeter 3,5"

€ 220,00 cod. VDSA15

- 450W - 2 vie
- Woofer 12"
- Titanium tweeter 3,5"

€ 158,00 cod. VDSA12



Amplificatore audio di potenza "SAIGOR" 2x120W rms

€ 164,00 cod. VPASG100



Portamicrofono

in ABS per aste con attacco Ø5/8"

€ 1,90 cod. MICAS10





Mixer professionale 3 canali

- 4 ingressi stereo su 2 canali
- Ingresso microfonico con switch "Talk Over"

€ 98,00 cod. PROMIX220



Mixer stereo 3 canali

- Ingressi MIC, CD/Line e Phono

€ 42,00 cod. PROMIX505



Microfono dinamico professionale

- Unidirezionale

€ 19,00 cod. MKPRO8



Stand per microfono*

- Altezza regolabile da 95 a 160cm

€ 24,00 cod. MICS5

* Portamicrofono non incluso



Treppiede per casse acustiche

- In alluminio
- Altezza regolabile da 108÷200cm

€ 42,00 cod. VDL15S

Cuffia stereo professionale per DJ

€ 35,00 cod. HPD15



Cuffia stereo con auricolari dinamici

€ 11,00 cod. HPD13



Prezzi IVA inclusa.

FUTURA ELETTRONICA

Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Maggiori informazioni su questi prodotti e su tutte le altre apparecchiature sono disponibili sul sito *www.futurashop.it* tramite il quale è anche possibile effettuare acquisti on-line.

RoBoTICA

LA FIERA INTERNAZIONALE DELLA ROBOTICA UMANOIDE E DI SERVIZIO

MILANO • 7-9 NOVEMBRE 2012

fieramilano

4ª EDIZIONE

AREE TEMATICHE:

- MECHATRONICS & COMPONENTS
- MEDICAL & HEALTH
- CYBORG & HUMANOID
- SEARCH & RESCUE
- SAFETY & SECURITY
- EDUCATIONAL
- ENTERTAINMENT
- INVENTORS
- 3D & PROTOTYPING
- ARDUINO WORKSHOP



EDIZIONE 2012*

- 15.000 VISITATORI
- 5.000 METRI QUADRI
- 15 CONFERENZE

*DATI PREVISTI

ORGANIZZATO DA

ARTENERGY PUBLISHING S.R.L.

ARTENERGY PUBLISHING S.r.l.
Via Antonio Gramsci, 57 - 20032 Cormano (MI) Italy
Tel.: +39-02-66306866 - Fax: +39-02-66305510
E-mail: info@robotica.pro

IN COLLABORAZIONE CON

RoBoTICA magazine

www.robotica.pro

RoboCup Jr: in finale a Città del Messico

I team italiani alla finale mondiale

a cura della Redazione



I Messico attende le squadre selezionate nelle finali delle competizioni italiane Robocup Jr tenutesi a Roma dal 21 al 23 marzo e a Riva del Garda (TN) dal 19 al 22 aprile. I team di studenti e insegnanti provenienti dalle scuole che si sono distinte nell'ambito delle finali si misureranno ora con quelli delle altre nazioni alla RobocupJunior, che si terrà a Città del Messico dal 18 al 24 giugno prossimi. A loro, che elenchiamo qui sotto, vanno i nostri complimenti, così come a tutti gli altri team che hanno partecipato alle selezioni. RoboCup è un'iniziativa educativa internazionale fondata nel 1997 che sponsorizza eventi di

Squadra	Categoria	Campionato	Scuola di appartenenza
SPQR Levis	Soccer Junior	Light Weight - Field A	ITIS G. Galilei - IISS J. Von Neumann
Bionics Soccer	Soccer Junior	Open League - Field A	IT "Archimede" Catania
SPQR Gravis	Soccer Junior	Open League - Field B	ITIS G. Galilei - IISS J. Von Neumann
CARDANO ROBOTIX	Soccer Junior	Open League - Field B	I.I.S. "Piazza della Resistenza" Monterotondo
Manetti Globetrotters	Soccer Junior	Open League - Field B	ISIT Manetti Grosseto
SOSBEARZI	Rescue	Rescue A	Istituto Salesiano G.Bearzi
Endless Runner	Rescue	Rescue A	ITIS "G.Chilesotti"
Jujitsurossi	Rescue	Rescue B	ITIS "A. Rossi"
GruzanteDance	Dance	CoSpace Theatre (Demo)	ITIS "Carlo Zuccante" - Venezia - Italy
Itimonaco1	Dance	Dance	ITI "A. Monaco" Cosenza

L' importante è partecipare

Anche se ad andare in Messico saranno le squadre finaliste, la Robocup Jr di Trento ha avuto tanti vincitori: quelli delle varie categorie in gara ed anche il team il cui robot è stato riconosciuto come il migliore autocostruito. Il premio è stato assegnato al team LMT dell'I.T.I.S. A.Rossi di Vicenza, formato da Davide Valisa (4ª indirizzo elettronica-informatica e telecomunicazioni) Andrea Padovan (4ª indirizzo elettronica-informatica e telecomunicazioni) e Gianluca Faccin (4ª indirizzo meccatronico) coordinati da Stefano Andriolo. Si tratta di un robot la cui elettronica è basata su Arduino e la cui meccanica è stata realizzata con una nostra CNC.
"Robottino", questo è il curioso nome dato dai ragazzi, ha per cervello un Arduino Mega2560 dotato di shield SparkFun a ponti ad H per gestire i due motori DC della trazione, comandati in PWM. Un secondo shield autocostruito e basato sulla Mega proto Shield per Arduino, ospita i connettori per i vari sensori ed attuatori, oltre ad un regolatore fisso di tensione (LM7805), un LED verde per segnalare che il robot è alimentato, più un buzzer, un LED rosso e un dip-switch usati in fase di prova. L'arduino comunica con 3 sensori ad ultrasuoni, 4 sensori ad infrarossi CNY70, un sensore di contatto, un sensore di tilt e 2 servomotori. È presente un altro dip switch vicino al sensore di tilt usato in prova e un interruttore a 3 vie per l' ON/OFF del Robottino.
Per il montaggio sono state adottate soluzioni per ridurre al minimo il "disordine" di cavi in giro per il Robottino in modo che sia pulito e modulare così da poter sostituire ogni singolo componente in caso di rottura. Per quanto riguarda la meccanica, è stato costruito da zero un telaio in alluminio, le cui parti sono state interamente ricavate dal pieno con una fresa a controllo numerico passando per la progettazione CAD ed il successivo CAM. Le varie parti sono state poi sabbiate. Il telaio consiste di 3 piastre di cui 2 laterali uguali con uno spessore di 5mm e con delle asole trasversali atte a ridurre peso e favorire il montaggio dei componenti. I motori sono fissati con delle viti a queste due piastre laterali.
La piastra centrale, di 10mm di spessore, è stata allegerita nella parte sottostante per ridurre il peso. Collegata alle altre due in modo da formare una struttura ad "H", è stata fissata con delle viti M4 e con dei perni di tolleranza H7 per assicurare un perfetto centraggio. Sopra a questa piastra più spessa hanno trovato posto le schede e la struttura di sollevamento, mentre al di sotto la batteria. Il sollevamento viene eseguito grazie un sistema di leve "a parallelogramma" in modo da mantenere la vittima sempre verticale durante il sollevamento. Per quanto riguarda la trazione, sono state montate delle semplici cinghie dentate con delle pulegge guidate, soluzione che permette un'ottima aderenza.



robotica regionali ed internazionali per giovani studenti, con l'obiettivo di promuovere la ricerca nei campi della robotica e dell'intelligenza artificiale, offrendo visibilità al pubblico e proponendo competizioni spettacolari. Per partecipare alla competizione i giovani partecipanti devono costruire dei robot, anche semplici, e programmarli per realizzare movimenti o superare prove di diversi livelli di difficoltà. Ci sono specifiche categorie (soccer, rescue, dance): i primi classificati di ciascun campionato partecipano alla Robocup.
Uno dei modi di promuovere la ricerca ingegneristica, che non sia quella di sviluppare applicazioni specifiche, è proporre obiettivi significativi di lungo periodo. Quando il raggiungimento di questi obiettivi richiede e comporta un impatto sociale significativo, allora si può configurare un Grand Challenge, una grande competizione che è anche una grande opportunità di confronto e crescita dell'intera comunità. Il costruire robot in grado di giocare a calcio non genera di per sè un grande impatto sociale o economico, ma il riuscire nell'intento è da considerare una notevole e concreta conquista sul campo. Noi consideriamo questo tipo di progetti come progetti di riferimento. RoboCup, nel suo settore, è da considerare come tutti gli altri progetti in sviluppo nei settori di ricerca più istituzionali. Nella sua visione più lungimirante, l'obiettivo finale dell'iniziativa RoboCup può essere descritto con la seguente proposizione: "Per la metà del ventunesimo secolo, una squadra di robot umanoidi completamente autonomi vincerà la gara di calcio contro la squadra umana vincitrice dell'ultima Coppa del Mondo (2050), nel pieno rispetto delle regole di gioco ufficiali FIFA."
Proponiamo questo obiettivo come una delle grandi sfide condivise dalla

comunità robotica e dell'intelligenza artificiale per i prossimi 50 anni. Da questi obiettivi generali si sono ramificati una serie di sotto-progetti finalizzati a sviluppare aree di ricerca e sviluppo specifiche e destinati a gruppi di ricercatori di diverse fasce di età e formazione. L'iniziativa nel suo complesso ha poi veicolato la nascita di gruppi di sviluppo che si organizzano in comunità nazionali. Sempre secondo la similitudine con il mondo del calcio, il giudizio sulla validità delle soluzioni adottate dai diversi gruppi avviene sul terreno di gara, in modo da fare emergere, nel corso dei diversi campionati, le strategie via via più remunerative.
Come in tutti i campionati che si rispettino, i vari team si confrontano in gare di selezione a livello regionale, per poi confrontarsi a livello nazionale ed infine a livello mondiale. Quest'anno l'evento si svolgerà a Città del Messico, dove i team finalisti dei campionati nazionali saranno accolti dalla **Mexican Robotics Federation** e si sfideranno nelle varie specialità che sono nate a partire dal tema fondante.
RoboCupSoccer, per mezzo dello sviluppo di strategie di automazione finalizzate al gioco del calcio si prefigge di realizzare soluzioni adatte ad applicazioni utili nel campo del sociale e dello sviluppo industriale. La specialità comprende diversi campionati, ai quali partecipano gruppi di età e formazione crescente, a partire dai simulatori bi e tridimensionali, alle piattaforme robotiche di dimensioni piccole, medie e grandi, alle vere e proprie squadre di robot umanoidi divisi in tre categorie in base all'altezza: (30-60 cm, 90-120 cm e oltre 130 cm). RoboCupRescue ha l'obiettivo di sviluppare robot adatti alla ricerca e salvataggio di superstiti e vittime di disastri di larga scala dovuti a eventi naturali e non. Anche qui sono previsti campionati separati per simulatori e per robot reali.



RoboCup@Home è una categoria recente che si concentra sullo sviluppo di applicazioni robotiche e di robot umanoidi che siano di ausilio nella vita di tutti i gironi a persone con difficoltà come disabili e anziani. Dopo due anni di sperimentazione, la categoria Logistica, sponsorizzata dalla Festo, è entrata nelle competizioni a tutti gli effetti. Lo scopo ultimo di questa categoria è di sviluppare team di robot in grado di auto organizzarsi per ottimizzare processi di trasporto e produzione industriale.
Veniamo infine alla categoria RoboCupJunior, orientata allo sviluppo di progetti in campo educational, destinata ai ragazzi delle scuole di età inferiore ai 19 anni. È un modo innovativo per assimilare concetti di scienza e tecnologia mediante esperienze concrete di elettronica, hardware e software. Nella categoria RoboCupJunior sono compresi i campionati Dance, robot danzanti, vestiti con costumi in coreografia, in grado di muoversi in sintonia con la musica, Soccer, che è il campionato di calcio con regole più semplici del campionato ufficiale, Rescue, robot in grado di compiere un percorso più o meno accidentato, riconoscere oggetti come una lattina di bibita e portarla in salvo in una posizione predefinita.



SHIELD DI CONTROLLO PER SPIDERIN, BIPE E FILIPPO

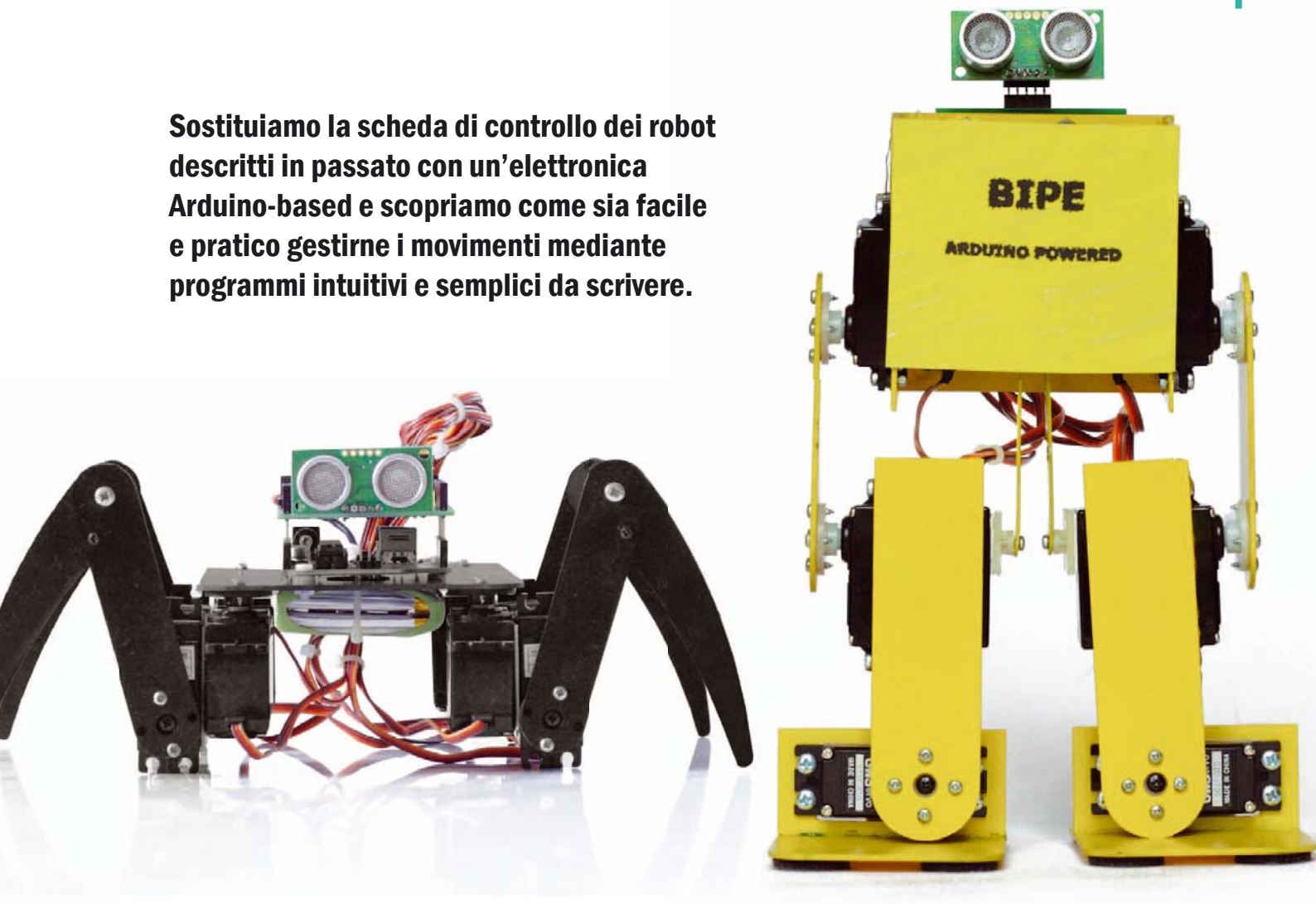
di MIRCO SEGATELLO



Certo ricordate i nostri progetti di robot: dai più semplici su ruote a quelli umanoidi, a quattro zampe e a forma di pesce. Oltre che simpatici gadget, essi nelle intenzioni dovevano essere vere e proprie piattaforme di sviluppo basate su microcontrollore Microchip, con le quali fare un po' di sperimentazione sulla robotica e a cui ognuno di voi può apportare tutte le modifiche e le migliorie che ritiene più opportune. Purtroppo il fatto che ogni progetto si basi su un hardware ed un software specifico scoraggia chi è alle prime armi. Allora ci siamo chiesti: "se dietro ci fosse Arduino, non andrebbe meglio?". Ebbene, l'idea che sta alla base di questo articolo è proprio riproporre i nostri robot di maggior successo in una nuova veste, cioè dotandoli di unità di controllo basata su Arduino. Per realizzare un



Sostituiamo la scheda di controllo dei robot descritti in passato con un'elettronica Arduino-based e scopriamo come sia facile e pratico gestirne i movimenti mediante programmi intuitivi e semplici da scrivere.



hardware adattabile ai vari robot, ci siamo concentrati sui progetti che impiegano servocomandi da modellismo, come ad esempio il robot BIPE ed il robot SPIDER, ma anche FILIPPO, che abbiamo deciso di inserire in questo progetto perché ben si adatta al controllo tramite il nostro hardware.

Questi robot hanno tre elementi in comune: una struttura meccanica, una scheda hardware di controllo ed un software; sebbene l'hardware sia diverso, Arduino è in grado, tramite un apposito shield che comanda i servo ed un firmware ad hoc, di gestire ognuno, in quanto tutti si muovono grazie a servocomandi e rilevano gli ostacoli mediante un sonar. Affinché sia adatto a tutti e tre i robot, lo shield di controllo deve poter gestire un numero di servo pari a quello del dispositivo che ne ha di più, cioè lo Spiderin, che si basa su 8 servo. Arduino ed il suo ambiente di sviluppo fanno il resto. Per avere la massima versatilità, ci siamo imposti la compatibilità con tutte le versioni originali di Arduino (duemilanove o UNO) ed anche con i cloni (Seeduino).

Lo shield ci permette di risolvere le problematiche funzionali, ma dobbiamo fare i conti con l'alimentazione, che non sempre può essere prelevata da Arduino: infatti questa può essere alimentata con una tensione compresa tra 6 e 12 volt e ricavare dal regolatore di tensione interno i 5 volt stabilizzati che, oltre a provvedere alle proprie necessità, possono essere sfruttati dallo shield (tramite il pin "5V") per alimentare eventualmente i servo, i quali tipicamente richiedono da 4,8 a 6 V. Questo ragionamento va bene finché dobbiamo controllare un paio di servo (è il caso del robot più semplice: Filippo) ma se serve più corrente va tenuto presente il limite del regolatore di Arduino; ad esempio, nella scheda Duemilanove il regolatore è un MC33269, che pur potendo erogare 800 mA, in realtà, tenendo conto della massima dissipazione di potenza, non può certo alimentare otto servocomandi contemporaneamente.

Per ovviare a questo limite possiamo alimentare la sezione di controllo dei servo distintamente da Arduino, con delle batterie, ma ricaricabili, in quanto (considerato

l'elevato assorbimento di corrente) le normali pile a secco non durerebbero molto; vanno bene quattro elementi NiMH in serie, i quali a piena carica forniscono 1,5x4=6 V. Man mano che si scaricano, gli elementi scendono a 4 V: non siamo nelle condizioni ottimali di funzionamento del servo e, considerata la notevole variazione di tensione possiamo ipotizzare anche notevoli differenze nelle prestazioni, ma se non scarichiamo le batterie del tutto possiamo farcela.
Come alternativa restano gli elementi Li-Ion o ai polimeri di litio, che sono più performanti, più piccoli e leggeri (a parità di capacità) rispetto alle Ni-Cd e Ni-Mh (le LiPo possono erogare correnti molto elevate); insomma, ottimali per il controllo di motori. Purtroppo ogni singola cella fornisce una tensione nominale di 3,7 volt e quindi, mettendone due in serie, si ottengono 7,4 V, che diventano 8,4 a piena carica: decisamente troppi per un servo normale. Vi è da dire che questo problema di alimentazione affligge anche i modellisti, che devono necessariamente ricorrere a specifici regolatori di tensione, anche se ultimamente sono entrati in commercio servi idonei al funzionamento con due celle LiPo, ma i loro prezzi sono ancora elevati.

IL NOSTRO SHIELD

Alla luce di tutte queste considerazioni, in fase progettuale ci siamo imposti che lo shield abbia un proprio alimentatore interno e, a questo punto, che con esso alimenti anche Arduino. Visto che ci siamo messi a progettare, ci è sembrato il caso di implementare alcune funzionalità in più, come il monitoraggio della tensione della batteria, fondamentale ad esempio per chi usa le batterie LiPo, perché se la tensione di ciascun elemento dovesse scendere sotto i 2,7 volt si danneggerebbero irrimediabilmente. Altre due funzioni che abbiamo aggiunto sono il controllo a distanza tramite un telecomando e la capacità di riconoscere gli ostacoli in modo che il robot non ci vada a sbattere contro.
I breve, il nostro shield deve:

- essere alimentabile con un elevato range di tensioni;
- fornire ai servocomandi una tensione stabilizzata;
- fornire alimentazione alla scheda Arduino;
- disporre di un sensore radar;
- disporre di un ricevitore per il telecomando;
- prevedere la lettura dello stato della batteria.

Analizziamo dunque lo schema elettrico, dove troviamo l'integrato LM2576-5, che nel nostro caso è il regolatore switching da cui ricaviamo l'alimentazione per i servo e per Arduino; alimentato dalla tensione in arrivo sulla morsettiera siglata + e -BAT, U1 eroga 5 V ben stabilizzati con tutta la corrente che serve. U1 è un regolatore switching di tipo buck e funziona sfruttando l'energia immagazzinata nell'induttore L1 quando il transistor collegato al piedino Vo è in conduzione; nelle fasi di interdizione la bobina restituisce l'energia accumulata e la scarica (tramite il diodo Schottky) nei condensatori C3 e C4, determinando la tensione continua V, la cui presenza viene segnalata dal LED D3, alimentato tramite la resistenza R5. La tensione stabilizzata in uscita dall'LM2576 alimenta tutti i servo, mentre la scheda Arduino viene alimentata dalla tensione a valle dell'interruttore di accensione, attraverso il contatto Vin.
Per la lettura della tensione della batteria di alimentazione usiamo un ingresso analogico di Arduino, preceduto dal partitore R1/R2 che la limita al disotto del massimo tollerabile dagli I/O di Arduino; i resistori sono dimensionati in modo che un valore compreso tra 0 e 20,55 volt

Scegliere l'alimentazione

L'adozione di un regolatore switching nello shield nasce dall'esigenza di alimentare il robot con un unico pacco batteria da 6÷12 volt (due celle LiPo oppure 6÷8 celle NiCd o NiMH). Siccome Arduino e i servo funzionano in modo ottimale a 5 V ci occore un regolatore di tensione, ma abbiamo subito scartato il classico 7805 perché è di tipo lineare serie e quindi ha un basso rendimento. Un esempio ci fa vedere il perché: supponiamo di aver bisogno di 1 A sul circuito a 5 volt; alimentando il 7805 a 10 V, esso assorbe 10 W e ne dissipa 5, quindi ha un rendimento del 50 %, assolutamente inaccettabile per un'apparecchiatura alimentata a batteria. La soluzione ottimale è il regolatore switching, perché converte la tensione con rendimento superiore all'80 %. Non è necessario pensare a circuitazioni particolarmente complesse perché in commercio esistono regolatori preassemblati (come il BOB9370, distribuito dalla ditta Futura Elettronica) e integrati DC/DC come l'LM2576-5 della National Semiconductors, che per realizzare un alimentatore switching richiede semplicemente un'induttanza, un diodo ed un condensatore. Il componente è in grado di erogare una corrente massima di 3 A ed accetta in ingresso tensioni comprese tra 4 e 40volt. Dispone, inoltre, di protezioni in temperatura e dai cortocircuiti.

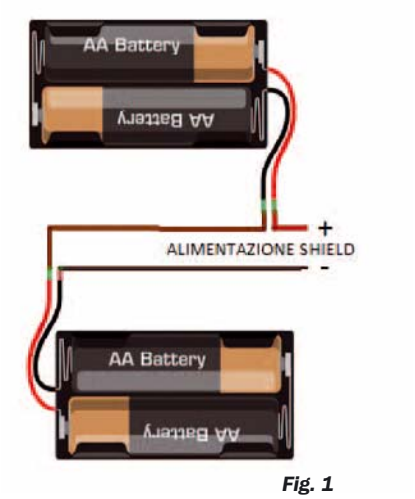
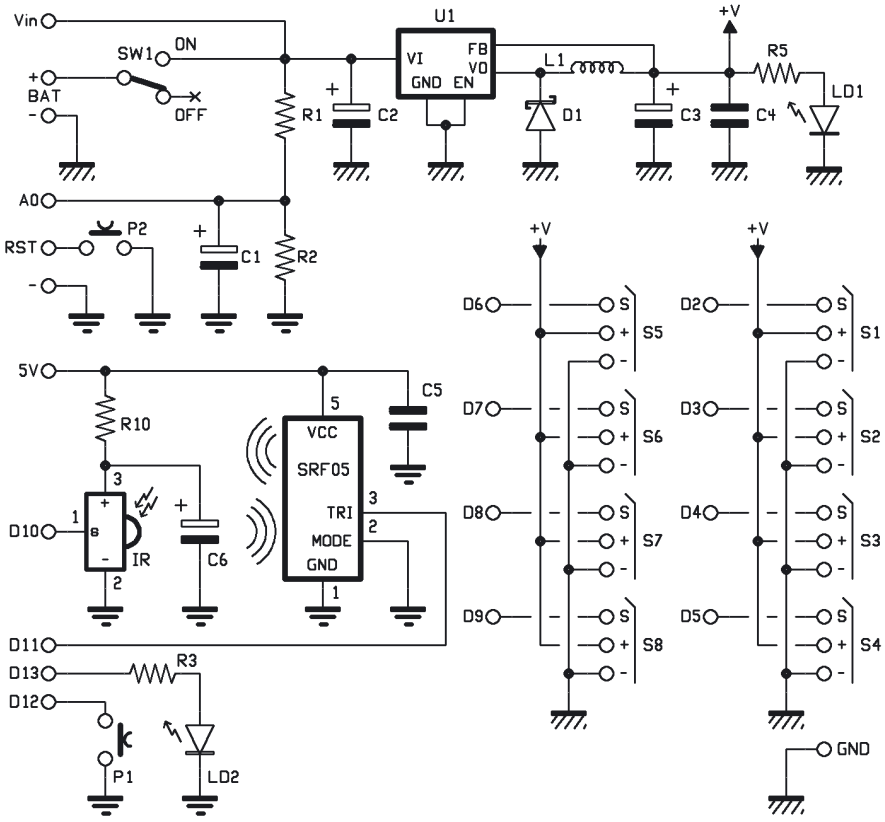


Fig. 1

Schema elettrico



sia compresso fra 0 e 5 V. I valori resistivi di R1 ed R2 sono tali che leggendo la tensione analogica con Arduino, è sufficiente dividere il dato acquisito per 50 per ottenere il valore di tensione in volt. Come sensore di ostacoli abbiamo scelto il sonar SRF05 della Futura Elettronica, la cui forma ricorda due occhi e quindi migliora l'aspetto estetico dei nostri robot. Il modulo comunica ad Arduino i dati sulla distanza rilevata dagli ostacoli, mediante una linea digitale facente capo all'I/O D11. Per quanto riguarda il controllo da remoto tramite telecomando, abbiamo scartato i costosi sistemi via radio per optare per un più economico sistema a raggi infrarossi; è sufficiente installare nella scheda un ricevitore IR integrato compatibile con i normali telecomandi commerciali, come l'integrato PNA4602. Grazie al software che abbiamo preparato, il robot potrà essere telecomandato mediante un normalissimo telecomando di quelli usati per i televisori o video-registratori. Sullo shield troviamo il pulsante per resettare Arduino, un pulsante per uso generico ed un LED connesso al D13 di Arduino. Quanto ai servo, il loro comando avviene mediante i PWM che Arduino fornisce tramite le proprie linee D2÷D9.

REALIZZAZIONE PRATICA

Lo shield è molto semplice da assemblare: per prima cosa procuratevi lo stampato e i componenti, tutti di facile reperibilità; l'induttanza (150 μH) deve essere in grado di sopportare una corrente di 2 A ed avere una resistenza serie non superiore a 0,1ohm. Può essere realizzata avvolgendo 50 spire di rame smaltato da 0,5 mm di diametro su un nucleo toroidale in poliferro (colore giallo-bianco) delle dimensioni di 17,8x9x6,6 mm. Sullo stampato è previsto il posizionamento in verticale ma comunque vi è spazio per posizionarla in orizzontale così da ridurre lo spessore dello shield.
Per l'integrato LM2576-5, lo stampato prevede un piccolo dissipatore ma, se avrete l'accortezza di saldarlo in modo che rimangano alcuni millimetri tra esso e lo stampato, potete ometterlo.
Per montare il sensore SRF05 saldate su di esso un pin-strip e sullo shield disponete uno strip femmina a passo 2,54 mm nel quale poi inserirete il modulo sonar.
Saldate i connettori strip femmina a pin lunghi che vi serviranno ad inserire lo shield in Arduino. Fatto ciò, potete fare un primo collaudo alimentando la scheda tramite il pacco batterie e verificando che si accendano il LED LD1 nonché il LED di alimentazione sulla scheda Arduino.
A questo punto la vostra piattaforma hardware è pronta per essere inserita nella meccanica del robot; per aiutarvi nel compito, vediamo i tre casi uno ad uno.

ROBOT FILIPPO

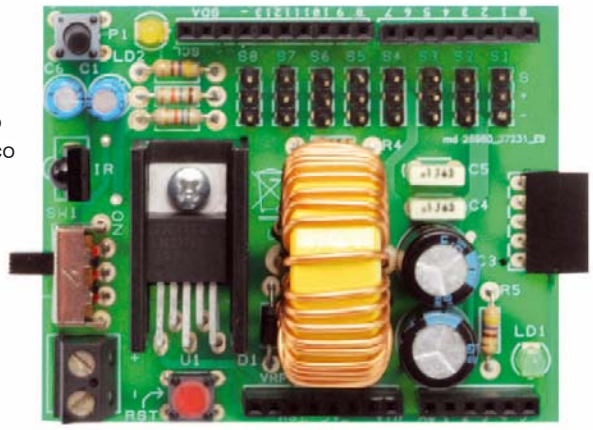
È un robot bipede i cui movimenti sono affidati a due soli servo, quindi è il più semplice di tutti e permette di cimentarsi con la robotica ed imparare l'uso dei servocoman-

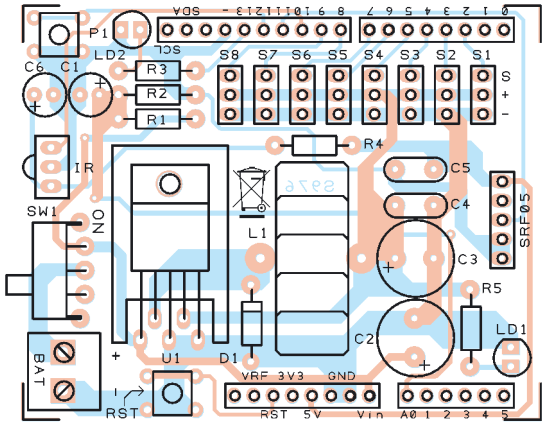
Piano di montaggio

Elenco componenti:

R1: 56 kohm
R2: 18 kohm
R3, R5: 470 ohm
R4: 100 ohm
C1: 10 μF 63 VL elettrolitico
C2: 470 μF 25 VL elettrolitico
C3: 1000 μF 16 VL elettrolitico
C4, C5: 100 nF multistrato
C6: 10 μF 63 VL elettrolitico
LD1: LED 3 mm rosso
LD2: LED 3 mm verde
U1: LM2576-5
SW1: Deviatore a slitta
P1: Microswitch
RST: Microswitch
IR: IR38DM
L1: 100 μH 5A
D1: 1N5819
SRF05: Modulo SRF05

Varie:
- Morsetto 2 poli
- Strip maschio 3 poli (8 pz.)
- Strip maschio 6 poli
- Strip maschio 8 poli (2 pz.)
- Strip maschio 10 poli
- Strip maschio 5 poli 90°
- Strip femmina 5 poli
- Dissipatore TO-220
- Vite 12 mm 3 MA
- Dado 3 MA
- Circuito stampato





di senza spendere cifre importanti. È in grado di camminare e di girare su se stesso, quindi potete dirigerlo in ogni direzione.
Il suo assemblaggio è facilitato in quanto tutti i pezzi si incastrano gli uni negli altri e si possono saldare o incollare (con colla epossidica) per stabilizzare gli incastri. Più delle parole, valgono le foto riportate in queste pagine, che vi fanno capire dove vanno inseriti i vari componenti meccanici (**Fig. 2, 3, 4** e **5**). Terminato l'assemblaggio meccanico, potete fissare l'elettronica (l'insieme di Arduino e del nostro shield) sulla parte superiore, rammentando che il sensore SRF05 deve guardare di fronte al robot. I servo vanno cablati seguendo le indicazioni della **Tabella 1**. Per l'alimentazione è previsto vengano usate 6 o 8 batterie ricaricabili NiMh o NiCd formato stilo, che vanno inserite in due distinti portabatterie e collegate in serie. Se optate per l'utilizzo di 6 celle dovrete usare due portabatterie a tre vani, mentre se usate otto celle dovrete usare due portabatterie da 4 vani. I portabatterie vanno posizionati uno a destra ed uno a sinistra dei due servo, collegandoli in serie con due clip per batterie da 9 volt (**Fig. 1**).
Consigliamo di programmare Arduino prima di collegare i servo, onde evitare che un eventuale programma precedentemente in memoria fornisca segnali che possono portare a movimenti scomposti. Ricordatevi che se alimentate Arduino solo tramite la USB, per programmarla, parte della tensione finirà sullo shield, quindi consigliamo di collegare il cavo USB solo quando è presente anche l'alimentazione principale sui punti BAT; diversamente, potrebbero esserci problemi con i servo.
Programmate Arduino con lo sketch *filippo.pde* ed avviate Serial Monitor, allorché vi ritroverete una schermata con una serie di dati; quelli che vi interessano sono la tensione della batteria e la posizione di neutro dei servo.
Inviate il comando "obs" per avere la lettura del sensore SRF05 ed il comando "lev" per verificare il livello della batteria. Assicuratevi che la tensione letta corrisponda al reale valore delle batterie; per accertare ciò è sufficiente confrontare i dati con una una misura effettuata con un buon tester.
Concentratevi ora sull'impostazione del neutro dei servo, indispensabile in quanto il robot, per funzionare correttamente, deve essere perfettamente centrato, cosa non sempre possibile a livello meccanico, in quanto i braccetti dei servo possono avere solo posizioni predefinite. Per centrare

Tabella 1 - Collegamento dei servo con robot Filippo

Servo	Pin Arduino	Connettore shield	Funzione
Servo 0	2	S1	Inclinazione (servo anteriore)
Servo 1	3	S2	Passo (servo in basso)



Fig. 2

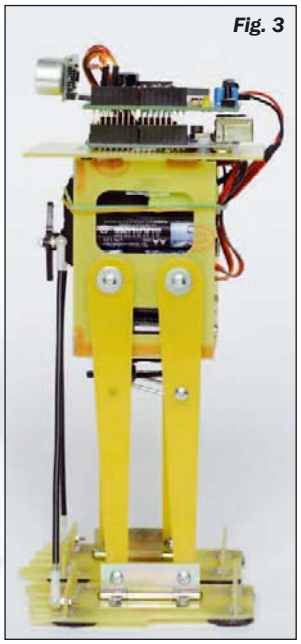


Fig. 3

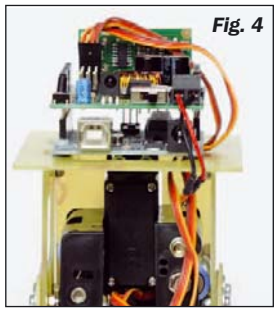


Fig. 4



Fig. 5

Robot bipede Filippo.

il servo che svolge la funzione di inclinare il robot dovete inviare il comando "axx" dove *xx* è la posizione da assegnare. Questo dato può assumere i valori compresi tra 80 e 100, con il punto centrale posto a 90; se non riuscite a centrare il servo rimanendo all'interno di questi valori, dovete spostare fisicamente il braccetto del servo. Agite, se necessario, anche sulle due astine che portano il comando dalla squadretta del servo ai piedi. Potete agire su questo valore qualora riscontrate una tendenza a deviare dalla linea retta durante la camminata. Passate ora al servo del passo ed impostate il neutro con il comando "bxx", dove *xx* è la posizione da assegnare (valore compreso tra 80 e 100).

Adesso siete pronti per il collaudo sul campo: potete usare il telecomando per attivare le funzioni oppure, in mancanza di quest'ultimo, usare il pulsante della shield che però avvia solo la camminata.

Tutte le funzioni hanno una velocità di esecuzione che può essere modificata da telecomando con i tasti prog+ e prog-, mentre molti altri parametri possono essere modificati sul software per adattare il funzionamento alle varie esigenze. Senza considerare l'intero sketch, vediamo ora le parti più interessanti dei programmi e quali parametri possono essere modificati.

Lo sketch inizia includendo le librerie dedicate all'uso dei servo, dei comandi IR e della gestione della EEPROM (**Listato 1**).

La libreria *servo* è già implementata nell'ambiente di sviluppo di Arduino ed è specifica per l'uso dei servocomandi da modellismo. La libreria EEPROM, anch'essa già disponibile, fornisce i comandi per gestire la memoria non volatile dell'ATmega di Arduino; noi la usiamo per salvare la posizione di

Listato 1

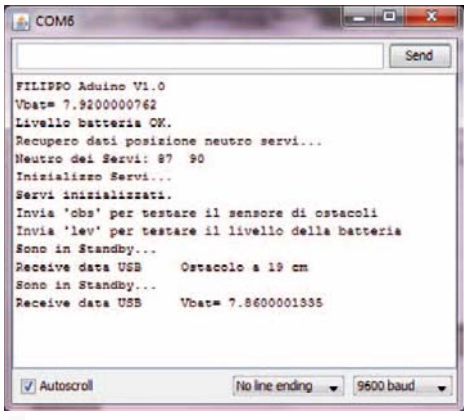
```
#include <Servo.h>
#include <IRremote.h>
#include <EEPROM.h>
```

neutro dei servo, in modo da non doverla settare ad ogni accensione. La terza libreria, *IRremote*, ci serve per gestire i comandi da telecomando e deve esse scaricata dal sito di riferimento *http://github.com/shirriff/Arduino-IRremote*; per il suo funzionamento è sufficiente specificare a quale pin è connesso il sensore di ricezione dei segnali IR. Per sapere se sono arrivati dati dal sensore si deve testare lo stato della variabile **irrecv.decode(&results)**; il codice arrivato è contenuto nella variabile **results.value**. Per sapere a quale tasto corrisponde il codice arrivato bisogna confrontare quest'ultimo con il codice di riferimento; allo scopo specifichiamo all'inizio dello sketch i codici relativi a ciascun pulsante (**Fig. 6 e 7**). Come potete vedere nel **Listato 2**, ad ogni tasto sono associati due codici (CODE1 e CODE2) per distinguere la pressione momentanea da quella prolungata: nel primo caso i codici vengono inviati uno all'attivazione ed uno al rilascio, mentre nel secondo il trasmettitore ripete il codice corrispondente alla pres-



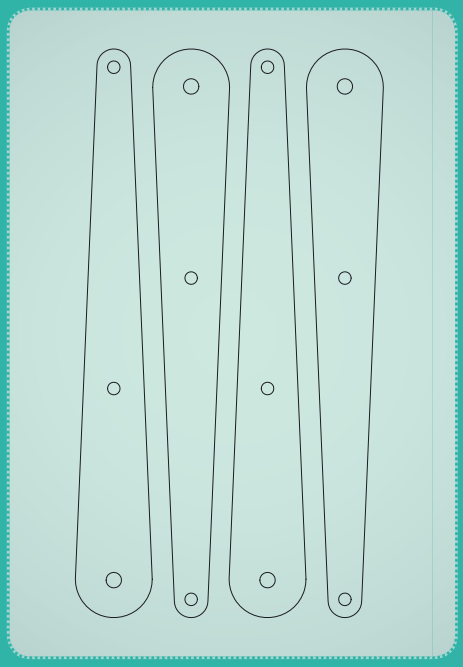
Fig. 6 Le funzioni dei tasti del telecomando Philips.

Fig. 7 - Schermata Serial monitor per il robot Filippo.



E la meccanica dei robot?

Se avete già realizzato uno dei robot Filippo, Bipe, Spiderin, il progetto di queste pagine sarà per voi un ottimo plug-in da realizzare al volo e sostituire alle schede di controllo originarie.
Se non avete in casa uno di tali robot ma leggendo questo articolo siete stati stuzzicati dall'idea di costruirvene uno, niente paura: trovate tutti i disegni per realizzare le parti e le foto che vi guideranno nell'assemblaggio, sul nostro sito Internet *www.elettronicain.it*. Inoltre, dei robot Spiderin e Bipe trovate i dettagli e la descrizione, con tanto di particolari meccanici, negli articoli loro dedicati nei fascicoli, rispettivamente, n° 143 e n° 133; per quanto riguarda Filippo, ne abbiamo parlato nei fascicoli n° 78 e n° 79. Siccome di quest'ultimo non abbiamo pubblicato i disegni meccanici per realizzare le parti che lo compongono, provvediamo in questo riquadro, dove li vedete in scala 1:2 (ossia grandi la metà): il disegno a sinistra riguarda le gambe, mentre la parte a destra riporta le parti del corpo e dei piedi, che vanno fissati alle gambe mediante cerniere da mobili e viti con dado autobloccante che permettono al piede di flettere in avanti e indietro. Chi volesse cimentarsi con i robot ma non è portato per l'autocostruzione, potrà acquistare la meccanica già pronta, presso Futura Elettronica (*www.futurashop.it*) e montarla solo l'elettronica, disponibile in kit di montaggio sempre da Futura Elettronica.





sione. Quindi se si preme il pulsante Play viene inviato ciclicamente il codice 0x81C, mentre al rilascio viene inviato il codice 0x1C.
Lo sketch è previsto per funzionare con telecomandi che usano la codifica Philips, che la più usata nei televisori; se usate un trasmettitore di marca differente i codici dei pulsanti non corrispondono e dovrete modificare lo sketch, inserendo i codici che Serial Monitor mostra sullo schermo del PC quando trasmettete col vostro telecomando (**Fig. 8** e **Fig. 9**). Ad esempio, se utilizzate un telecomando LG noterete che il codice inviato è unico indipendentemente da come premete i pulsanti. Volendo, ad esempio, assegnare il pulsante "freccia in su" della navigazione a menù alla funzione di camminata, non dovrete fare altro che premerlo, leggere il codice IR ricevuto e trascriverlo nella riga dello sketch che assegna il codice per la camminata:

```
#define WALK_CODE1 0x20DF02FD
#define WALK_CODE2 0x20DF02FD
```

I parametri che possono essere modificati da programma per adattare i movimenti sono:

```
int TimeOneStep = 2000;
int AmpPasso = 30;
```

Listato 2

```
#define STNB_CODE1 0x81D   // Stop Philips TV remote (AZIONE STOP)
#define STNB_CODE2 0x1D    // Codice alternativo
#define WALK_CODE1 0x81C   // Play Philips TV remote (AZIONE CAMMINA)
#define WALK_CODE2 0x1C    // Codice alternativo
#define SPEEDUP_CODE1 0x820 // Prg+ Philips TV remote (AZIONE Più VELOCE)
#define SPEEDUP_CODE2 0x20  // Codice alternativo
#define SPEEDDW_CODE1 0x821 // Prg- Philips TV remote (AZIONE Più LENTA))
#define SPEEDDW_CODE2 0x21  // Codice alternativo
#define LEFT_CODE1   0x2C  // Left Philips remote (RUOTA SX)
#define LEFT_CODE2   0x82C  // Codice alternativo
#define RIGHT_CODE1   0x2B  // Right Philips remote (ROUTA DX)
#define RIGHT_CODE2   0x82B  // Codice alternativo
```

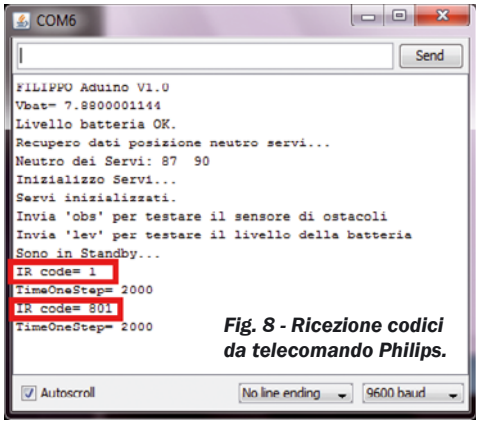
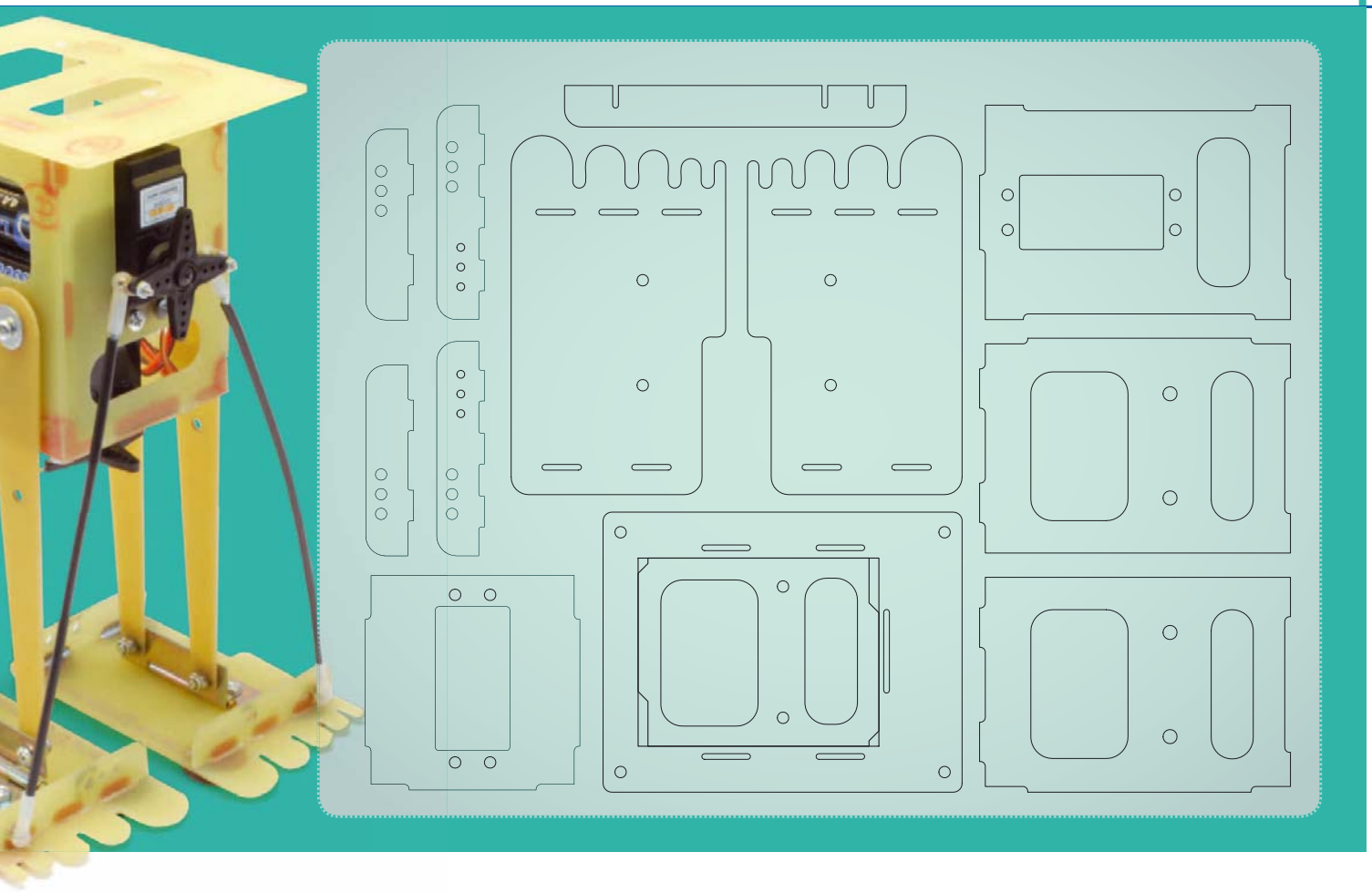


Fig. 8 - Ricezione codici da telecomando Philips.



```
int IncPasso = 15;
int AmpRuota = 30;
int IncRuota = 18;
```

TimeOneStep è il valore iniziale del tempo impiegato per eseguire un passo, può essere modificato da telecomando da un minimo di 1 ad un massimo di 4 secondi.
AmpPasso è la massima ampiezza (in gradi) dell'angolo formato dalle gambe durante la camminata; il suo valore può variare da un minimo di 5 ad un massimo di 40.
IncPasso è la massima inclinazione del robot durante la camminata; il suo valore può variare da un minimo di 5 ad un massimo di 20 e deve garantire al robot di rimanere in equilibrio su una gamba mentre compie un passo.
AmpRuota è la massima ampiezza in gradi dell'angolo formato dalle gambe durante la rotazione; il suo valore può variare da un minimo di 5 ad un massimo di 40.
IncRuota è la massima inclinazione del robot durante la rotazione; il suo valore può variare da un minimo di 5 ad un massimo di 20.

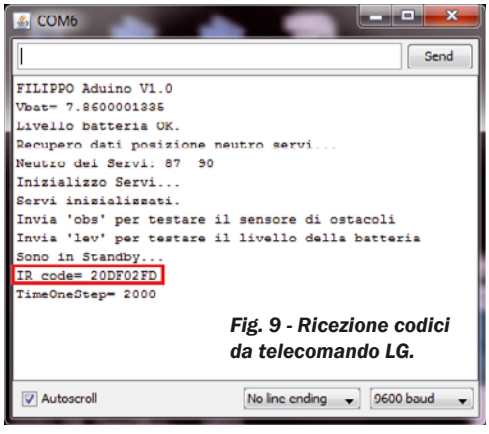


Fig. 9 - Ricezione codici da telecomando LG.

ROBOT BIPE

Veniamo ora al Bipe, un robot bipede di cui abbiamo parlato nel fascicolo n° 133; in esso il nostro blocco Arduino+shield sostituirà la scheda di controllo basata su PIC16F876. La meccanica è realizzata in vetronite ramata e l'assemblaggio avviene saldando tra loro i vari componenti per garantire la robustezza necessaria a reggere il peso dei componenti. Per chi avesse dei dubbi sull'assemblaggio, consigliamo di leggere l'articolo sulla rivista n° 133.
L'elettronica Arduino-based va posizionata sul dorso del robot, mentre, il pacco batterie deve stare sulla parte anteriore. Diciamo subito che per le nostre prove abbiamo utilizzato una batteria composta da due celle LiPo da 850 mah, che, essendo piccole e leggere, trovano facilmente posto e non appesantiscono la struttura.
Per posizionare correttamente il sensore SRF05 dovete utilizzare un piccolo adattatore realizzato con uno strip femmina ad angolo retto e pin lunghi.
Il collegamento dei servo alla scheda va fatta seguendo le indicazione della **Tabella 2**. Per la messa in

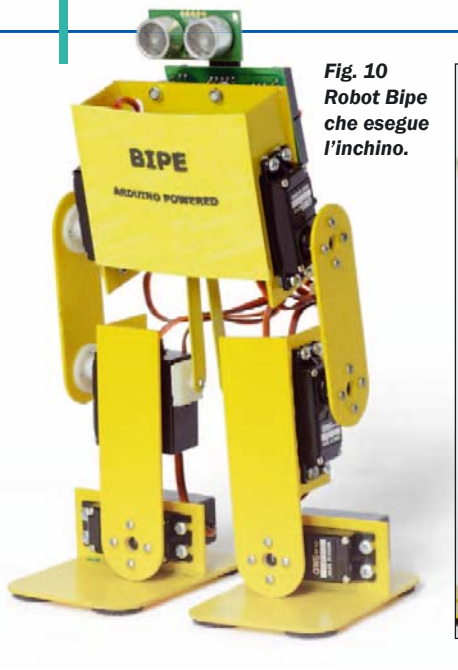


Fig. 10 Robot Bipe che esegue l'inchino.



Fig. 11 Posizione di equilibrio.



Fig. 12 Particolare dei collegamenti dei servi sullo shield.

Tabella 2 - Collegamento dei servo con robot Bipe

Servo	Pin Arduino	Connettore shield	Funzione
Servo 0	2	S1	anca sinistra
Servo 1	3	S2	ginocchio sinistro
Servo 2	4	S3	piede sinistro
Servo 3	5	S4	anca destra
Servo 4	6	S5	ginocchio destro
Servo 5	7	S6	piede destro

servizio valgono le stesse indicazioni riportate per il robot Filippo.
Lo sketch da caricare in Arduino si chiama *bipe.pde* e mantiene la stessa struttura del software utilizzato con Filippo, con le dovute modifiche e aggiunte.
Per questo robot è importante la taratura del neutro dei servo, che avviene come descritto per Filippo. Tramite Serial Monitor si deve inviare il comando composto dalla lettera del servo da settare, seguito dal valore che si vuole impostare (compreso tra 80 e 100). Al Servo0 corrisponde la lettera "a" ed al Servo5 la "f"; gli altri vanno di conseguenza. Se il comando è andato a buon fine, su Serial Monitor ne avrete conferma; ad esempio, "set servo: a to 90". Dopo che avrete messo in funzione il robot, probabilmente dovrete ritoccare la posizione di neutro: ad esempio, se il robot tende a cadere in avanti o indietro, oppure se durante la camminata tende a deviare dalla linea retta. Dalle foto riportate nell'articolo noterete che il dorso è leggermente piegato in avanti nell'intento di bilanciare i pesi. A lavoro ultimato usate del nastro adesivo e delle fascette per mettere in ordine e raggruppare i cavi dei servocomandi, così il robot avrà un aspetto più ordinato. Nel Bipe sono previste ulteriori funzioni come l'inchino ed il calcio (**Listato 3**).
Al pulsante **1** del telecomando corrisponde la funzione calcio con la quale il robot può colpire, ad esempio una pallina; al pulsante **2** corrisponde l'inchino (**Fig. 10**). Vi consigliamo di guardare il video associato a questo progetto per meglio capire come si comportano i robot e quali movimenti sono possibili.
I parametri settabili da programma sono simili a quelli del robot Filippo:

```
int TimeOneStep = 4000;
int AmpPasso = 10;
int AmpRuota = 12;
int IncPasso = 8;
```

La modifica di questi parametri è associata alla costruzione meccanica (dipende ad esempio dalla ripartizione del peso delle batterie). Essendo il baricentro quasi tutto spostato verso l'alto, la struttura è comunque instabile e tende a cadere; i grandi piedi ed un movimento lento favoriscono la stabilità ma, se aumentate la velocità di esecuzione i movimenti diventano più bruschi ed il pericolo di cadute aumenta.

Listato 3

```
#define PRG1_CODE1 0x801 // Program 1
#define PRG1_CODE2 0x1 // Program 1
#define PRG2_CODE1 0x802 // Program 2
#define PRG2_CODE2 0x2 // Program 2
```

ROBOT SPIDERIN

Veniamo ora al terzo ed ultimo robot, descritto per la prima volta nel n° 143: somiglia ad un ragno, che per semplificare la meccanica (e ridurre il costo di realizzazione) è provvisto di quattro sole zampe. Il movimento viene governato da due servo per zampa, che comunque

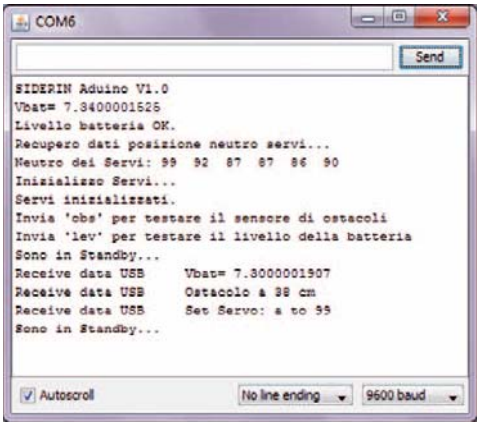


Fig. 13 - Schermata Serial Monitor per i robot Bipe e Spiderin.

assicurano un buon risultato.
Il maggior numero di servo rispetto al robot Bipe non deve spaventare, perché Spiderin è il più semplice da realizzare ed ha una stabilità intrinseca (durante i movimenti, tre zampe rimangono sempre a contatto con il suolo mentre una sola si muove, alzandosi).
Per questo robot non abbiamo previsto la regolazione del neutro dei servo, perché alcune imprecisioni non vanno ad influenzarne il funzionamento. L'unica accortezza è fissare il pacco batterie in modo che il robot presenti il baricentro esattamente al centro.
La meccanica è realizzata in vetronite, che consigliamo di verniciare di nero per migliorarne l'aspetto esteriore. Non sono necessarie saldature ed ogni pezzo va semplicemente avvitato ai servo, che contribuiscono a formare la meccanica; trovate tutti i dettagli nel fascicolo n° 143 di Elettronica In.
Suggeriamo, per prima cosa, di programmate la scheda Arduino con il programma *spiderin.pde* e successivamente collegare elettricamente i servo, che così facendo saranno tutti posizionati esattamente al centro. Successivamente collegherete meccanicamente i servo nella loro giusta posizione. Assicuratevi che essi siano allineati e che le zampe non siano né troppo aperte, né troppo chiuse (riferitevi alle foto di **Fig. 14** e **Fig. 15**). Per quanto riguarda le connessioni dei servo con lo shield, fate riferimento alla **Tabella 3**.
All'accensione, dopo un secondo (trascorso il warm-Up dell'alimentatore switching) tutti i servo saranno posizionati nel punto di neutro; potrebbe capitare che qualche servo vibri leggermente perché il meccanismo interno non ha ancora trovato il punto centrale.

Tabella 3 - Collegamento dei servo con robot Spiderin.

Servo	Pin Arduino	Connettore shield	Funzione
Servo 0	2	S1	Angolazione zampa anteriore destra
Servo 1	3	S2	Elevazione zampa anteriore destra
Servo 2	4	S3	Angolazione zampa posteriore destra
Servo 3	5	S4	Elevazione zampa posteriore destra
Servo 4	6	S5	Angolazione zampa posteriore sinistra
Servo 5	7	S6	Elevazione zampa posteriore sinistra
Servo 6	8	S7	Angolazione zampa anteriore sinistra
Servo 7	9	S8	Elevazione zampa anteriore sinistra

Ebbene, alle volte è sufficiente un piccolo colpo al servo per sistemare le cose.
Per questo robot non abbiamo previsto parametri da modificare, in quanto i movimenti sono molti e molto complessi e ci sarebbero troppe variabili da settare; abbiamo preferito trovare un setup che potesse offrire il miglior funzionamento in ogni situazione. Per il nostro prototipo la batteria LiPo che lo alimenta è stata fissata sotto la struttura, proprio in mezzo ai servo, assicurando, oltre al centraggio dei pesi, un baricentro molto basso.
Giusto per verificare se le ipotesi iniziali erano corrette, abbiamo eseguito alcune misure sulle correnti in gioco, visto che si usano otto servo: durante la camminata la corrente media assorbita dai servo è stata di 1,6 A, mentre quella erogata dalla batteria si è attestata su 1 A. Con una tensione sulla batteria di 8,4 volt, la potenza erogata è di 8,4 watt, mentre quella dissipata dai servo è di 8 W.
Per il miglior funzionamento di questi robot, consigliamo di posizionare sotto i piedi dei feltrini tipo quelli usati sulle sedie; la superficie ottimale di movimento è un piano liscio.

Listato 4

```
long readDistance()
{
  long duration, cm;
  pinMode(PING_PIN, OUTPUT);
  digitalWrite(PING_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(PING_PIN, HIGH);
  delayMicroseconds(5);
  digitalWrite(PING_PIN, LOW);
  pinMode(PING_PIN, INPUT);
  duration = pulseIn(PING_PIN, HIGH);
  // ritorna distanza ostacolo in cm
  return duration / 29 / 2;
}
```

IL FIRMWARE

Ora analizziamo il firmware, descrivendo le funzioni comuni a tutti i robot, come la procedura che legge la distanza dagli ostacoli tramite il sensore SRF05 (**Listato 4**).
La procedura prevede di inviare un

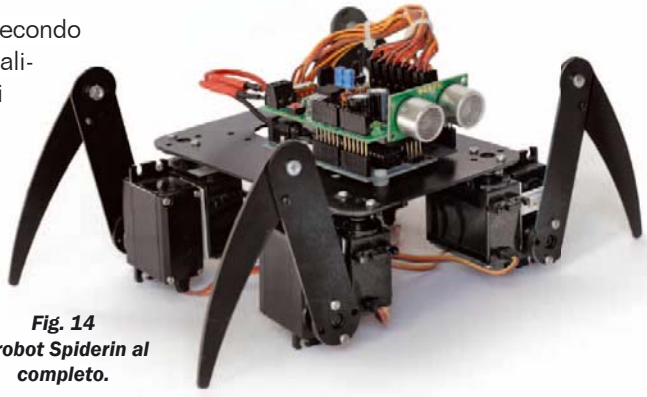
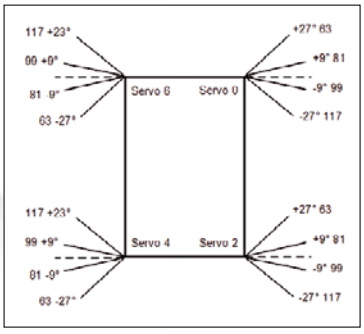


Fig. 14 Il robot Spiderin al completo.

Fig. 15 - Angolazioni raggiunte dalle zampe nelle varie fasi della camminata.



Guarda i nostri tre robot in azione su YouTube



Vuoi vedere come si muovono i nostri tre robot controllati da Arduino? Nulla di più semplice! Abbiamo realizzato e postato un video su YouTube all'indirizzo *http://youtu.be/-_NkdFeDG_I* che illustra le capacità di movimento di questi simpatici automi.

impulso della durata di 2 µs. verso il sensore, il quale procede generando un treno di ultrasuoni. Lo stesso pin viene poi usato come ingresso per leggere, tramite la funzione **PulseIn**, il tempo impiegato dalle vibrazioni acustiche a raggiungere l'ostacolo, rimbalzare e tornare al sensore, periodo durante il quale il sensore mantiene alto il livello della linea. Il tempo impiegato dagli ultrasuoni per andare e ritornare al sensore è proporzionale alla distanza dell'oggetto.

Ora passiamo alla lettura della tensione sulla batteria (**Listato 5**): si tratta di leggere il canale analogico A0 di Arduino e salvare il dato nella variabile **VbatRAW**, quindi dividerne il valore per 50; così si ottiene la tensione in volt della batteria. Per evitare l'eccessiva scarica abbiamo previsto una protezione che, al di sotto di una certa tensione, toglie tensione a tutti i servo; il consumo in corrente è quindi ridotto al minimo ma in tempi brevi è opportuno scollegare e ricaricare le batterie.

La soglia è 6 volt, ideale per un

Listato 5

```
void levelBat()
{
  int VbatRAW = analogRead(A0);
  float Vbat = float(VbatRAW)/50;
  Serial.print("Vbat= ");
  Serial.println(Vbat, DEC);
  if (Vbat<6.0)  {
   Serial.println("Livello batteria basso!");
    inOff();
  }
}
```


VOLUME SPECIALE
"Primi passi con Arduino"

Pensato per i principianti che si avvicinano all'elettronica e per gli hobbisti che non hanno mai affrontato la programmazione.



Acquistalo su
www.futurashop.it

Via Adige, 11 - 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

FUTURA ELETTRONICA

Listato 6

```
void initServo()
{
  // posizione iniziale dei servi
  servo0.attach(2);  // connette servo
  delay(300);
.....
  servo7.attach(9);  // connette servo
  delay(300);
}
```

pacco batterie composto da due celle LiPo o sei NiMH/NiCd; per pacchi da 8 di queste ultime potete portare la soglia ad 8 volt. Un'altra procedura comune a tutti i robot si chiama **initServo()** e provvede ad inizializzare i servo (**Listato 6**).
La funzione **servo.attach()** associa ciascun oggetto servo alla rispettiva uscita; da questo momento al servo giunge il segnale PWM di comando. I ritardi tra un'associazione e la successiva sono necessari per evitare pericolosi picchi di corrente che si produrrebbero se tutti i servo entrassero in azione contemporaneamente.

Listato 7

```
void inOff()
{
  leg[0] = 90;
.....
  leg[7] = 90;
  setServo();
  servo0.detach();  // connette servo
....
  servo7.detach();  // connette servo
  RobotMode = MODE_OFF;
}
```

La procedura **inOff()** svolge la funzione opposta: rimuove l'associazione dei servo ai pin e disattiva i servo, che perciò assorbono il minimo di corrente (**Listato 7**).
La procedura che mette in posizione di riposo il robot si chiama **Standby()** e semplicemente posiziona tutti i servo al centro (**Listato 8**). Nel caso dei robot Filippo e Bipe, il centro corrisponde al punto di neutro i cui valori sono contenuti nel vettore di nome **leg_ntr[]**.

Listato 8

```
void inStandby()
{
  leg[0] = 90;
....
  leg[7] = 90;
  setServo();
  RobotMode = MODE_STANDBY;
  Serial.println("Sono in Standby...");
}
```

L'ultima procedura che descriviamo è quella che provvede a posizionare i servo sulle coordinate desiderate (**Listato 9**); il raggiungimento della posizione è graduale e avviene in un tempo che possiamo definire a piacimento. Se impostassimo direttamente la coordinata da raggiungere, il servo, avrebbe un movimento brusco ed il robot si muoverebbe a scatti. Per ottenere lo spostamento graduale è necessario definire il vettore **leg[]** che contiene le coordinate da raggiungere da ciascun servo ed un vettore **leg_old[]** che contiene le coordinate attuali.
La procedura comanda i servo inviando loro tutte le coordinate intermedie dal valore attuale al valore da raggiungere in un tempo definibile tramite la variabile **TimeOneStep**. Solo per il robot spider è prevista una procedura suppletiva denominata **setServoWalk()** perché è necessario che alcuni servo si muovano a velocità diverse rispetto agli altri. Durante la camminata, le tre zampe a contatto con il pavimento si muovono lentamente, mentre la quarta zampa deve sollevarsi e muoversi velocemente nel senso opposto all'avanzamento. Notate che per ciascuna procedura di movimento è prevista una procedura di preparazione perché, inizialmente, il robot parte da fermo e solo successivamente è possibile richiamare ricorsivamente la procedura di camminata o rotazione. ☐

Listato 9

```
void setServo()
{
  for (i=0; i<18; i++) {
     servo0.write(leg_old[0] + i*(leg[0]-leg_old[0])/18);
     servo1.write(leg_old[1] + i*(leg[1]-leg_old[1])/18);
     servo2.write(leg_old[2] + i*(leg[2]-leg_old[2])/18);
     servo3.write(leg_old[3] + i*(leg[3]-leg_old[3])/18);
     servo4.write(leg_old[4] + i*(leg[4]-leg_old[4])/18);
     servo5.write(leg_old[5] + i*(leg[5]-leg_old[5])/18);
     servo6.write(leg_old[6] + i*(leg[6]-leg_old[6])/18);
     servo7.write(leg_old[7] + i*(leg[7]-leg_old[7])/18);
     delay(TimeOneStep/4/18);
   }
   leg_old[0]=leg[0];
   leg_old[1]=leg[1];
   leg_old[2]=leg[2];
   leg_old[3]=leg[3];
   leg_old[4]=leg[4];
   leg_old[5]=leg[5];
   leg_old[6]=leg[6];
   leg_old[7]=leg[7];
}
```



per il MATERIALE

Tutti i componenti utilizzati in questi progetti sono facilmente reperibili in commercio mentre sia gli stampati che gli sketch sono scaricabili gratuitamente dal sito della rivista (www.elettronicain.it). La scatola di montaggio dello shield descritto in questo articolo costa 18,00 Euro (codice ROBOT_SHIELD) mentre il sensore ad ultrasuoni SRF05 costa 26,00 Euro.
Per completare l'elettronica è necessaria una board Arduino Uno (Euro 24,50). La meccanica -completa di servo- del robot ARDUSPIDERIN costa 105,00 Euro, la meccanica del robot ARDUFILIPPO costa 68,00 Euro, infine la meccanica del robot ARDUBIPE costa 85,00 Euro.
Le scatole di montaggio non comprendono le batterie.
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

Sistemi di azionamento e controllo per ROBOTICA



DRIVER PER MOTORI



RDB10507 € 82,00

Driver quadruplo per motori passo-passo bipolari • 2A

Controlla fino a 4 motori passo-passo con tensione fino a 35V e corrente massima di 2A.



DRIVER24V20 € 66,50

Driver per motore DC • 20A

Controlla in modo bidirezionale un motore DC con tensione fino a 40Vdc e corrente massima di 20A.



JNRF21V3USB € 56,00

Driver con feedback per motore DC • 3A

Controlla in modo bidirezionale un motore DC con tensione fino a 28Vdc e corrente massima di 3A.



MD22 € 90,00

Driver per 2 motori DC • 5A

Controlla in modo indipendente due motori DC con tensione fino a 24Vdc e corrente massima di 5A.



D3NIC12 € 28,00

Micro driver seriale per 2 motori DC • 1A (in kit)

Controlla in modo indipendente due motori DC con tensione fino a 25V e corrente massima di 1A.

RUOTE / BALLCASTER / SUPPORTI

Ruota Ø120x60 mm

RUOTA120X60 € 19,00

Pneumatico in gomma morbida ideale per terreni accidentati. Adattatore per il montaggio su alberi motore da 4mm incluso.





Set 2 ruote Tamiya Sport

SET2RUOTE € 9,90

Diametro esterno 56mm; larghezza pneumatico 25mm. Complete di 2 mozzi per alberi tondi da 4mm e 2 per alberi esagonali da 3mm.

Coppia di ruote Pololu Ø43mm

Diametro esterno 42mm; larghezza del pneumatico 19mm. Compatibili con motoriduttore MMG298 ed encoder ENCODER4219.

RP4219 € 8,00





Ruota per motoriduttori EMG30

WHEEL100 € 17,00

Diametro esterno 100mm, mozzo 5mm (completo di grano).

Set 4 ruote dentate

Costituito da 4 robuste ruote a 8 denti in ABS nero. Ideali per realizzare cingoli per piccoli robot.

IMSS4 € 16,00



Set 10 maglie per cingoli

MTLP10 € 10,50

Realizzate in ABS, sono componibili tramite piccoli perni in acciaio inox. Ideali per realizzare piccoli robot. Dimensioni con 10 maglie 22,86x135mm.

MOTORIDUTTORI



Motoriduttori 12Vdc

Dotati di soppressori disturbi su collettore, sono particolarmente indicati per la realizzazione di Robot. Direzione di rotazione secondo polarità.

da € 12,00

Codice	Tensione/ Consumo max.	Rapporto di riduzione	Coppia massima	Velocità coppia max / a vuoto	Dimensioni (Ø x L totale)	Peso	Encoder	Euro
RH15812630	12Vdc/270mA	629,82:1	100 N/cm	9 rpm/10 rpm	39,6 x 72mm	190 g	-	22,50
RH15812200	12Vdc/580mA	198,5:1	100 N/cm	23 rpm/33 rpm	39,6 x 69mm	190 g	-	21,00
RH1591275	12Vdc/230mA	76,84:1	30 N/cm	28 rpm/43rpm	39,6 x 66,5mm	190 g	-	20,00
RH15912630	12Vdc/130mA	629,82:1	100 N/cm	4,5 rpm/5rpm	39,6 x 72mm	190g	-	22,50
LE1491243	12Vdc/50mA	43,3:1	3,8 N/cm	40 rpm/60 rpm	27 x 67mm	55 g	effetto Hall (3 imp./giro)	20,00
L1491243	12Vdc/50mA	43,3:1	3,8 N/cm	40 rpm/60rpm	27 x 57,2mm	55 g	-	12,00
EMG30	12Vdc/530 mA	30:1	1,5 kg/cm	170 rpm/216 rpm	30 x 86,6mm	140 g	effetto Hall (360 imp./giro)	49,50
E1921225	12Vdc/1750 mA	25:1	90 N/cm	105 rpm/160 rpm	40,5 x 123,4mm	450 g	-	57,50

Micro motoriduttore in metallo • 6V / 45 RPM

Rapporto di riduzione 298:1; coppia max. 1,8kg/cm; alberino da 9,27mm (Ø3mm); tensione nominale 6Vdc; consumo a vuoto 30mA; dimensioni 26 x 10 x 12mm (alberino escluso).

MMG298 € 20,00



Motoriduttore in metallo • 6V / 130 RPM

Rapporto di riduzione 75:1; coppia max. 9,4kg/cm; alberino in metallo da 8mm (Ø4mm); tensione nominale 6Vdc; velocità 130rpm; consumo a vuoto 400mA; dimensioni Ø25 x 54mm.

MG25054L € 25,00



Ballcaster

Utilizzabile come terzo punto d'appoggio nei robot. Disponibile con sfera in metallo o in plastica.

da € 3,10

Codice	Tipo sfera	Diametro	Altezze date dai distanziali	Pz x conf.	Euro
PLASTBCASTER	plastica	Ø 9,4mm	-	3	3,10
PBCASTER1	plastica	Ø 25,4mm	28mm e 35,5mm	1	8,50
PBCASTER38	metallo	Ø 10mm	10mm e 15mm	1	3,90
PBCASTER12	metallo	Ø 12mm	12,7mm e 20mm	1	4,70
BCASTER	metallo	Ø 13mm	25mm e 35mm	2	8,90
PBCASTER34	metallo	Ø 18mm	20mm e 28mm	1	6,50

Mozzo in alluminio per accoppiamento asse motore e ruota • 2 Pz

Compatibile con alberi motore rotondi, a "D" ed esagonali. Fori filettati per il montaggio di ruote o meccanismi.



Codice	Rif.	Foro centrale	Dimensioni in mm	Bloccaggio	Euro
HUB3MM	1	Ø3mm	Ø17.46 x 5.08	1 grano	7,00
HUB4MM	1	Ø4mm	Ø19.05 x 5.08	1 grano	8,00
HUB6MM	2	Ø6mm	Ø12.07 x 4.13 Ø25.4 x 5.08	2 grani	9,50

Supporto per motoriduttori GM2, BCM601 e MOTRID1201

Dimensioni 22,5 x 22,6 x 12,7mm.

GM2SSUPP € 2,50



Supporto per motoriduttori GM3 e GM9

Dimensioni: 22,1 x 17,8 x 12,7mm; supporto in metallo.

GM3SSUPP € 2,90



Coppia di supporti per MMG298

Completi di 4 viti e 4 dadi.

MMGSUPP € 5,50



Coppia di supporti per MMG298 e ENCODERW4219

Completi di 4 viti e 4 dadi.

MMGSUPP1 € 6,50



Coppia staffe di supporto per motoriduttori

Per motoriduttori serie RH158 e RH159. Corpo in alluminio; spessore 1,59mm; dimensioni 50,8 x 39,2 x 37,59mm.

SUPP370 € 9,90



Staffa di supporto per EMG30

Corpo in alluminio; spessore 2 mm; dimensioni 74 x 25 x 45mm.

EMG30SUPP € 6,00



Supporti per RPM2

Ideale per montaggio su circuito stampato.

CLIPRPM2 € 1,00



Prezzi IVA inclusa

FUTURA ELETTRONICA idee in elettronica
www.futurashop.it
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Maggiori informazioni su questi prodotti e su altri dispositivi per robotica sono disponibili sul sito www.futurashop.it tramite il quale è anche possibile effettuare acquisti on-line.



UN PROGRAMMATORE HV MODULARE PER MCU Atmel

Analizziamo la sezione "Keypad & Display" e spieghiamo come assemblare l'insieme ed inserirlo in un contenitore adatto.
Seconda parte.



del prof. MICHELE MENNITI

Finora vi abbiamo presentato le prime due sezioni hardware del nostro Programmatore HV (*Master e Socket*), illustrando una serie di possibili combinazioni modulari di comunicazione seriale e di alimentazione. In queste pagine completiamo la descrizione dell'hardware, analizzando una sezione, che abbiamo denominato "*Keypad & Display*", la quale ci permetterà di renderlo completamente stand-alone, quindi indipendente dal PC. Infatti, una volta programmati i due microcontrollori ATMEL ATmega328P della sezione *Master* e della *Keypad & Display* con i firmware che vi forniremo nel nostro sito web, il keypad esadecimale ed il display 4x20 (4 righe da 20 caratteri ciascuna) vi permetteranno di utilizzare tutte le funzioni dello strumento. Dunque, potremo montare il programmatore in un contenitore adatto, realizzando uno strumento professionale che farà bella mostra di sè nel nostro laboratorio. Chi intendesse realizzare il progetto stand-alone dovrà decidere quali parti, tra quelle descritte nel fascicolo n° 165, montare sul PCB Master:

- Master minimal; componenti indispensabili per il funzionamento dello strumento e per il collegamento con gli altri PCB (ATmega328P con componenti di contorno, circuito switch 12V, connettori a 2, 6 e 26 pin, jumper);
- Convertitore USB-Seriale interno (è formato dai componenti che sul PCB Master sono racchiusi nel riquadro "USB→Serial");
- Alimentazione.

Per quanto riguarda il convertitore USB-Seriale, potete montarlo se volete programmare direttamente l'ATmega328P Master; se

CARATTERISTICHE TECNICHE DEL PROGRAMMATORE HV

- **Alimentazione: 5V**
- **Consumo max: 300mA**
- **Visualizzazione dati: Display LCD 4x20 retroilluminato**
- **Inserimento dati: Keypad esadecimale a matrice**
- **MCU programmabili: famiglie ATMEL in case PDIP con pin 40W, 28, 20, 14, 8.**
- **Tipi di programmazione: HVPP (con variante per tiny) e HVSP**
- **Operazioni di lettura: Lock bits, Fuse bits, Signature**
- **Operazioni di scrittura: Lock bits, Fuse bits (valori di default o personalizzati), Chip Erase**
- **Inserimento dei valori di scrittura direttamente in formato esadecimale.**
- **Indicazione automatica modello di MCU sotto test e selezione automatica del tipo di programmazione specifica**

invece disponete già di un qualsiasi convertitore USB-Seriale esterno, questa sezione non serve. Se intendete programmare i due microcontrollori (Master e K&D) mediante la tecnica ISP (che abbiamo trattato nei numeri 159, 160 e sul 164, dove abbiamo presentato anche il Programmatore ISP), allora non vi serve alcun tipo di convertitore, né interno e né esterno.

Quanto all'alimentazione, servono 5 V per le tre schede e 12 V per la programmazione HV; per lo stand-alone, l'una e l'altra possono essere fornite da due alimentatori esterni, ma noi vi consigliamo di montare il convertitore step-up interno (sul PCB Master, i relativi componenti sono quelli racchiusi nel riquadro "5V→12V"), visto che il costo dei componenti è anche inferiore a quello di un alimentatore esterno a 12 V e che tale opzione è molto più comoda, dovendo ricorrere al solo alimentatore esterno a 5 V per l'intero strumento. In questo caso non servirà più montare il connettore a due pin previsto per l'alimentazione esterna a 12 V, ma servirà quello previsto per l'interruttore di alimentazione, posto in serie alla linea esterna dei 5 V, che interrompe l'alimentazione di tutto il circuito. Naturalmente, se disponete di un alimentatore con interruttore integrato, al posto di questo connettore potete saldare un ponticello con un filo di rame di adeguato spessore.

Oltre al circuito *Master*, bisognerà realizzare anche quello *Socket* che, come ormai ben sapete, servirà ad ospitare il micro target da programmare e, naturalmente, il circuito *Keypad & Display* che vi stiamo presentando.

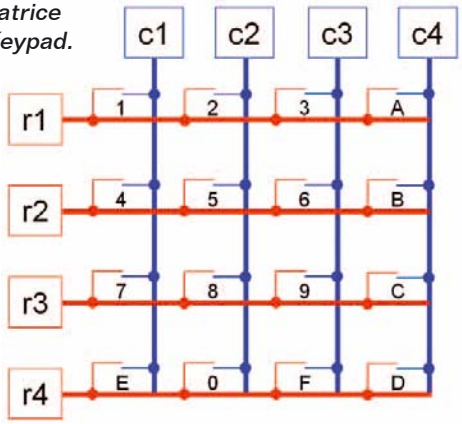


Fig. 1 *La matrice del Keypad.*

SCHEMA ELETTRICO KEYPAD & DISPLAY

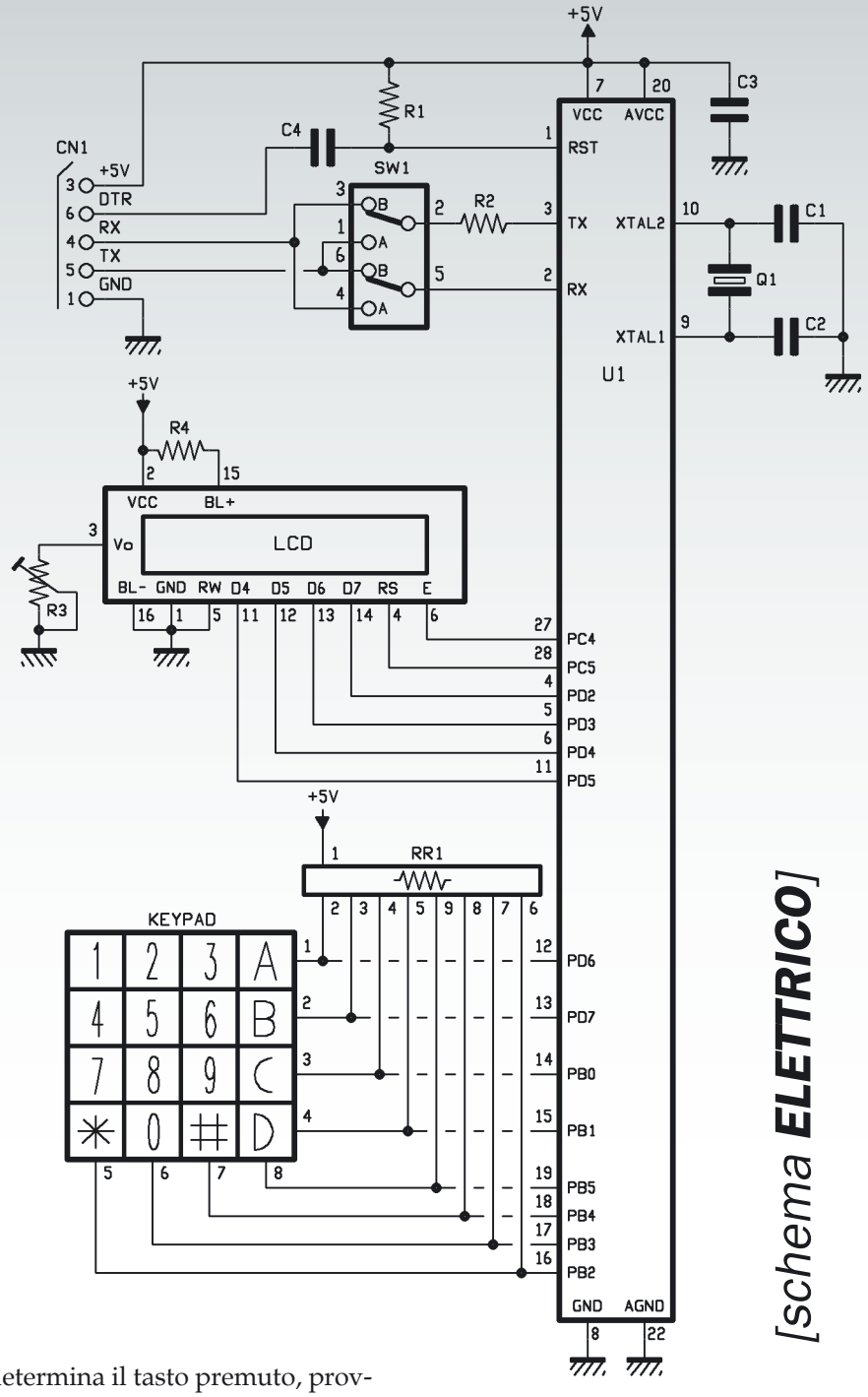
Ora descriviamo lo schema elettrico della sezione *Keypad & Display*. Iniziamo dal circuito integrato, che è un microcontrollore ATMEL ATmega328P nella tipica configurazione stand-alone, con la resistenza R1 in funzione di pull-up per il pin RESET, il quale mediante il condensatore C4 è collegato al connettore CN1; il quarzo Q1 e i due condensatori ceramici C1 e C2 costituiscono l'oscillatore esterno a 16 MHz, mentre C3 si occupa di filtrare l'alimentazione da eventuali disturbi residui.

Sui pin 2 e 3, rispettivamente RX e TX della seriale integrata nel chip, è collegato il mini-switch SW1, la cui funzione è quella di incrociare questi due segnali sul connettore CN1.

Il display DSP1 (nel nostro caso un prodotto della Sparkfun) è un 4x20 (4 righe da 20 caratteri ciascuna) usato nella configurazione con controllo a 4 bit (sono usati i soli pin D04÷D07 e ignorati i pin D00÷D03); è alimentato dai 5 V comuni del circuito e gestito dall'ATmega328P. Il trimmer R3 serve per regolare il contrasto in modo ottimale. Riguardo a questa regolazione, notate che mentre nella gran parte degli schemi troviamo un trimmer configurato come potenziometro (estremi rispettivamente a 5 V e GND e cursore collegato al pin Vo) nel nostro caso il trimmer è collegato come reostato; ciò perché abbiamo notato che nei modelli più recenti di questi display la regolazione si ottiene semplicemente variando la resistenza tra il pin del contrasto e massa (probabilmente all'interno c'è un resistore collegato verso il positivo di alimentazione).

Andiamo avanti e vediamo i due pin 15 e 16, destinati al controllo del LED della retroilluminazione del display (D-A e D-C) i quali sono connessi all'alimentazione tramite un resistore: se questo è un cortocircuito (0 ohm) ottenete la massima luminosità, mentre se volete ridurre l'entità della retroilluminazione potete prevedere valori di 50÷100 ohm.

Se usate un display che non ha la resistenza di limitazione integrata, montate sempre una R4 da alcune decine di ohm, perché altrimenti bruciate rapidamente il LED della retroilluminazione.
Passiamo ora al tastierino esadecimale (keypad), che è costituito da una matrice di 4x4 tasti siglati da 0 a 9, e da A ad F; in realtà molti modelli al posto di "E" e "F" hanno i due caratteri "*" e "#", ma cambia nulla, perché la libreria che andremo ad usare ci permette di definirli a piacere. Per cambiarli basta stampare le due lettere su un'etichetta adesiva, ritagliarle ed incollarle sopra i due simboli originali.
Come ogni matrice, anche quella del keypad è costituita da righe e colonne: la pressione di un tasto del keypad mette in diretta comunicazione la riga e la colonna alla quale il tasto appartiene.
Per ben comprendere il funzionamento hardware e software del keypad è necessario partire dallo studio della sua matrice (**Fig. 1**): come si può vedere, sono presenti quattro colonne (c1-c4) e quattro righe (r1-r4), che corrispondono alle otto piazzole presenti sul keypad per i collegamenti; questi otto contatti sono collegati ad altrettanti pin dell'ATmega328P, che per le colonne sono impostati come uscite, mentre per le righe sono settati come input.
Le righe vengono tenute, a riposo, a livello alto dalle resistenze di pull-up contenute nella rete resistiva RN1; il software esegue rapidissimamente un ciclo continuo (loop) che pone i contatti delle colonne sequenzialmente a livello basso. Appena si preme un qualsiasi tasto, il piedino corrispondente alla riga cui esso è collegato legge lo stato logico basso; il software verifica in quello stesso istante quale pin di colonna sia a livello basso e determina il tasto premuto, provvedendo a identificare il carattere corrispondente.
Avrete già notato che nello schema elettrico abbiamo dotato di pull-up sia i pin di riga che quelli di colonna; ciò per adattare il circuito ai vari tipi di tastiera reperibili in commercio, in quanto alcuni hanno, sul connettore, l'ordine delle righe invertito con quello delle colonne. Dunque, abbiamo previsto una rete resistiva da otto resistenze più comune, collegate indistintamente alle piste di righe e colonne; una volta in possesso del keypad, sarà sufficiente localizzarne le righe e saldare sul circuito stampato solo il pin 1 (comune) della rete resistiva e i quattro pin che combaciano con le piste delle righe; gli altri quattro pin potete indifferentemente



[schema **ELETTRICO**]

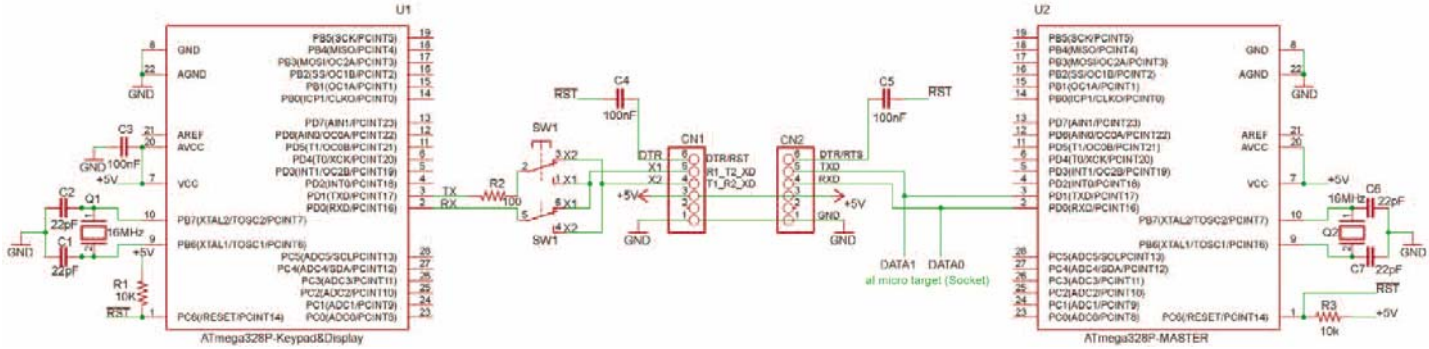


***Fig. 2** - Comunicazione seriale tra i due ATmega328P.*

tagliarli, o lasciarli non saldati o, al limite, anche saldarli. Ovviamente se i collegamenti del vostro keypad sono uguali a quelli del tipo usato da noi (acquistabile presso Futura Elettronica, cod. 5150-TST16), potreste anche usare una rete resistiva a cinque pin, lasciando gli altri quattro fori del circuito stampato completamente liberi.

Chiudiamo la descrizione dello schema elettrico parlando della comunicazione seriale tra i due microcontrollori ATmega328P (Master e K&D) e tra quello di questa scheda ed un eventuale convertitore USB-Seriale esterno.

In **Fig. 2** abbiamo disegnato uno schema elettrico parziale, in cui sono riportati i due chip delle sezioni Master e Keypad & Display, con i relativi connettori; le sigle usate in questo schema non corrispondono necessariamente a quelle reali, per le quali dovete far riferimento ai relativi schemi elettrici e piani di montaggio.

Considerate questo schema una sorta di simulazione.

Durante il normale funzionamento, immaginate che i due connettori CN2 e CN1 siano collegati pin-to-pin mediante un cavetto flat da 6 poli, lungo circa 15 cm.

Nello schema, il mini-switch SW1 è in posizione "1" ("B" sullo schema elettrico e piano di montaggio): ciò significa che il pin TX di U2 (Master) viene a trovarsi collegato con il pin RX di U1 (K&D) e, viceversa, il pin TX di U1 viene a trovarsi collegato al pin RX di U2.

Abbiamo quindi la tipica connessione seriale incrociata.

In condizioni normali, per far dialogare tra loro i micro basterebbe aprire su entrambi una comunicazione, ma nel nostro caso i pin TX/RX di U2 svolgono anche la delicata funzione di trasferimento dei dati durante la programmazione HV, in quanto appartenenti al bus relativo a PORTD; a riguardo, abbiamo riscontrato dei problemi derivanti essenzialmente dallo stato dell'uscita TX di U1, che disturbava la linea DATA0 del bus "D" di U2, quando questa veniva posta in ingresso al micro.

Il problema è stato risolto inserendo in serie a tale connessione la resistenza R2 da 100 ohm, con funzione di disaccoppiamento tra il pin TX di U1 ed il pin RX di U2. Non si sono verificati, invece, problemi sulla linea DATA1, in quanto dall'RX di U1 non poteva arrivare nulla.

Il mini-switch SW1 serve perché CN1 viene impiegato anche per la programmazione del micro U1, mediante un convertitore USB-Seriale esterno; in questo caso il connettore deve avere la stessa configurazione di CN2, così lo stesso convertitore permette di programmare sia l'uno che l'altro micro (ovviamente con due operazioni separate). Per ottenere questa configurazione è necessa-

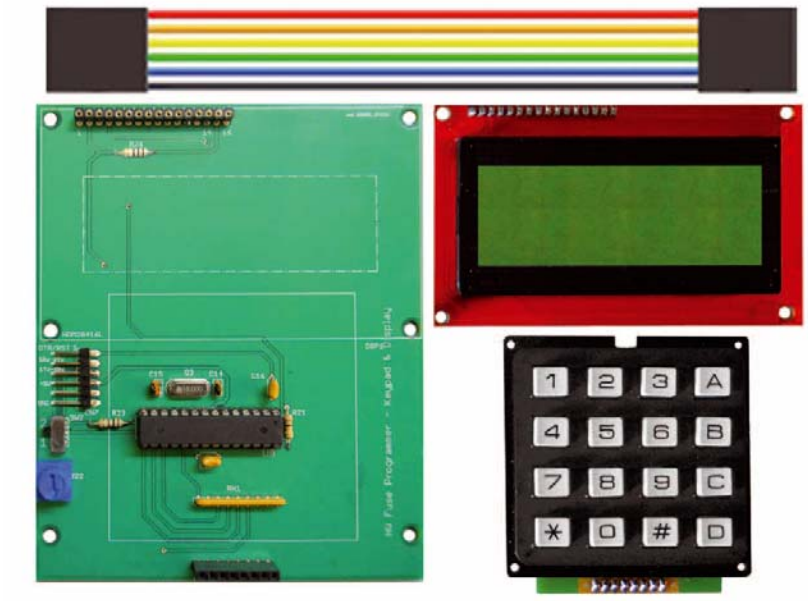


***Fig. 3** - I componenti della sezione Keypad & Display.*

rio porre il mini-switch SW1 in posizione "A".
Questa scelta ci ha permesso di usare una comune piattina a 6 poli per collegare le due schede; in caso contrario avremmo dovuto proporvi l'autocostruzione di un cavetto incrociato.
Riepilogando, il mini-switch SW1 va messo in posizione "B" per il collegamento mediante cavo flat alla scheda Master, mentre la posizione "A" serve per collegare alla scheda *Keypad & Display* ad un convertitore USB-Seriale tipo il già citato FTDI5V.

REALIZZAZIONE SCHEDA K&D

In **Fig. 3** potete vedere le quattro parti necessarie per preparare la sezione *Keypad & Display*: il PCB (lo vedete già con tutti i componenti montati), il display 4x20, il keypad esadecimale e il cavetto flat a 6 poli. I componenti dovranno essere collocati sulla faccia superiore del circuito stampato, preferibilmente rispettando questo ordine: resistenze, zoccolo, connettore strip maschio, quarzo, mini-deviatore, condensatori, rete resistiva, trimmer, connettori di display e tastiera. Particolare attenzione dovrà essere rivolta alla tipologia di connettori da usare per il display e la tastiera, in special modo se decidete di montare tutto nel mobile da noi previsto; nel nostro caso, per ospitare il display ed il keypad abbiamo usato delle file di contatti femmina torniti a passo dip a basso profilo per il display e alti 8 mm per il keypad.
Sul display e sul keypad dovete invece saldare degli strip maschi: la saldatura deve essere effettuata sul lato superiore dei due oggetti, mentre i pin devono sporgere sul lato inferiore; se osservate attentamente la **Fig. 3** potete vedere le saldature già effettuate; non sono visibili gli strip maschi in quanto si trovano sul lato inferiore.
Fate attenzione alle misure: gli strip del keypad possono arrivare alla lunghezza massima di 1cm (distanza tra la basetta del keypad e la punta dei pin), mentre per il display tale lunghezza non dovrà superare 0,8 cm. Questi riferimenti valgono a prescindere dal fatto che poi installerete tutto in un contenitore, in quanto più avanti vi spiegheremo come montare i distanziatori, con le relative misure.

IL FIRMWARE PER LA SCHEDA K&D

Passiamo ora ad analizzare il firmware occorrente a programmare l'ATmega328P che controlla la sezione *Keypad & Display*. Il firmware, denominato "HV_KeD" è scaricabile dalla sezione download del sito *www.elettronicain.it* ed è abbastanza semplice, visto che la maggior parte delle attività necessarie è svolta dalle due librerie specifiche "*LiquidCrystal.h*", che è in dotazione all'IDE (qualsiasi versione) e "*Keypad.h*", scaricabile anch'essa dalla sezione download del sito *www.elettronicain.it*; questa libreria può essere copiata nella cartella originale dell'IDE (percorso *MyPath\arduino-xxxx\libraries*) oppure, se l'avete creata, nella cartella "libraries" della vostra cartella personale Arduino (tipicamente *Documenti\Arduino\libraries*). In ogni caso, il firmware

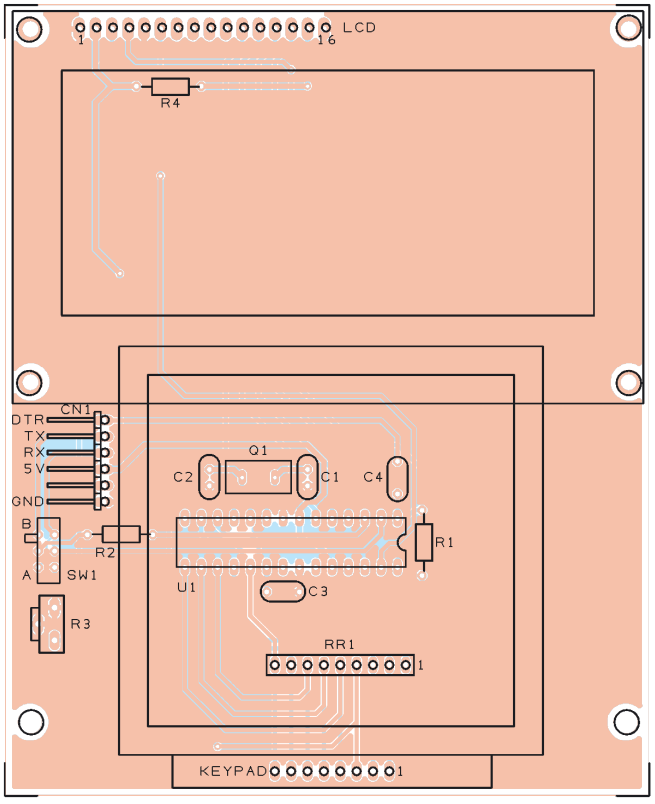


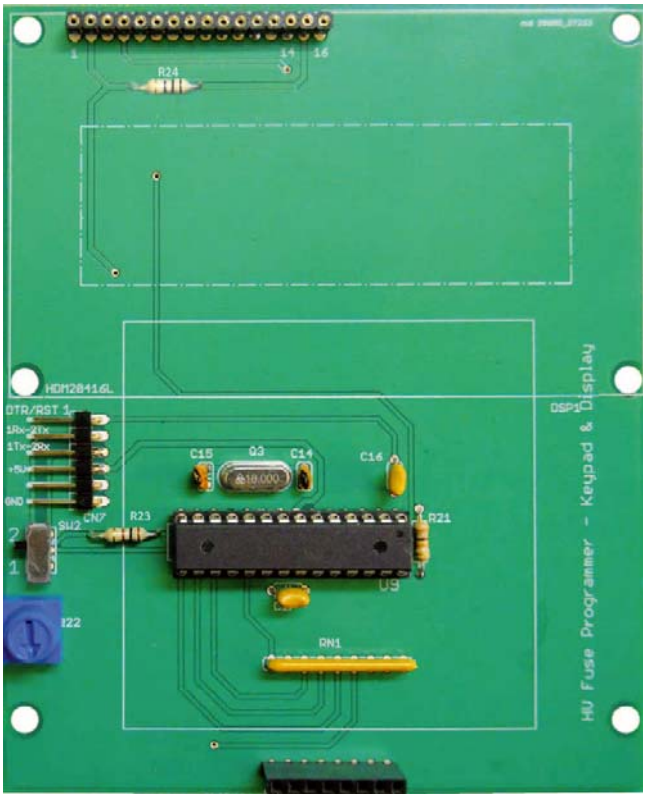
Non solo ATmega...

Atmel è conosciuta principalmente per i suoi microcontrollori, ma in realtà produce anche display per smartphone come molti HTC, Galaxy Note e Galaxy Tab di casa Samsung. In questo campo, sta introducendo XSense, un innovativo display touch-sensitive a tecnica resistiva (caratterizzato da bassa resistenza del film, quindi alta immunità ai disturbi) flessibile, ultrasottile, con le dimensioni di una pellicola. I nuovi smartphone che monteranno XSense non avranno i bordi e i loro lati saranno touch-sensitive, così da sostituire anche quei comandi che si trovano lateralmente (per esempio il volume); la flessibilità e la curvatura che si può dare ai visualizzatori potrebbero dar vita a dispositivi mobili dai form factor innovativi, più ergonomici e funzionali. L'idea sembra buona, tanto che oltre ad Atmel anche LG sta lavorando su un display con touch-sensor destinato, per il momento, agli eReader.
Il filmato su www.atmel.com/microsite/xsense/ spiega la nuova tecnologia Atmel meglio d'ogni parola.

la localizzerà in automatico.
Se disponete di un ATmega328P contenente il bootloader di Arduino Duemilanove e di un convertitore USB-Seriale, il caricamento del firmware diventa un'operazione estremamente semplice. Se invece avete un ATmega328P vergine dovrete ricorrere alla tecnica ISP, che abbiamo già citato ad inizio articolo, per caricarvi il bootloader e poi usare Arduino per aggiungere il firmware, oppure, se non prevedete di ricorrere a successive programmazioni on-board, potrete anche caricare il solo firmware nel chip senza bootloader.
Prima di spiegare il firmware è il caso di fare una precisazione riguardo al funzionamento del display LCD: se provate ad inviare direttamente un testo lungo 80 caratteri ad un qualsiasi LCD 4x20, basato sul notissimo controller HITACHI HD44780, noterete che i primi 20 caratteri riempiranno la prima riga, quelli dal 21 al 40 staranno nella terza riga, i caratteri dal 41 al 60 riempiranno la seconda riga e, infine i caratteri dal 61 all'80 riempiranno la quarta riga. In pratica, solo la prima e l'ultima riga saranno posizionate correttamente, mentre la seconda e la terza risulteranno invertite tra

[piano di **MONTAGGIO**]





Il circuito stampato sopra riportato è stato ridotto in scala all'85 %.

Elenco Componenti:

R1: 10 kohm
R2: 100 ohm
R3: 10 kohm
R4: 0 ohm
RR1: Rete resistiva 8x4,7 kohm + C
C1: 22 pF ceramico
C2: 22 pF ceramico
C3: 100 nF multistrato
C4: 100 nF multistrato
LCD: Display LCD 4x20
U1: ATmega328P (MF1004)
Q1: Quarzo 16 MHz
KEYPAD: Tastiera a matrice 4x4
SW1: Micro deviatore a slitta 2 vie

Varie:
- Strip maschio 90° 6 poli
- Strip maschio 8 poli
- Strip femmina 8 poli
- Strip maschio 16 poli
- Strip femmina 16 poli
- Zoccolo 14+14
- Circuito stampato

loro. Tutto ciò è dovuto semplicemente alla gestione della RAM interna al controller, che causa questo comportamento.
Quanto abbiamo appena spiegato serve per comprendere perché abbiamo dovuto creare un piccolo "protocollo" di comunicazione tra la sezione *Master* e la sezione *Keypad & Display*: se avessimo fatto diversamente, ci sarebbe stato impossibile gestire correttamente le informazioni da visualizzare sul display.
È questa la ragione per cui nella prossima puntata, nel presentarvi il firmware per la sezione *Master* diremo che ne esistono due versioni: una "normale", per la comunicazione col PC (la quale non ha particolari necessità) ed una specifica per la versione stand-alone del Programmatore HV (quella dotata, appunto, del display LCD).
Trovate il firmware per la sezione *Keypad & Display* nel **Listato 1**, dove potete notare le numerose note autoesplicative. A grandi linee diciamo che la parte iniziale serve per caricare le due librerie specifiche per la gestione del keypad e del display LCD 4x20, quindi vengono inseriti i vari parametri di inizializzazione delle due periferiche, operazione, questa, indispensabile in quanto la libreria "*Keypad.h*" permette di gestire anche keypad di tipo telefonico a 12 tasti; permette inoltre la totale personalizzazione della disposizione dei tasti, cosa estremamente utile se disponete, per esempio, di una tastiera esadecimale con una diversa disposizione dei tasti */E e #/F.
La libreria "*LiquidCrystal.h*" permette anch'essa di gestire diverse tipologie di display LCD, sia a 8 bit che a 4 bit (il nostro caso), e con diverse tipologie di quantità di righe e caratteri, quindi anche in questo caso occorre fornire una

Listato 1

```
/* Firmware per il progetto HV Programmer
   Sezione Keypad & Display
   realizzato dal Prof. Michele Menniti
   Rivista Elettronica In*/

 /* Un particolare ringraziamento a Leonardo Miliani
    per il prezioso contributo fornito */

/* LIBRERIE*/
#include <Keypad.h>
#include <LiquidCrystal.h>

/* INIZIALIZZAZIONE PERIFERICHE */

// inizializza il Display LCD indicando i collegamenti dei pin all'ATmega328P
// rispettivamente: RS, Enable, D4, D5, D6, D7
LiquidCrystal lcd(A5, A4, 5, 4, 3, 2);

// parametri del keypad esadecimale
const byte ROWS = 4; // 4 righe
const byte COLS = 4; // 4 colonne

// mappa del Keypad esadecimale, deve corrispondere ai tasti fisici del keypad
char keys[ROWS][COLS] = {
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'E','0','F','D'}
};

// pin digitali dell'ATmega328P a cui sono connessi ROW1, ROW2, ROW3 e ROW4 del keypad
byte rowPins[ROWS] = { 9, 8, 7, 6 };
// pin digitali dell'ATmega328P a cui sono connessi COL1, COL2, COL3 e COL4 del keypad
byte colPins[COLS] = { 13, 12, 11, 10 };

// inizializzazione del keypad
Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

void setup()
{
  Serial.begin(9600);  // apertura della comunicazione seriale
  lcd.begin(20, 3);   // numero di caratteri per riga, numero righe (0,1,2,3)
  lcd.setCursor(0, 0); // inizialmente il cursore viene messo sulla 1a posizione
}

void loop()
{
  while (Serial.available() == 0) // resta in attesa di un carattere
  { //finché non arrivano caratteri dalla seriale controlla il keypad
    char key = kpd.getKey(); // controlla se viene premuto un tasto
    if(key)   // in caso di pressione di un tasto
      {
      lcd.print(key);    // scrive il relativo carattere sul display
      Serial.print(key); // e lo manda al Master
      }
  }
    char code = Serial.read();
    if (code == '$' || '|' || '^' || '%' || '#') // caratteri speciali
    {
      switch (code) {
         case '$': // cancella il display
           lcd.clear();
           break;
         case '|': // cursore sulla prima riga
           lcd.setCursor(0, 0);
           break;
         case '^': // cursore sulla seconda riga
           lcd.setCursor(0, 1);
           break;
         case '%': // cursore sulla terza riga
           lcd.setCursor(0, 2);
           break;
         case '#': // cursore sulla quarta riga
           lcd.setCursor(0, 3);
           break;
       default: // stampa tutti i caratteri e ignora i caratteri codice ascii esteso
           if (int(code)>0 && int(code)<127) lcd.print(code);
      }
   }
}
```

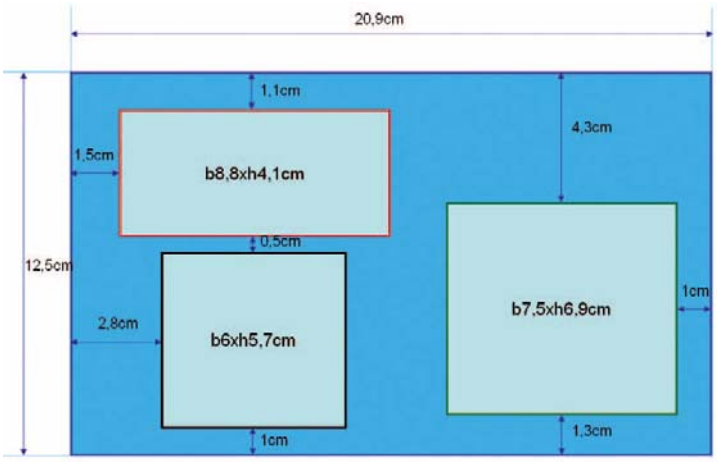


***Fig. 4** - La lavorazione della maschera del contenitore.*

serie di elementi fondamentali per il corretto funzionamento del display.
Ci soffermiamo solo sulle righe della sezione esecutiva (loop), per spiegare il funzionamento del protocollo cui abbiamo fatto riferimento in precedenza: appena si fornisce alimentazione al Programmatore HV, immediatamente il firmware della sezione Keypad & Display si pone in ascolto sulla seriale, cioè resta in attesa del primo carattere proveniente dalla sezione *Master*; in questa fase è attivo l'uso della tastiera, perciò se viene premuto un qualsiasi tasto, il relativo carattere esce sul display e contemporaneamente viene trasmesso alla sezione *Master*. Appena arriva un qualsiasi carattere, esso viene analizzato in base al protocollo, in modo da distinguere i caratteri "comando" dai caratteri "testo". I caratteri "comando" sono quelli che permettono di fornire impostazioni al display LCD, tramite le istruzioni della libreria, e sono:

- $: esegue il comando "lcd.clear();" che permette di cancellare tutto il contenuto del display e posizionare il cursore sulla prima posizione (primo carattere della prima riga);
- | : esegue il comando "lcd.setCursor(0, 0);" che posiziona il cursore all'inizio della prima riga;
- ^ : esegue il comando "lcd.setCursor(0, 1);" che posiziona il cursore all'inizio della seconda riga;
- % : esegue il comando "lcd.setCursor(0, 2);" che posiziona il cursore all'inizio della terza riga;
- # : esegue il comando "lcd.setCursor(0, 3);" che posiziona il cursore all'inizio della quarta riga.

A titolo di esempio, la stringa "%prova di stampa" significa che la frase "prova di stampa" dovrà essere visualizzata sul display a partire dall'inizio della terza riga (%). I caratteri "comando" possono essere usati anche in sequenza: ad esempio la stringa "$^prova" significa che prima viene cancellato il contenuto del display ($), e subito dopo la parola "prova" sarà visualizzata sul display a partire dall'inizio della seconda riga (^).
Tutti i caratteri non appartenenti all'elenco precedente saranno trattati come testo da visualizzare, a patto che rientrino nella sezione base della codifica "ASCII", quindi tutti quei caratteri il cui valore decimale rientri nel range da 1 a 126; l'esclusione dei caratteri con valore decimale "0" e da "127" compreso in su, permette al microcontrollore di ignorare caratteri strani che potrebbero derivare da disturbi presenti sulla linea seriale.
Per questa volta non vi mostriamo immagini relative al funzionamento del programmatore; le vedremo tutte nella prossima puntata, quando avremo caricato il firmware nel micro della sezione *Master*. Però possiamo suggerirvi una semplice verifica per la sezione *Keypad & Display*; se non avete commesso errori di montaggio e se avete caricato correttamente il firmware, appena fornirete l'alimentazione alla scheda, tramite i pin +5V e GND del connettore CN7, provate a premere, uno alla volta, tutti i tasti del keypad: dovranno uscire i relativi caratteri sul display. Vi ricordiamo che se riempirete la prima riga, la successiva diventerà la terza e non la seconda come ci si aspetterebbe; sappiate che ciò è normale e che non vi creerà alcun problema, dato che ad ordinare le righe provvederà il protocollo di dialogo che abbiamo creato.

IL CONTENITORE

Se avete deciso di realizzare il Programmatore HV in modalità standalone, vi consigliamo vivamente di sistemarlo all'interno di un mobiletto adeguato. Se usate un contenitore simile a quello che abbiamo adottato noi (è il modello PULT363 della Teko, acquistabile presso la Futura Elettronica -*www.futurashop.it*- con il codice 5100-PULT363), la parte più complessa della lavorazione è

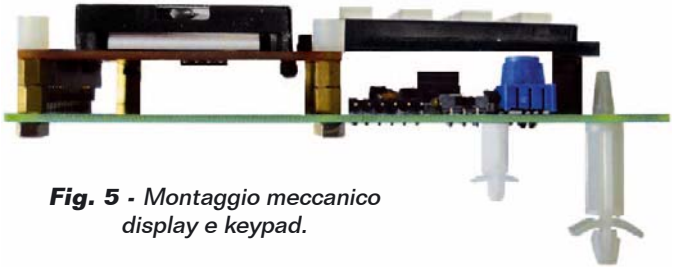


***Fig. 5** - Montaggio meccanico display e keypad.*

quella riguardante la mascherina in alluminio, nella quale andranno create le finestre per far fuoriuscire il display LCD, i tasti del keypad e gli zoccoli, i due tasti e i due LED del PCB Socket e, infine, l'eventuale interruttore di accensione dello strumento.
Nella **Fig. 4** potete vedere il disegno di lavorazione della maschera con le tre finestre e le relative dimensioni e distanze rispetto ai bordi del contenitore.
Le misure indicate sono quelle relative ai componenti da noi usati; in particolare il display, il keypad e gli zoccoli ZIF. Purtroppo non esiste uno standard e le misure dei vari display 4x20 sono simili ma non identiche; inoltre, spesso i fori di fissaggio sono in posizioni differenti. Anche sul keypad abbiamo riscontrato differenze dell'ordine di 1-2 mm su entrambi i lati; per questa ragione nell'immagine non vedete alcun foro per le viti.
Dunque, se le misure dei vostri componenti sono diverse da quelle previste dovrete lavorare la mascherina di conseguenza. La cosa migliore sarebbe riuscire a stampare in scala 1:1 il disegno della **Fig. 4**; allo scopo, fate il rapporto tra la misura di un lato risultante dalla quota e quella dello stesso sulla pagina: otterrete la scala di riduzione. Poi fotocopiate il disegno ingrandendolo di conseguenza (cioè dell'inverso della scala di riduzione) ed otterrete la maschera di lavorazione, che incollerete su un cartoncino rigido. A questo punto appoggiate i componenti sul disegno e verificate se le varie dimensioni e spaziature coincidono con le nostre, altrimenti tagliate il cartoncino con le misure esatte, fategli anche i fori per le viti e poi sovrapponetelo alla maschera per tracciare i vari punti da tagliare o forare.
La lavorazione di una maschera di alluminio non è difficile ma, se non siete esperti, rimedierete qualche rigatura e imperfezione; una buona verniciatura fatta con una bomboletta spray, potrà rendere ciò meno visibile.
Una volta pronta la maschera, possiamo fissarvi tutti i componenti previsti. Iniziamo con il completamento del blocco relativo alla sezione *Keypad & Display* che, come già visto in **Fig. 3**, è costituito dal PCB e dalle due periferiche, le quali vanno innestate negli appositi connettori, iniziando dal display: come si vede in **Fig. 5**, nei quattro fori predisposti per il fissaggio del display bisogna montare dei distanziatori (plastici o metallici) maschio-femmina alti 1 cm (nel nostro caso ognuno è ottenuto da due da 0,5 cm sovrapposti) e fissarli con un dado nella parte inferiore del PCB. A questo punto potete innestare il display prima ed il keypad dopo (visto che il suo lato superiore appoggia sul lato inferiore del display).
Ora dovete fissare il display ai quattro distanziatori, operazione fattibile con ulteriori quattro distanziali maschio-femmina da 0,5 cm, rigorosamente in plastica, in modo da mantenere la mascherina isolata dal circuito; in pratica, l'estremità "maschio" andrà infilata nel foro del display e quindi avvitata al distanziatore sottostante, così da fissare il display, mentre l'estremità "femmina" si troverà a contatto con la mascherina, ed in tale corrispondenza andranno realizzati i relativi fori per le quattro viti.
Nei due fori restanti, posti ai lati della zona del PCB riservata al keypad, potete decidere per due diverse opzioni: la prima è usare due distanziali in plastica maschio-femmina da 1,6 cm (comunque che siano perfettamente allineati agli altri quattro) e realizzare ulteriori due fori sulla mascherina; la seconda, invece, consiste nel montare due distanziatori da 2,8 cm, nella parte inferiore del PCB, in modo che, a mascherina montata, essi tocchino il fondo del mobile facendo da sostegno al PCB (**Fig. 5** e **Fig. 6**).
Il PCB Master va montato sul fondo del mobiletto (**Fig. 7**) utilizzando magari dei distanziali adesivi a basso profilo (ne vanno usati due per i fori che si trovano nella

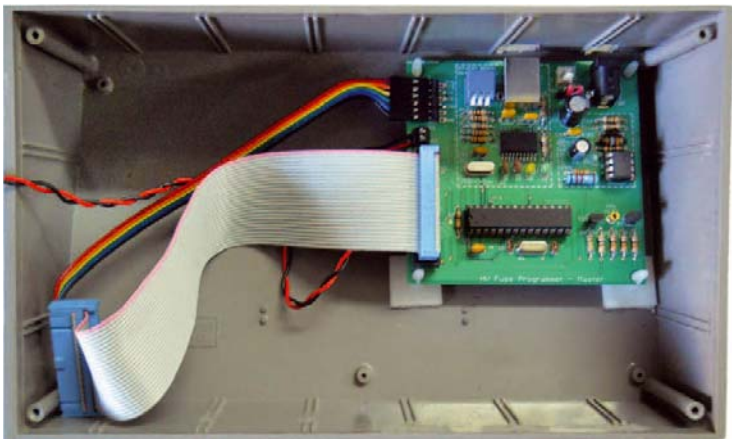


***Fig. 7** - Montaggio del PCB Master nel contenitore.*



***Fig. 6** - Collegamento del keypad e distanziatori.*

parte centrale del contenitore); il PCB va sistemato in modo che il lato posteriore (quello con la porta USB e la presa plug per l'alimentazione) sia perfettamente a contatto con la parete posteriore del contenitore. Nei due fori del lato posteriore del PCB vanno inseriti dei distanziatori normali, sempre a basso profilo, che avranno la stessa funzione di quelli visti per la parte relativa al keypad; quelli adesivi andranno messi nei fori del PCB e poi attaccati sul fondo, in modo da ottenere la massima adesione del PCB alla parete posteriore. La posizione ideale dei due distanziatori adesivi, sul fondo del contenitore, è a circa 4 cm dal bordo anteriore del contenitore stesso. Una volta fissato il PCB, dovrete segnare i contorni della presa USB e della presa plug, poi toglierete nuovamente il PCB e forerete la parete posteriore del contenitore, per permettere l'introduzione dei relativi spinotti a mobile chiuso. Il PCB Socket va fissato sulla maschera in alluminio del mobiletto (**Fig. 8**); in pratica, in corrispondenza di ogni foro il PCB va distanziato dalla maschera mediante una rondella in plastica spessa circa 3 mm (noi abbiamo usato una rondella di plastica da 2 mm ed una in ferro da 1 mm). Nella maschera va infilata una vite di lunghezza tale da fuoriuscire nella parte inferiore del PCB, dove bisognerà aggiungere una rondella isolante sottile e serrare con un dado. Considerando la notevole pressione che verrà esercitata ad ogni inserimento di un chip da programmare, abbiamo previsto 6 fori per le viti; se decidete di usare gli zoccoli ZIF, basteranno i soli quattro fori laterali.

Da notare, infine, il foro più grande nell'angolo anteriore destro del PCB Socket: serve per farvi passare il cilindro plastico del contenitore, sul quale si fissa una delle viti di chiusura della maschera di alluminio; questo espediente vi permette di montare il PCB il più possibile vicino ai bordi della maschera, aumentandone la rigidità. Se però preferite montare il PCB in posizione centrale, tale foro va ignorato. L'importante è che poi usiate gli altri sei fori previsti per le viti.

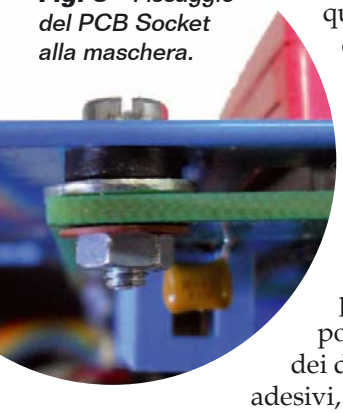


***Fig. 8** - Fissaggio del PCB Socket alla maschera.*

per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono di facile reperibilità. Il master del circuito stampato ed il software per programmare il microcontrollore possono essere scaricati dal sito della rivista.
Il microcontrollore vergine ATMega328 (cod. ATMEGA328P-PU) è disponibile al prezzo di 7,00 Euro 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

Terminiamo qui la seconda puntata dedicata al Programmatore HV. Nel prossimo numero approfondiremo alcune tematiche teoriche sui microcontrollori, che sono state nostro oggetto di ricerca per la progettazione di questo eccellente strumento. Inoltre vi presenteremo le due versioni del firmware da installare sul microcontrollore della sezione Master: quella per l'uso di un Terminale seriale su PC e quella per l'uso del display LCD per il progetto stand-alone. Naturalmente il firmware sarà disponibile gratuitamente.



ARDUINO

lo strumento per stimolare il tuo ingegno e la tua creatività

Tre proposte per iniziare a sperimentare con la piattaforma Made in Italy più famosa al mondo.

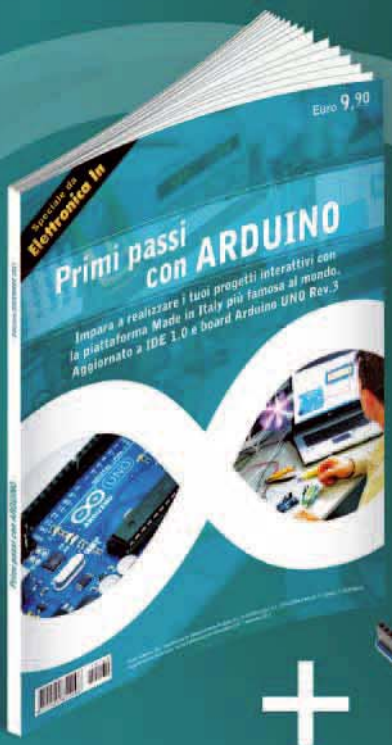


Starter kit con Arduino UNO R3

Comprende la scheda Arduino UNO R3 e tutto il materiale necessario per realizzare gli esperimenti didattici descritti nel volume "Primi passi con ARDUINO".

cod. ARDUINOKITBOOK

€ 52,00



cod. ARDUINOKITBOOK1

solo € 29,90

risparmi € 4,50 !



Volume " Primi passi con ARDUINO " e scheda Arduino UNO R3

Prezzi IVA inclusa.



Arduino UNO R3 è la più recente versione della scheda di sperimentazione basata su Atmega328.

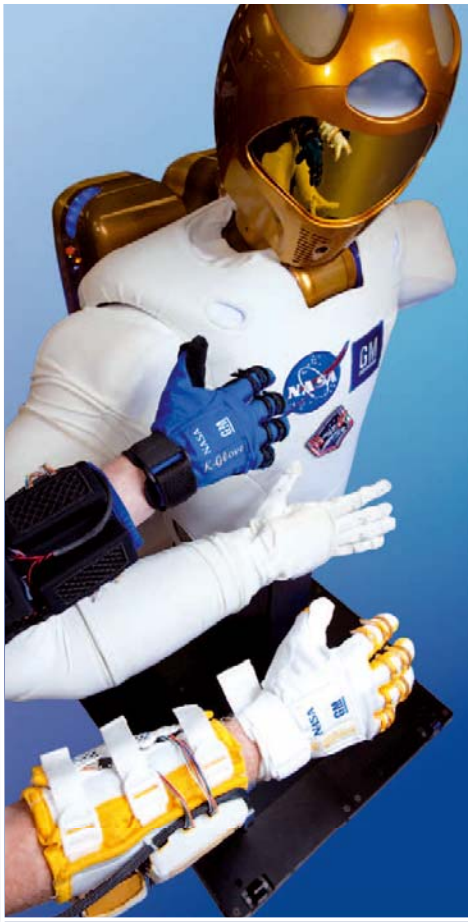
cod. ARDUINOUNOREV3

€ 24,50

FUTURA ELETTRONICA

Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Maggiori informazioni su questi prodotti e su tutte le altre apparecchiature sono disponibili sul sito *www.futurashop.it* tramite il quale è anche possibile effettuare acquisti on-line.



Guanto robot per meccanici ed astronauti

General Motors e NASA hanno collaborato per realizzare un guanto robot, cioè una sorta di protesi in grado, se indossata, di incrementare la forza della mano e di proteggere l'operatore nei lavori pericolosi e particolarmente faticosi. La protesi si indossa dall'avambraccio alla mano e contiene degli attuatori elettromeccanici con aste che simulano il movimento dei tendini, in grado di seguire il movimento di chiusura del pugno amplificando la forza delle dita. K-Glove, così si chiama il guanto robotico, è un esoscheletro per il braccio pensato appositamente per chi fa movimenti ripetitivi, che dimezza lo sforzo (ad esempio afferrare uno strumento che richiede 9 kg di forza "a mani nude" con il guanto robotico richiede circa 4,5 kg) e può ridurre il rischio di infortuni provocati da stress prolungati. Il prototipo pesa circa 900 grammi e si compone di batterie al litio, attuatori, tendini sintetici e sensori incorporati nei polpastrelli che rilevano la pressione della mano sull'oggetto che sta stringendo e forniscono il segnale di attivazione agli attuatori. Il K-Glove è stato in realtà sviluppato dal progetto Robonaut 2, l'androide che dalla scorsa primavera si trova a bordo della Stazione Spaziale Internazionale, dove aiuterà gli astronauti umani nei lavori più rischiosi e gravosi anche all'esterno della stazione stessa.



www.nasa.gov

Manganese: alta capacità, basso costo

L'auto elettrica è sempre più una realtà, anche se rimane ferma ai blocchi di partenza o confinata all'ambito cittadino a causa dell'ancora limitata autonomia delle pur prestanti batterie sviluppate per le più recenti vetture Tesla e dal fatto che mediamente occorre sostituire la batteria dopo 80÷100.000 km di percorso, con un esborso di almeno 5.000 euro (il prezzo di un motore a benzina per una vettura medio-grande). Ma dalla California giunge una notizia che potrebbe smuovere le acque: la ENVIA è riuscita a mettere a punto una batteria agli ioni di litio che consente una densità di carica di 400Wh/kg, quindi potrebbe accumulare l'energia che permette ad un motore elettrico da 400 W di viaggiare a pieno regime per un'ora. Questo risultato è stato ottenuto impiegando manganese per costruire il catodo; questo elettrodo viene ottenuto con metalli a basso costo (compreso il manganese) che permette di immagazzinare più energia per unità di peso di qualsiasi altra batteria (il doppio delle batterie con ossido di litio cobalto). La stabilità del materiale usato permette di allungare la durata anche ad alte tensioni di carica; inoltre ogni cella si carica a 4,6 V e quindi un pacco batteria conta meno celle di uno formato da elementi Li-Ion o Li-Po, che si caricano a 4,2 V. Meno celle vuol dire anche minori costi del caricabatteria e del circuito di bilanciamento, che notoriamente deve monitorare la carica individualmente cella per cella.

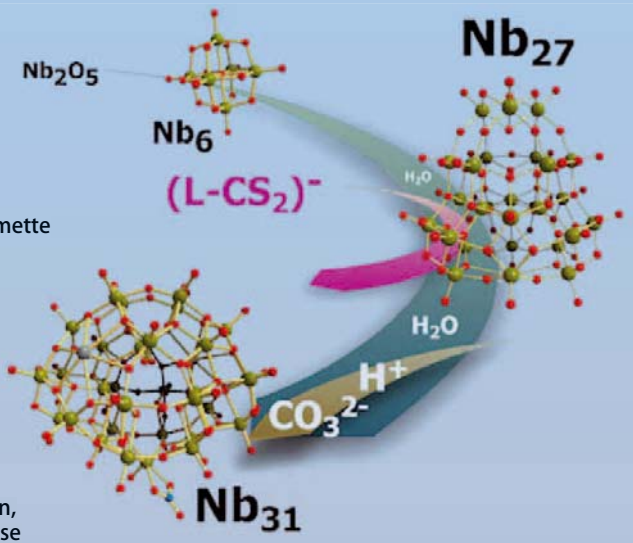


Le nuove batterie sono state testate presso l'Electrochemical Power Systems Department al Naval Surface Warfare Center, dove è stato riscontrato che la tecnologia ENVIA permette densità di energia tra 378 e 418 Wh/kg con fattori di carica fra 1/3 C e 1/10 C ; il test è stato condotto su accumulatori da 45 Ah .

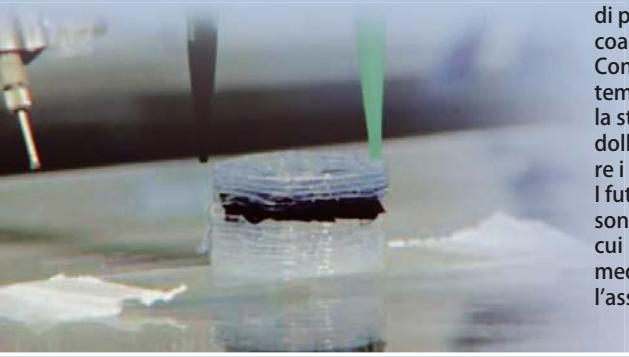
http://enviasystems.com/



La pillola si stampa in 3D



È ora di prendere la medicina ma vi siete accorti che la farmacia è chiusa? Nessun problema: accendiamo il computer, cerchiamo la ricetta elettronica e la nostra stampante 3D casalinga ci stamperà le pastiglie! Sembra una scena da film di fantascienza, ma presto potrebbe essere una realtà; infatti la diffusione della stampa in 3D tocca ormai sempre più settori e non solo dell'industria.
Lo dimostra la ricerca condotta all'Università di Glasgow, dove hanno ideato un sistema che permette di produrre medicinali utilizzando una stampante 3D commerciale ed un composto, battezzato "reactionware" dal gruppo di ricerca capitanato dal professor Lee Cronin, e basato su gel a base di polimeri che ospitano e coadiuvano le reazioni chimiche.



Con la tecnica messa a punto a Glasgow, tempi e costi di produzione si riducono: la stampante utilizzata costa circa 2.000 dollari, e bastano poche ore per ottenere i gel più complessi.
I futuri sviluppi di questa soluzione possono cambiare radicalmente il modo in cui vengono prodotte e distribuite le medicine, cosa rilevante soprattutto per l'assistenza sanitaria nei Paesi in via di sviluppo. Addirittura, Cronin pensa a un futuro in cui le stampanti si troveranno in ogni casa per la produzione di ogni sorta di oggetti utili, medicinali compresi; le istruzioni per la fabbricazione potranno essere impartite da delle app che permetterebbero di personalizzare i composti a seconda delle necessità del paziente.

www.chem.gla.ac.uk/cronin/

GCV: il carro armato ecologico

A leggere la notizia viene da fare dell'ironia: BAE Systems e Nortrop Grumman stanno mettendo a punto GCV (Ground Combat Veichle) un carro armato a propulsione ibrida diesel-elettrica per la Difesa degli Stati Uniti; insomma, un'arma ecologica. Sì, preoccuparsi dell'inquinamento di un mezzo destinato a distruggere sembra quasi un paradosso, ma visto che l'uomo in tutte le sue attività sta cercando di operare in modo green, tanto vale farlo anche quando si tratta di guerra. Il potente mezzo corazzato può trasportare fino a 12 soldati, è mosso da due motori da ben 700 cavalli di potenza e pesa, batterie comprese, circa 60 tonnellate; dovrebbe costare circa 11 milioni di dollari. La propulsione è ottenuta dai due motori elettrici che comandano un cingolo ciascuno e vengono alimentati da una batteria agli ioni di litio, ricaricata da due gruppi elettrogeni diesel che intervengono sia se la batteria scende sotto un certo livello di carica, sia quando ai motori di trazione è richiesto un surplus di potenza. Inoltre l'elettricità prodotta a bordo permetterebbe di alimentare sistemi d'arma avanzati come cannoni a laser, a microonde e coil-gun.

www.teamgcv.com



L'auto pensa alla nostra salute

Ford sta sviluppando interessanti novità per le sue automobili del futuro; particolarmente suggestive sono quelle "healthcare" tra cui il monitor dell'attività cardiaca del conducente e la guida attraverso le zone a ridotta concentrazione di sostanze allergiche. Il monitor cardiaco si basa su un computer che, tramite sensori inseriti nello schienale del sedile lato guida, monitorerà il battito cardiaco del conducente; questo potrebbe essere utile per tenere sotto controllo lo stato di salute al volante o inviare dettagli a un'ambulanza in caso di incidente, o anche ad allertare gli altri guidatori (mediante una connessione in rete tra veicoli) nel caso il conducente sia preso da un attacco cardiaco. L'Allergy Alert è invece un sistema in grado di dirci se stiamo entrando in un'area con tanto polline nell'aria e suggerirci strade alternative per evitare di stuzzicare l'allergia. È sufficiente indicare il tipo di allergia di cui si soffre e il computer fa il resto. La piattaforma Sync che Ford usa per far girare applicazioni come Allergy Alert è basata su Windows CE, ma funziona anche con qualsiasi altra piattaforma smartphone.

http://media.ford.com/

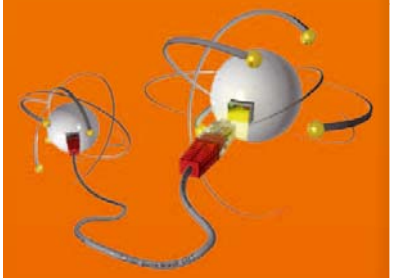


Il nanovetro per display e pannelli solari

Al MIT hanno sperimentato uno speciale vetro "multifunzionale" basato su nanostrutture superficiali che lo rendono repellente all'acqua ed allo sporco, antiappannamento e antiriflesso; i ricercatori sperano ora che il processo costruttivo possa essere reso economico in modo da destinare il vetro agli schermi di smartphone e televisori, ai parabrezza delle automobili e persino alle vetrate fisse degli edifici. Il nuovo vetro è particolarmente adatto a rivestire i pannelli fotovoltaici, che a causa della polvere e dello sporco che vi si depositano possono perdere fino al 40 per cento della loro efficacia entro sei mesi dall'installazione; inoltre, con esso i pannelli guadagnerebbero in efficienza, perché il vetro non riflette la luce come quello comune, specialmente nelle ore del primo mattino o del tardo pomeriggio, quando i raggi del sole giungono con una certa inclinazione. Altre applicazioni del nuovo vetro potrebbero essere nei microscopi e nelle macchine fotografiche da utilizzare in ambienti umidi, dove sia le capacità antiriflesso sia quella antiappannamento potrebbero essere utili.
La nanostruttura di superficie è costituita da una serie di coni con base di 200 nanometri e altezza di 1 µm; si realizza rivestendo la superficie di vetro con diversi strati sottili, compreso uno strato di photoresist, quindi procedendo all'incisione secondo la stessa tecnica fotolitografica utilizzata nella fabbricazione dei semiconduttori: si rimuove il fotoresist dove bisogna incidere la superficie e poi si procede ad un attacco chimico di precisione nanometrica.

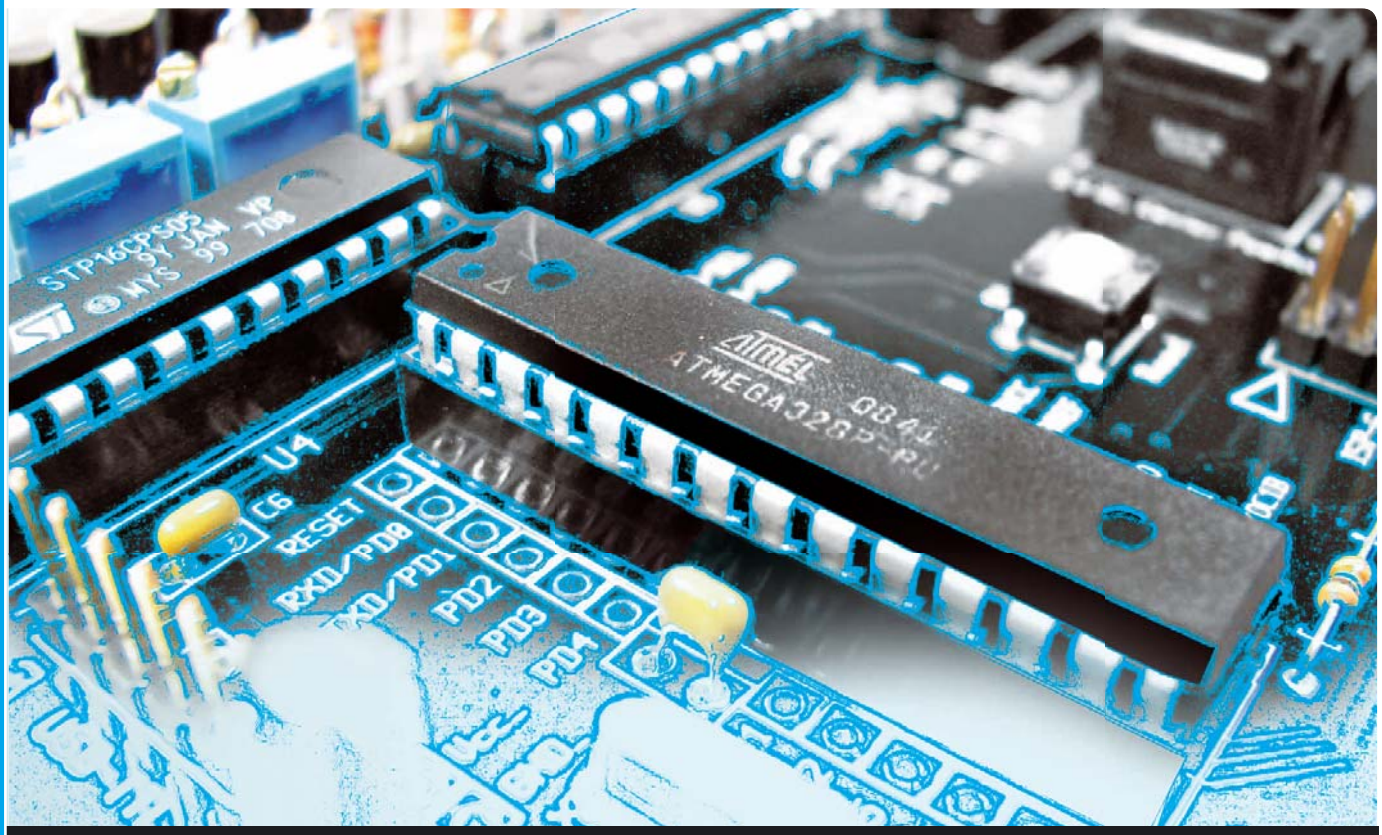
http://web.mit.edu/newsoffice/

Verso l'Internet quantistica



Stephan Ritter e il suo gruppo di ricerca all'Istituto Max Planck di Garching, in Germania, hanno costruito una rete quantistica elementare con due nodi; questa struttura potrebbe divenire la base per creare grandi reti di informazione quantistica che garantiscono la sicurezza della trasmissione dei messaggi dai dati di codifica che si basano sullo stato quantistico di fotoni.
Dunque, dopo i computer a stati quantistici, ora si parla anche di reti e non solo perché possono essere realizzate con dimensioni e tempi di propagazione minimi, ma anche per il fatto che sembrano essere indenni dall'hacking. Questa sicurezza si deve a una regola fondamentale della meccanica quantistica, secondo cui quando si compie una misura su un sistema quantistico essa implica dei cambiamenti; quindi qualsiasi intromissione cambia le proprietà quantistiche in modo evidente e rivela così il tentativo di intercettazione dei dati.
Le reti di informazione quantistiche non sono una novità assoluta: già nel 2008 la città di Vienna ha installato una rete di informazione quantistica, ma in essa alcuni nodi potevano solo inviare i dati e altri potevano solo ricevere. Invece Ritter ha costruito un nodo universale, in grado di ricevere informazione quantistica, memorizzarla e trasmetterla. Il nodo di Ritter è stato costruito con atomi di rubidio intrappolati in cavità ottiche; singoli fotoni sono serviti come vettori di informazioni, con l'informazione quantistica da atomi di rubidio codificati nella polarizzazione dei fotoni.

www.mpq.mpg.de/cms/mpq/en/news/press/



ATMEL OPEN SOURCE

dell'ing. OSVALDO SANDOLETTI

Presentiamo gli strumenti per lavorare praticamente con il microcontrollore ATMEGA16: apprendiamo i sistemi per programmarlo costruendo una scheda di sviluppo in cui inserirlo. Quinta puntata.

In questa puntata descriviamo una applicazione pratica basata sui concetti sinora discussi, per mettere in funzione il microcontrollore ATMEGA16 della nostra demoboard.
Ebbene ci siamo: dopo avere visto che cos'è un microcontrollore, avere imparato a scrivere, compilare e simulare i programmi, ed infine avere discusso l'utilizzo degli strumenti hardware (programmatore e demoboard) è venuto il momento di mettere tutto insieme per sviluppare un'applicazione che faremo concretamente eseguire dal microcontrollore ATMEGA16: trattandosi della prima prova pratica in assoluto, il suo scopo sarà l'accensione di un LED ad esso collegato. Anche se la cosa potrebbe suonare apparentemente banale, tenete presente che questa prima applicazione permetterà di imparare nuovi concetti di programmazione e preparare la base per la realizzazione di programmi molto più complessi.

PREPARAZIONE DEL SISTEMA

Per lo sviluppo delle applicazioni pratiche vi

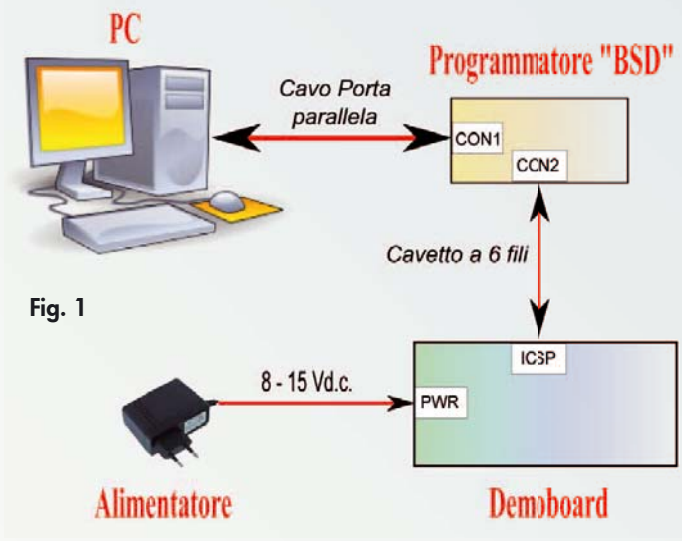


Fig. 1

consigliamo di allestire un "banco di prova" (il cosiddetto setup) che, come schematizzato in **Fig. 1**, è composto da un PC (su cui sono installati WinAVR e AVR Studio), la demoboard ed il programmatore parallelo "BSD".
Se tuttavia il vostro PC non dispone di una porta parallela, in base a quanto esposto nella scorsa puntata, dovrete ricorrere ad un programmatore seriale (se avete disponibile tale porta) oppure USB (come il modello "Pololu USB AVR Programmer", reperibile su Internet). Il cavo per la connessione alla porta parallela può essere acquistato in qualunque negozio di informatica; e va collegato al PC sulla relativa porta (vaschetta D-SUB a 25 piedini). Il cavetto a sei fili che collega il programmatore al connettore ICSP della demoboard, invece, si può auto costruire con sei spezzoni di filo lunghi al massimo 20cm, che si possono saldare ai capi di un connettore strip femmina per poterla collegare e scollegare agevolmente sul corrispondente correttore maschio della scheda. Con questo sistema potete programmare l'ATMEGA16 direttamente sulla demoboard, senza doverlo sempre sfilare per inserirlo nello zoccolo del programmatore. Utilizzando un programmatore commerciale, che "esce" con un connettore a 3+3 oppure 5+5 piedini detto "AVR ISP", dovrete anche realizzare le connessioni di adattamento (Figura 2 della scorsa puntata).
Infine potete alimentare il sistema collegando un alimentatore che fornisca una tensione compresa tra 8 e 15V. In **Fig. 2** si può osservare la disposizione dei componenti del sistema che abbiamo utilizzato per sviluppare i programmi del corso.
Nota: anche se non avete ancora costruito una demoboard, per eseguire l'esercitazione di questa puntata è sufficiente collegare il micro (anche utilizzando collegamenti filati su una scheda millefori) secondo lo schema proposto nella Figura 3 della scorsa puntata, e collegarvi un diodo Led, tramite un resistore da 220-330 ohm, tra il piedino 21 e la massa (come nello schema della demoboard). Ovviamente dovrete anche fornire un'alimentazione di 5V stabilizzati.

E LUCE SIA!

Innanzitutto scegliamo quale LED utilizzare: sulla demoboard abbiamo disponibile LD9, che è collegato direttamente sul pin PD7 del microcontrollore tramite la resistenza di limitazione di corrente R16. Poiché LD9 è collegato verso massa, potremo accenderlo mandando PD7 a +5V (1 logico) e spegnerlo mettendo PD7 a 0V (0 logico). Una volta eseguiti questi collegamenti possiamo scrivere un primo listato del programmino: per farlo



Fig. 2



Fig. 3

creiamo innanzitutto in AVR Studio un nuovo progetto, chiamandolo "ControlloLED": sul "Project Wizard" di AVR Studio (da barra menu: Project → Project Wizard) clicchiamo sul tasto "New project", quindi su "AVR GCC" e compiliamo la finestra come riportato in **Fig. 3**; completiamo cliccando su "Finish".

C:\AVR\ControlloLED\ControlloLED.c

```
#include <avr/io.h>

void main(void) {

/* Inizializzazione */

    DDRA = DDRB = 0xFF;  /* Tutti i pin di PORTA e PORTB sono impostati come uscite */
    DDRC = 0b10111100; /* PC0, PC1 e PC6 sono ingressi */
    DDRD = 0b11110011; /* PD2 e PD3 sono ingressi */

/* Fine inizializzazione */

    PORTD = 0x80; /* scrive '1' su PD7 -> Led acceso */

    while (1) {
        /* main loop: ciclo infinito (vuoto in questo caso) */
    }
}
```

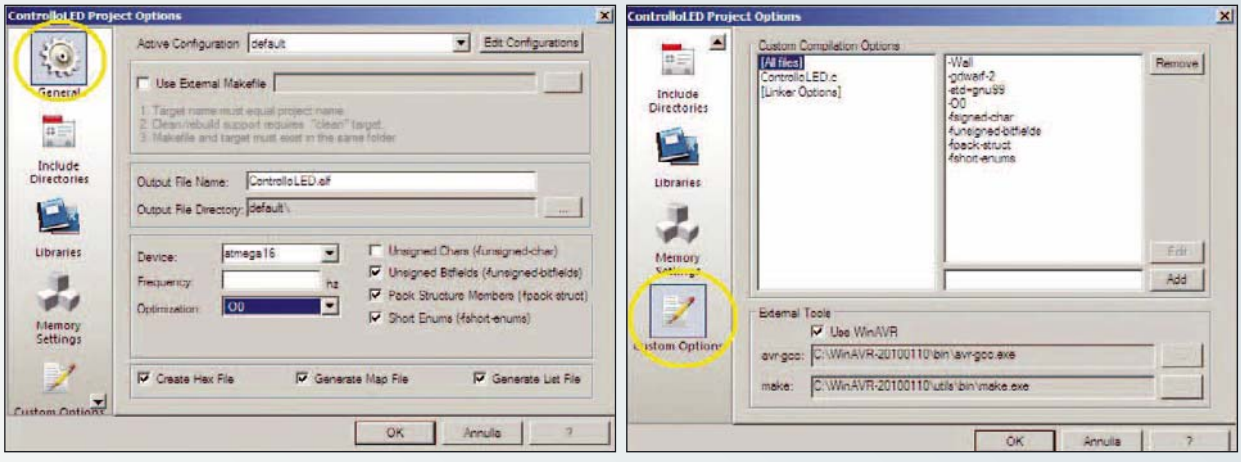
Fig. 4

Il nostro programma, visibile in **Fig. 4**, andrà immesso nella finestra di editor del sorgente (ControlloLED.c).
La spiegazione del suo funzionamento è molto semplice: all'interno della funzione principale `main()` eseguiamo innanzitutto la corretta inizializzazione della direzione dei piedini di tutti i port (ricordate quanto detto in proposito nella seconda puntata?) in accordo con le connessioni del micro nel nostro sistema. Dallo schema elettrico della demoboard risulta che tutti i pin dei port A, B, C, D devono essere configurati come uscite tranne PC0, PC1, PC6, PD2 e PD3 i quali, essendo collegati al potenziale di +5V tramite resistori di *pull-up*, possono essere impostati come ingressi anche se le periferiche a cui sono collegati non vengono (per il momento) utilizzate.
L'istruzione

```
PORTD = 0x80;
```

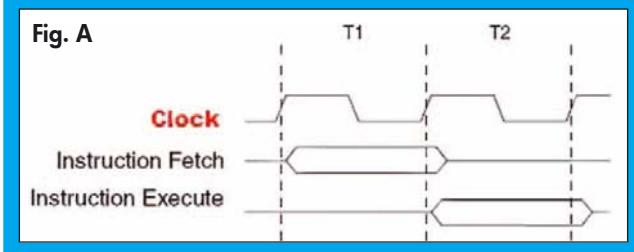
serve a scrivere '1' in PD7: infatti 0x80 (oppure `0b10000000`) equivale in binario a 1000 0000, ovvero mettiamo a '1' il bit più a sinistra (detto Most Significant Bit o MSB, il bit "più significativo" del byte) che, nella numerazione da 0 a 7, è appunto il settimo. Per completezza diciamo che analogamente il bit più a destra è detto LSB (da Least Significant Bit) ovvero bit meno significativo. Gli altri sette bit del port (PD<6:0>) vengono invece impostati a zero.
Scrivere '1' su PD7 significa che quando il processore eseguirà quell'istruzione il piedino omonimo verrà collegato alla tensione di alimentazione dell'ATMEGA16 (+5V) e il LED si accenderà.
Il programma entra infine in un ciclo (*loop*) ottenuto con l'espressione `while` (1), che si richiude su se stesso all'infinito, senza peraltro eseguire nulla (il corpo del ciclo, tra le parentesi graffe, è vuoto). È comunque necessario poiché sappiamo che nei microcontrollori la funzione principale `main()` non deve mai terminare.
Passiamo alla compilazione: innanzitutto, trattandosi di un nuovo progetto, bisogna preliminarmente impostare le opzioni di configurazione (da menu: Project → Configuration Options) come avevamo già visto nella terza puntata e che sono riassunte per comodità in **Fig. 5**.

Fig. 5

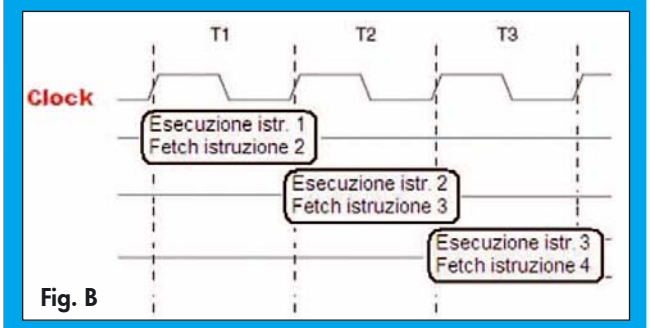


La velocità della **CPU**

Avevamo già visto che il cuore dei microcontrollori, la CPU, trasforma le istruzioni in arrivo dalla memoria di programma in azioni concrete attraverso le operazioni di lettura, decodifica ed esecuzione. Tali operazioni non avvengono in maniera istantanea e contemporaneamente, ma sono cadenzate dal clock (orologio) del sistema: esso è un segnale che, cambiando continuamente stato (0 → 1 → 0 → 1...) ad intervalli di tempo perfettamente regolari, gestisce la temporizzazione di tutte le fasi delle attività svolte della CPU. La figura seguente mostra come avviene un ciclo completo di esecuzione di un'istruzione, in relazione al segnale di clock, per la CPU dei micro AVR.



La prima fase è il prelievo dell'istruzione dalla memoria (*fetch*), che dura il tempo T1 pari ad un periodo completo del segnale di clock; la seconda fase è la decodifica ed esecuzione dell'istruzione, che impiega il tempo T2 coincidente anch'esso con un periodo di clock. Pertanto, per ogni istruzione il tempo impegnato dalla CPU vale complessivamente due periodi di clock. Nella realtà la nostra CPU è un pochino più sofisticata e permette, durante la fase di esecuzione di un'istruzione, di prelevare contemporaneamente il codice dell'istruzione successiva: grazie a questo doppio lavoro, il tempo effettivo di elaborazione di ogni istruzione viene dimezzato e va a coincidere esattamente con un periodo di clock.



Ma quanto vale questo periodo di clock? I microcontrollori ATMEGA permettono di scegliere la sorgente del segnale di clock tra diverse sorgenti, attraverso l'impostazione di alcuni *fuse bits* (CKSEL<3:0> come visto nella scorsa puntata) che impostiamo in fase di programmazione del microcontrollore: nei nostri esercizi, nella riga di comando di avrdude il valore dei fuse bit è dato dai parametri:

```
-U hfuse:w:0x99:m -U lfuse:w:0x2E:m
```

I bit CKSEL<3:0> sono i quattro meno significativi di lfuse, pertanto valgono "1110": in base al datasheet (paragrafo "Clock Sources"), con questi valori selezioniamo l'oscillatore collegato al quarzo esterno Q1. Poiché nella demoboard tale quarzo è da 16MHz, il periodo di clock vale 1/16.000.000 = 0,0625μs e questo è dunque il tempo che la CPU impiega per eseguire ogni singola istruzione. Naturalmente qui parliamo di istruzioni del processore in *linguaggio macchina*, quelle "a basso livello" per intenderci, e non a quelle che noi scriviamo nel codice sorgente in linguaggio C: una riga di codice C può essere tradotta dal compilatore in diverse istruzioni in linguaggio macchina dunque il tempo di esecuzione aumenta in proporzione. Nota: in realtà anche alcune istruzioni a basso livello possono durare più cicli di clock: per conoscerne il numero bisogna consultare il manuale "Instruction set" del processore.

Lanciamo quindi il comando Build (da menu: Build → Build, oppure da tastiera con F7, oppure ancora dall'icona relativa sulla barra degli strumenti): la finestra dei messaggi (in basso) si auto-posiziona sulla linguetta "Build" per mostrare i processi di compilazione e, se tutto 'va bene', apparirà infine la fatidica scritta:

```
Build succeeded with 1 Warnings...
```

Se andiamo a indagare scorrendo all'indietro la finestra dei messaggi scopriremo che il "warning" riportato, come già visto, è relativo al fatto che la funzione `main()` non ritorna 'int' ma 'void': funzionalmente non cambia nulla quindi possiamo ignorarlo.
Importante: controllate che fra i messaggi vi sia anche la scritta "`Device: atmega16`", in quanto può succedere che chiudendo e riaprendo il progetto, AVR Studio decida "autonomamente" di modificare il micro selezionato (che deve essere l'ATMEGA16, altrimenti i programmi non potranno funzionare quando li caricheremo sul micro vero e proprio); in caso contrario bisogna andarlo a selezionare nuovamente nella finestra di **Fig. 5**.
Completata con successo la compilazione, troverete che è stato creato all'interno della cartella "C:\AVR\ControlloLED\default" il file "ControlloLED.hex" che contiene il *codice oggetto* (ovvero le istruzioni del nostro programma tradotte in istruzioni nel *linguaggio* della CPU dell'ATMEGA16) da scaricare nella memoria programma del microcontrollore: per fare questo trasferimento (file su PC → memoria μC) ci serviremo del programmatore. Come avevamo già visto, il programma che utilizziamo per gestire le fasi di programmazione si chiama "*avrdude*", e funziona da linea di comando (non considereremo la variante grafica *avrdudeGUI* che pure è utilizzabile): conviene pertanto creare un *file batch* che esegua automaticamente le operazioni richieste ad *avrdude* in modo da evitare di immetterle manualmente tutte le volte. Per farlo, con "Esplora risorse" di Windows ci spostiamo nella cartella "C:\AVR\ControlloLED\default", clicchiamo col tasto destro del mouse e selezioniamo "Nuovo → Documento di testo" (**Fig. 6**).
Chiamiamo questo documento "ControlloLED.bat": Windows informa che modificando



Fig. 6

l'estensione del file esso potrebbe diventare inutilizzabile, ignoriamo l'avviso e confermiamo la scelta. Editiamo quindi questo file cliccandovi sopra col tasto destro e selezionando la voce "Modifica" (siccome è un file con estensione "*bat*", se vi cliccassimo con il tasto sinistro anziché editarlo lo manderemmo in esecuzione). Si apre un blocco note: andiamo a scrivervi (su un'unica riga):

```
avrdude -p m16 -c bsd -P lpt1 -u
-U flash:w:ControlloLED.hex -U
hfuse:w:0x99:m -U lfuse:w:0x2E:m
```

Salviamo infine il file, dopodiché possiamo chiuderlo.
A proposito del parametro "-P lpt1", se la porta parallela utilizzata non è la LPT1 bisognerà sostituire LPT1 con la porta corretta.
Apriamo ora un terminale (Prompt dei comandi) di Windows e posizioniamoci sulla cartella contenente il file batch appena creato digitando:

```
CD \AVR\ControlloLED\
default <invio>
```

Possiamo quindi effettuare la programmazione del microcontrollore scrivendo nel terminale:

```
ControlloLED <invio>
```

Il file batch passa in esecuzione, invocando a sua volta avrdude (con i parametri specificati), il quale esegue tutte le fasi richieste dalla programmazione; il loro esito viene riportato sul terminale, come mostrato in **Fig. 7**, sulla quale sono state chiaramente evidenziate.
Possiamo osservare che la prima operazione compiuta da avrdude è l'interrogazione del microcontrollore per verificare che esso sia un ATMEGA16, dunque compatibile col parametro 'm16' che avevamo specificato all'inizio.
La riga in cui compaiono numerosi caratteri cancelletto '#' serve a simulare una barra di progressione per evidenziare l'avanzamento dell'operazione in corso. Di fatto, l'identificazione del microcontrollore è molto rapida, di conseguenza si nota solo la riga ormai completa, con a fianco la percentuale di completamento 100% ed il tempo (approssimativo) effettivamente impiegato.
Segue l'operazione di cancellazione della memoria FLASH, indispensabile in quanto essa potrebbe già contenere un programma diverso dal nostro (non dimentichiamo che tale memoria si può riscrivere fino a 10.000 volte!), che va preliminarmente eliminato.

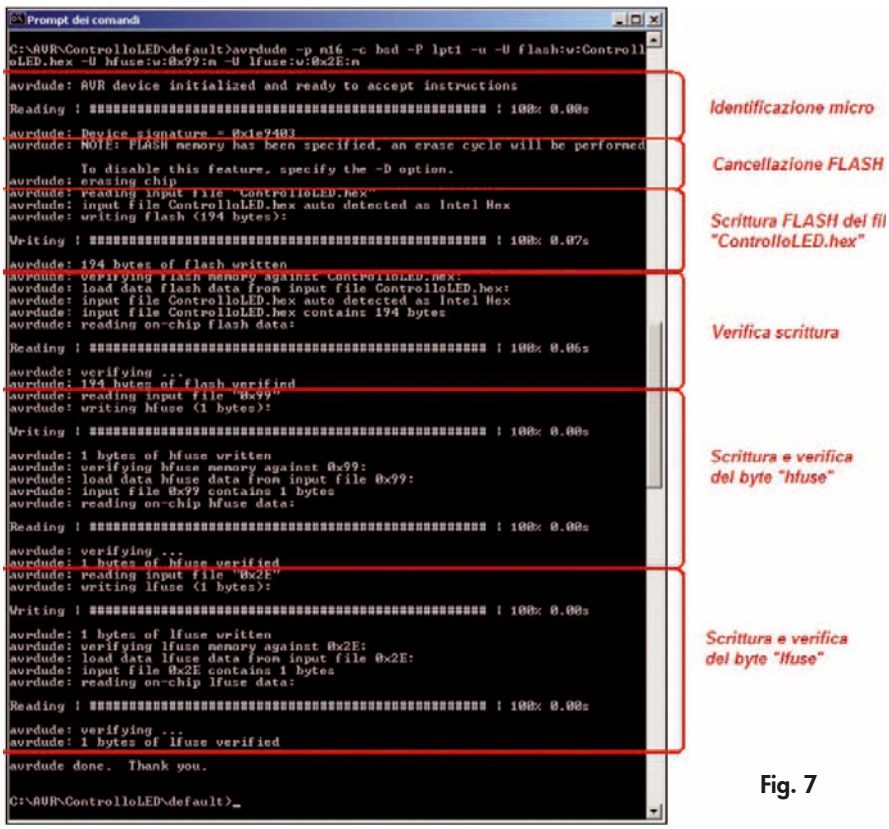


Fig. 7

Fatto ciò, *avrdude* va a leggere il file oggetto (ControlloLED.hex) presente sul PC e lo "riversa" byte per byte sulla FLASH del micro. La fase successiva è la lettura della FLASH stessa per verificare che contenga esattamente il programma che vi avevamo appena scritto; questa fase non è strettamente indispensabile ma caldamente consigliata, in quanto se a causa di un disturbo durante la scrittura viene alterato anche solo un bit della FLASH, questo significa cambiare un'istruzione o un dato del programma, che non funzionerà più. In tal caso, eseguendo la verifica, *avrdude* potrà comunicare che la programmazione non è andata a buon fine; l'utente può allora provare a ripetere l'operazione eseguendo nuovamente il file batch "ControlloLED.bat"; se l'errore persiste, può darsi che la memoria del micro sia danneggiata ed in tal caso occorre sostituire il dispositivo. Se però è la prima volta che si effettua l'operazione di programmazione, conviene prima verificare la correttezza dei collegamenti tra micro e programmatore.
Le ultime operazioni sono la scrittura (e la verifica) dei "fuse bits". Notare che se avrdude arriva a queste fasi, vuol dire che le operazioni precedenti sono andate a buon fine, altrimenti s'interrompe al primo errore. Pertanto per conoscere l'esito di tutto il processo di programmazione è sufficiente verificare che si arrivi alla programmazione (e verifica) di *lfuse*. La frase di cortesia finale "avrdude done. Thank you" compare invece sempre (anche in caso di errore!).
In ogni caso, se tutto è andato bene, al termine della programmazione avrdude fa partire immediatamente il nuovo programma, quindi vedrete il led LD9 accendersi: in tal caso congratulazioni, il programma funziona!

NUOVO ESPERIMENTO

Possiamo iniziare a complicare il programma, visto che siamo in grado di comandare l'accensione del led: facciamolo lampeggiare! Per fare ciò, basta ovviamente mandare consecutivamente a '1' e a '0' il piedino PD7; modifichiamo pertanto il programma sorgente di **Fig. 3** come mostrato nel **Listato 1**.
Molto semplicemente, dopo aver mandato PD7 a '1' vi è subito l'istruzione che lo riporta a '0'; poiché entrambe le istruzioni sono all'interno del ciclo 'infinito' while(1), esse verranno ripetute indefinitamente, ottenendo su PD7 la sequenza 0-1-0-1-0-1...
Compiliamo il programma con AVR Studio e quindi riprogrammiamo il micro chiamando nuovamente "ControlloLED.bat" dalla console del prompt dei comandi: avendolo già digitato prima, basta premere il tasto "freccia su" per far ricomparire il comando, e premere quindi il tasto Invio.
Dopo che *avrdude* ha compiuto il suo lavoro, possiamo osservare LD9: esso apparirà probabilmente ancora come se fosse costantemente acceso, dandoci l'impressione che il programma non stia funzionando correttamente. In realtà non è così: il programma funziona ed il LED lampeggia, però ad una velocità tale che i nostri occhi non sono in grado di distinguere tra l'istante nel quale il led è acceso e quando è spento. Infatti il microcontrollore impiega pochissimo tempo a decodificare ed eseguire ogni singola istruzione (tempo dell'ordine dei microsecondi, si veda il riquadro "La velocità della CPU") per cui possiamo pensare che il nostro LED si accenda e si spenga all'incirca un milione di volte al secondo: difficile da percepire, non vi pare? L'unico indizio osservabile del suo stato *on-off* è una luminosità leggermente inferiore a quella che aveva quando era acceso in modo continuativo, poiché il nostro occhio fa "la media" tra la luminosità che caratterizza lo stato acceso e lo stato spento.

Listato 1

```
/*********** Programmino per far lampeggiare LD9 **********/
#include <avr/io.h>
void main(void) {
/* Inizializzazione */
        DDRA = DDRB = 0xFF;  /* Tutti i pin di PORTA e PORTB come uscite */
        DDRC = 0b10111100; /* PC0, PC1 e PC6 sono ingressi */
        DDRD = 0b11110011; /* PD2 e PD3 sono ingressi */
/* Fine inizializzazione */
        while (1) { /* ciclo infinito */
                PORTD = 0x80; /* scrive '1' su PD<7> => LED acceso */
                PORTD = 0x00; /* scrive '0' su PD<7> => LED spento */
        }
}
```

Listato 2

```
/*********** Programmino per far lampeggiare LD9 **********/
#include <avr/io.h>
void main(void) {
        uint32_t ContaRitardo; /* variabile di conteggio nei cicli for */
/* Inizializzazione */
        DDRA = DDRB = 0xFF;  /* Tutti i pin di PORTA e PORTB come uscite */
        DDRC = 0b10111100; /* PC0, PC1 e PC6 sono ingressi */
        DDRD = 0b11110011; /* PD2 e PD3 sono ingressi */
/* Fine inizializzazione */
        while (1) { /* ciclo infinito */
                PORTD = 0x80; /* scrive '1' su PD<7> => LED acceso */
                for (ContaRitardo = 0; ContaRitardo < 300000; ContaRitardo ++)
                        ;
                PORTD = 0x00; /* scrive '0' su PD<7> => LED spento */
                for (ContaRitardo = 0; ContaRitardo < 300000; ContaRitardo ++)
                        ;
        }
}
```

Per poter osservare una corretta sequenza di accensioni e spegnimenti del led dobbiamo rallentare notevolmente la sequenza 0-1 su PD7, quindi abbiamo due possibilità:

1) Rallentiamo il clock della CPU del microcontrollore;
2) Gli facciamo "perdere tempo" tra l'esecuzione delle istruzioni di accensione e di spegnimento.

L'ipotesi 1) è irrealistica, poiché se utilizziamo l'oscillatore interno il suo clock più basso è comunque 1 MHz, che corrisponde ad una velocità di esecuzione potenziale di un milione di istruzioni al secondo ed inoltre verrebbero penalizzati i tempi di esecuzione di tutte le istruzioni presenti nel programma (oltre a quelle utilizzate per il lampeggio del LED).
Pertanto rimane l'opzione 2): dopo aver acceso il LED facciamo in modo che la CPU "ritardi un certo tempo X" prima di eseguire l'istruzione che lo spegne; analogamente, dopo averlo spento deve attendere un altro tempo X prima di riaccenderlo. Se ad esempio X = 0,5 secondi, otterremo che il LED sta acceso per mezzo secondo e spento per un altro mezzo secondo, dunque lo vedremo lampeggiare a 1 Hz.
Per ritardare il processore gli facciamo eseguire altre istruzioni, poiché sappiamo che per ognuna impiega un certo tempo, per quanto piccolo: così se un'istruzione dura 1 µs, se gliela facciamo eseguire 500.000 volte otteniamo il ritardo di mezzo secondo. Naturalmente non dobbiamo scriverla 500.000 volte nel programma ma utilizziamo *un ciclo*, ad esempio il "for", nella seguente maniera:

```
for (ContaRitardo = 0; ContaRitardo
< VALORE_MASSIMO; ContaRitardo ++);
```

La spiegazione è semplice: la variabile "ContaRitardo" viene innanzitutto inizializzata a zero, e poi incrementata fino a quando raggiunge il numero VALORE_MASSIMO (che nel programma verrà associato a un valore preciso). Per una spiegazione esaustiva del funzionamento dei cicli 'for' del C si può naturalmente far riferimento ad Internet, dove la documentazione è sovrabbondante; a noi basta sapere che questa riga di codice C si traduce in un certo numero di istruzioni per la CPU, che essa deve ripetere VALORE_MASSIMO volte per realizzare appunto l'incremento della variabile "ContaRitardo" da zero a VALORE_MASSIMO.
Per provare in pratica questi concetti, provate a modificare il programma come mostrato nel **Listato 2**.
In tale listato abbiamo inserito i cicli 'for' dopo le istruzioni di accensione e spegnimento del LED, sostituendo 300.000 al valore VALORE_MASSIMO: quindi in ciascun ciclo la variabile "ContaRitardo" viene incrementata da 0 a 300.000, causando un ritardo complessivo di esecuzione di circa mezzo secondo. Verificatelo dopo una nuova compilazione del programma e programmazione del micro: il LED lampeggerà a circa 1 Hz. A proposito di "ContaRitardo", essa è stata dichiarata all'inizio del programma come `uint32_t`, ovvero variabile intera senza segno (**uint** significa **u**nsigned **int**eger) a 32 bit, poiché per arrivare a 300.000 non poteva essere né da 8 bit (valore massimo 255) e nemmeno da 16 bit (massimo 65535).
Naturalmente potete provare a variare i numeri nei cicli 'for' per verificare come cambiano i tempi di accensione e spegnimento del LED in relazione ai loro valori.
Nota: a differenza del linguaggio ANSI C standard, la libreria del compilatore avr-gcc (chiamata *avr-libc*) ha una notazione

Listato 3

```
void FunzioneRitardo(void) /* dichiarazione della funzione */
{
/* corpo della funzione */
     uint32_t ContaRitardo; /* variabile di conteggio */
     for (ContaRitardo = 0; ContaRitardo < 300000; ContaRitardo ++)
;
} /* fine funzione */
```

molto semplice per indicare il tipo di variabili intere e la loro grandezza: "int**N**_t", dove **N** va sostituito con la dimensione voluta (8, 16, 32, 64). Se la variabile è senza segno (non codificata in complemento a 2), bisogna far precedere il nome con una 'u' (ad esempio uint8_t è una variabile da 8 bit senza segno). Per maggiori dettagli, consultate la guida della libreria avr-libc contenuta nel documento "avr-libc-user-manual.pdf" in "C:\WinAVR-20100110\doc\avr-libc".

LE FUNZIONI

Su questo programma, seppure molto semplice, possiamo già introdurre alcune migliorie, sia a livello di leggibilità che di efficienza.
Abbiamo visto che per rendere visualizzabile il lampeggio del LED dobbiamo scrivere nel sorgente due identici cicli 'for' di ritardo in due punti diversi del programma, con la conseguenza che il codice oggetto che il compilatore genera per il singolo ciclo 'for' verrà anch'esso duplicato nel file eseguibile del microcontrollore, con il risultato di una maggiore occupazione della memoria FLASH che dovrà contenerlo. Per evitare questo "spreco" di memoria si potrebbe cercare di raggruppare il codice del ciclo 'for' in un'area separata, dalla quale possa venire richiamato (dal programma principale) ogni volta che ve ne sia la necessità: il funzionamento complessivo non cambia, ma in questo modo il codice del ciclo 'for' viene scritto una sola volta nel file oggetto e quindi in memoria, con conseguente risparmio di spazio. Questo meccanismo è detto *chiamata a funzione*, dove per "funzione" s'intende la parte di codice relativa al ciclo 'for', che viene "chiamata" (ovvero eseguita) dal programma principale (che in C è la nostra funzione `main()`) quando e quante volte ce n'è bisogno.
La nostra nuova funzione va scritta come riportato nel **Listato 3**; il nome "FunzioneRitardo" è arbitrario, si può dare quello che si preferisce, basta che non contenga spazi: per esempio il nome "Pippo" va bene, "Funzione ritardo" no.
La sintassi della dichiarazione è simile a quella della funzione `main()`: il `void` tra parentesi indica che la funzione non riceve parametri dal chiamante mentre il `void` prima del nome indica che, quando termina, non restituisce nulla al chiamante. Nel corpo della funzione abbiamo riportato il codice del ciclo 'for', nonché la dichiarazione della variabile "ContaRitardo": se l'avessimo lasciata nel `main()` non sarebbe stata visibile all'interno di questa funzione ed il compilatore avrebbe segnalato l'errore; inoltre, nel `main()` questa variabile non viene utilizzata per null'altro.
Per chiamare la nuova funzione dal nostro programma principale dobbiamo scrivere, nei punti in cui vogliamo che venga eseguita:
`FunzioneRitardo();`
Quando il programma "incontra" questa istruzione (che è in realtà la *chiamata* alla funzione), passa ad eseguire il codice contenuto nel corpo della funzione stessa; al termine, il processore torna ad eseguire il codice del programma principale dal punto immediatamente successivo alla chiamata della funzione.
Il **Listato 4** riporta il codice completo del programma modificato con l'adozione della nuova funzione: in rosso sono visibili le righe relative alle modifiche introdotte.
Notate che prima dell'inizio della funzione `main()` è stata inserita una riga con la dichiarazione della funzione (detta *prototipo*) che verrà poi utilizzata nel corso del programma: questo serve per dire al compilatore, quando incontra nel `main()` le chiamate a `FunzioneRitardo()`, che esiste una funzione con tale nome, il cui codice tuttavia viene 'incontrato' dopo: infatti l'abbiamo scritto, nel file sorgente, dopo la funzione `main()`.
Si può notare come ne risulti un codice sorgente più "snello" e leggibile: una qualsiasi funzione, anche complessa, viene sostituita nella funzione principale `main()` con una sola riga di chiamata, in tutti i punti necessari, e magari chiamando la funzione stessa con un nome esplicativo del suo scopo.
L'altro vantaggio, come già detto, è una riduzione delle dimensioni del codice oggetto prodotto dal compilatore, tanto maggiore quante più volte la funzione dev'essere utilizzata dal programma principale.

Listato 4

```
/*********** Programmino per far lampeggiare LD9 **********/
#include <avr/io.h>
void FunzioneRitardo(void); /* prototipo della funzione di ritardo */
void main(void) { /* funzione principale */
/* Inizializzazione */
        DDRA = DDRB = 0xFF;  /* Tutti i pin di PORTA e PORTB come uscite */
        DDRC = 0b10111100; /* PC0, PC1 e PC6 sono ingressi */
        DDRD = 0b11110011; /* PD2 e PD3 sono ingressi */
/* Fine inizializzazione */

  while (1) {
        /* ciclo infinito */
        PORTD = 0x80; /* scrive '1' su PD<7> => LED acceso */
        FunzioneRitardo(); /* prima chiamata alla funzione di ritardo */
        PORTD = 0x00; /* scrive '0' su PD<7> => LED spento */
        FunzioneRitardo(); /* seconda chiamata alla funzione di ritardo */
  }
}
/* fine funzione principale main()*/

void FunzioneRitardo(void) { /* codice della funzione di ritardo */
        uint32_t ContaRitardo;
        for (ContaRitardo = 0; ContaRitardo < 300000; ContaRitardo ++)
                ;
}
```

DEFINIZIONE DI COSTANTI

Abbiamo visto che la velocità del lampeggio varia cambiando il numero massimo di conteggi del ciclo 'for'. Tale numero si trova 'annidato' dentro al programma; la prima impressione è che non sia di immediata reperibilità, specialmente se vi trovaste a doverlo modificare dopo qualche mese da quando l'avete scritto. Per migliorare la leggibilità, in C è possibile definire i valori costanti utilizzati dal programma direttamente all'inizio del programma stesso utilizzando la seguente sintassi:

```
#define RITARDO_CICLO 300000
```

La riga definisce una costante chiamata "RITARDO_CICLO" alla quale abbiamo assegnato il valore 300.000. Se la scriviamo in cima al nostro sorgente (prima della funzione `main()` per intenderci), essa potrà in seguito essere utilizzata ovunque nel programma, così ad esempio nel ciclo 'for' potremo scrivere:

```
for (ContaRitardo = 0; ContaRitardo
< RITARDO_CICLO; ContaRitardo ++);
```

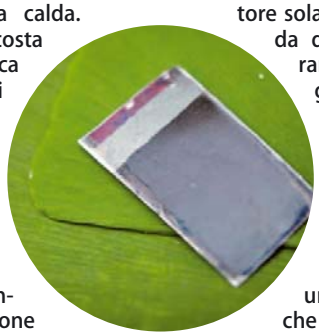
senza alterare il funzionamento del programma: il compilatore (più esattamente il preprocessore) provvederà a sostituire RITARDO_CICLO con 300.000 ovunque essa venga richiamata. Il vantaggio di questa soluzione è che avendo definito il numero in una riga ad inizio programma, è più facile da trovare e modificare a piacere, soprattutto se nel programma la costante in questione viene usata più volte (nel nostro listato 2 ad esempio era usata due volte, per i due cicli 'for'), lasceremo così al compilatore il compito di trovare ed operare le sostituzioni per tutte le occorrenze della costante all'interno del programma. Da osservare che, nel linguaggio C, i nomi che fanno riferimento a delle costanti vengono scritti utilizzando esclusivamente lettere maiuscole: nel nostro caso abbiamo infatti "RITARDO_CICLO" e non "Ritardo_Ciclo" oppure "ritardo_ciclo". Questa non è una regola del linguaggio (il compilatore non dà errore scrivendoli minuscoli) ma una convenzione universalmente riconosciuta, allo scopo di rendere le costanti più evidenti all'interno del programma a vantaggio della leggibilità.

Per completezza è giusto accennare al fatto che `#define` è una *direttiva* del preprocessore del compilatore C che permette, oltre alla semplice dichiarazione di costanti, di definire vere e proprie *macro*, ovvero intere righe di codice che poi vengono sostituite nel programma in modo analogo a quanto abbiamo visto per le costanti; rimandiamo alla documentazione sul linguaggio C per un eventuale approfondimento.

Terminiamo quindi questa puntata pratica sul microcontrollore nella quale abbiamo visto come gestire lo stato di un piedino di I/O del microcontrollore, in questo caso per accendere un LED, e l'utilizzo delle funzioni in un programma. Dalla prossima lezione passeremo ad affrontare lo sviluppo di applicazioni più complesse grazie anche all'utilizzo delle periferiche interne dell'ATMEGA16 che inizieremo ad esplorare.Vi aspettiamo!

Foglie artificiali producono idrogeno

L'idrogeno è il combustibile del futuro: se bruciato nei motori non produce inquinanti, mentre introdotto nelle fuel-cell consente di ottenere elettricità restituendo acqua calda. Purtroppo produrlo costa ancora molto e implica l'uso delle fonti fossili come metano e petrolio; esiste l'alternativa dell'elettrolisi dell'acqua, ma l'energia elettrica consumata è superiore a quella ottenibile dall'uso dell'idrogeno come combustibile. L'unica soluzione per avere idrogeno a basso costo e senza inquinare è impiegare processi artificiali di fotosintesi, come quello che il professor Daniel Nocera, docente di chimica al Massachusetts Institute of Technology, ha sviluppato basandosi su un prototipo di foglia artificiale. Quello di Nocera non è il primo dispositivo in grado di realizzare una fotosintesi artificiale, però quelli approntati finora richiedono catalizzatori a base di platino e altri ossidi di metalli nobili, nonché processi di produzione che ne rendono i costi proibitivi. Invece Nocera è riuscito a realizzare un collettore solare a sandwich costituito da due pellicole che generano ossigeno e idrogeno gassosi: nel primo, il catalizzatore di platino che produce idrogeno è sostituito da un composto economico a base di nichel-molibdeno-zinco; sull'altro lato della "foglia" è applicata una pellicola di cobalto, che permette la liberazione dell'ossigeno. Le foglie artificiali del professor Nocera possono diventare la soluzione per ottenere idrogeno dall'energia solare, senza passaggi intermedi come la produzione fotovoltaica e la successiva elettrolisi dell'acqua, con evidenti risparmi in fatto di costi e ingombri degli impianti.



http://web.mit.edu/newsoffice/2011/artificial-leaf-0930.html

Rallenta, ma produce corrente

Chiunque guidi un'automobile detesta i dossi di rallentamento e non tanto perché costringono a moderare l'andatura, quanto per il contraccolpo che la vettura subisce nell'attraversarli, ed anche per l'aumento del consumo dei freni e di combustibile (rallentare e accelerare fa incrementare il consumo del motore rispetto a procedere ad andatura costante). Ma se proprio bisogna rallentare il traffico vei-



53.000 km sull'onda del fotovoltaico

Turanor, il catamarano di PlanetSolar, è rientrato trionfale dal suo giro del mondo iniziato nel lontano 27 settembre 2010. Questa sarebbe una notizia qualsiasi, se non fosse che l'imbarcazione, il cui nome è stato preso in prestito dal best-seller "Il signore degli anelli", ha navigato per 584 giorni sospinta da motori elettrici alimentati esclusivamente dall'energia solare. Il catamarano, ideato dallo svizzero Ra-



colare, ad esempio in prossimità di una scuola, perché non trarne beneficio e recuperare l'energia che si fa sprecare ai veicoli? Questa è l'idea che ha mosso UP (Underground power) l'azienda, che ha realizzato "Lybra".
Si tratta di una soluzione di Energy Harvesting mascherata da dosso rallentatore stradale, capace di produrre elettricità mediante una pedana mobile che si abbassa al transito del veicolo, azionando un sistema piezoelettrico che produce ogni volta un impulso di corrente. La pedana ha un'altezza di 10 centimetri e può essere anche interrata, per esser tutt'uno con il manto stradale.
Sopra il dosso è applicata una copertura in gomma riciclata proveniente da vecchi pneumatici, che garantisce isolamento e impermeabilità al sistema elettrico sottostante, ma anche aderenza ai veicoli che vi transitano. Il generatore integrato nel dosso produce 1 kWh di elettricità ogni 100 veicoli che vi passano sopra.
L'elettricità prodotta può essere sia utilizzata o immagazzinata localmente per far funzionare infrastrutture stradali (segnaletica luminosa, pannelli informativi, ricetrasmittenti per il monitoraggio dei dati e la comunicazione delle emergenze) e illuminazione, sia convogliata verso la rete elettrica.

www.upgen.it



phael Domjan, è tornato al porto di Montecarlo dopo aver percorso ben 53.000 km toccando le coste di Cancun, San Francisco, Tonga, Brisbane, Manila, Singapore e Abu Dhabi. Costruito a Kiel, nella Germania del nord, Turanor è lungo circa 31 metri e largo 15, pesa oltre 90 tonnellate e le superfici del suo ponte e della cabina sono ricoperte da 537 metri quadrati di pannelli solari fotovoltaici, che hanno permesso ai suoi motori di spingerlo fino alla velocità di 15 nodi, ossia circa 27 km orari. La sua realizzazione è costata 24 milioni di dollari, ma la soddisfazione che ha dato ai sei membri dell'equipaggio che hanno partecipato all'impresa non ha prezzo. Inestimabile è anche il valore dell'impresa, per ciò che può significare nell'evoluzione del trasporto marittimo ecosostenibile.

www.planetsolar.org



La turbina eolica vola in alto

Dispositivi come Kitenergy e KiteGen ci hanno insegnato che in quota l'energia eolica si sfrutta meglio; su questo principio, l'americana Altaeros Energies ha realizzato un rivoluzionario aerogeneratore a turbina che viene mantenuto sospeso in aria grazie al suo involucro. Non si tratta di un aquilone eolico che sale e scende sfruttando l'energia del vento azionando (con un cavo) un generatore elettrico: questo è una sorta di dirigibile, ovvero un aerostato a forma di ventola intubata, il cui involucro è un manicotto riempito di gas elio, più leggero dell'aria. All'interno si trova un aerogeneratore a turbina da 2,5 kW, con rotore tripala da 3,7 metri di diametro, che viene orientato nella direzione in cui il vento soffia più forte; l'aerostato è ancorato al suolo mediante tiranti, che permettono di farlo sospendere ad un'altezza di circa 300 metri. L'elettricità prodotta viene trasmessa a terra per mezzo di cavi; le operazioni di sollevamento e atterraggio dell'aerostato sono completamente automatizzate e adattate alle condizioni del vento.
L'aerogeneratore dell'Altaeros può essere utilizzato nei luoghi isolati non serviti dalla rete elettrica, negli avamposti militari o anche negli insediamenti delle spedizioni scientifiche, in luoghi lontani dalla civiltà.
Dopo aver testato un prototipo funzionante l'azienda sta lavorando a una versione di dimensioni maggiori, destinata alle piattaforme off-shore; sul sito della Altaeros Energies è disponibile un video che mostra il generatore in funzione.

www.altaerosenergies.com



25 MW geotermici per gli USA

Enel Green Power, attraverso la sua controllata Enel Green Power North America, ha avviato i lavori per la realizzazione dell'impianto geotermico di Cove Fort, situato nello Utah meridionale (Stati Uniti). Il nuovo impianto, con una capacità installata di 25 MW, sarà in grado di produrre circa 160 GWh di energia ogni anno, evitando l'emissione in atmosfera di circa 115.000 tonnellate di CO_2 all'anno. L'investimento complessivo per la realizzazione dell'impianto, la cui entrata in esercizio è prevista entro il 2013, è di circa 126 milioni di dollari.
Non è la prima volta che Enel Green Power realizza opere negli Stati Uniti: infatti ha già installato due impianti a tecnologia binaria in Nevada, di cui uno (Stillwater) è abbinato, primo caso al mondo, a un impianto fotovoltaico (si tratta del primo impianto ibrido in grado di unire la capacità di generazione continua della geotermia a ciclo binario a media entalpia con la capacità di picco del solare). L'installazione di Cove Fort andrà ad aggiungersi alla capacità geotermica netta già installata da Enel Green Power negli USA, che attualmente ammonta a 47 MW.

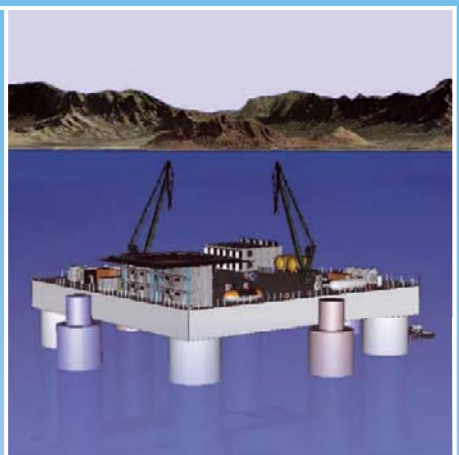
www.enelgreenpower.com

2 MW di fotovoltaico a concentrazione

È stato completato, a Orta Nova (FG), il più grande impianto d'Italia e d'Europa realizzato con la tecnologia fotovoltaica a concentrazione. L'impianto, secondo a livello mondiale, per il nostro Paese rappresenta il passaggio dalla fase sperimentale (prima erano installati solo 3 impianti per complessivi 30 kW) all'impiego effettivo del fotovoltaico a concentrazione. L'impianto pugliese è doppiamente innovativo, perché in esso si coniuga la tecnologia a bassa concentrazione (LCPV) con il sistema ad inseguimento a singolo asse MX1, frutto della collaborazione tra Convert Italia S.p.A. e l'azienda statunitense Solaria Corporation.

I moduli fotovoltaici a bassa concentrazione utilizzati, sono stati progettati esplicitamente per i sistemi ad inseguimento ed il loro vetro protettivo è anche un'efficace lente del modulo; il vetro, infatti, concentra l'irraggiamento solare su celle di silicio monocristallino. Quello a concentrazione, è il fotovoltaico su cui più si concentrano attenzioni e investimenti, sia per l'elevata resa a parità di superficie e di semiconduttore richiesto per unità di potenza, sia perché il IV° conto energia assegna incentivi più sostanziosi proprio agli impianti basati su tale tecnologia.

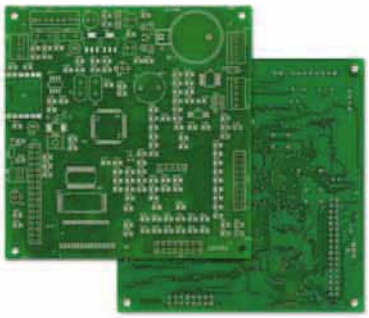
www.convertitalia.com



Elettricità dal gradiente marino

Gli oceani presentano un notevole gradiente termico fra la superficie e le acque profonde (ad esempio nelle zone tropicali si passa dai 29° ai 5° dal pelo dell'acqua a 1.000 m di profondità) che può essere sfruttato per produrre elettricità. Ne sono convinti Lockheed Martin, il colosso americano del settore aereospaziale, e la Ocean Thermal Energy di Lancaster, in Pennsylvania, che insieme stanno provando a migliorare la tecnologia OTEC (Ocean Thermal Energy Conversion) nata oltre trent'anni fa. Il meccanismo OTEC prevede che le acque in superficie riscaldino un fluido, come l'ammoniaca, avente un punto di ebollizione inferiore alla loro temperatura; quando il fluido bolle, il gas ottenuto esercita una pressione sulla turbina, producendo energia elettrica. A questo punto il gas viene fatto raffreddare grazie all'acqua gelida pompata da 1 km di profondità e, una volta fatto condensare, è pronto per ripetere il ciclo termodinamico.
Su questo principio, Lockheed Martin sta realizzando alle Hawaii un impianto da 10 MW che dovrebbe essere operativo entro il 2015. Il successo dell'operazione dovrebbe dare il via alla costruzione di una vera centrale elettrica da 100 MW entro il 2020. Il problema maggiore consiste nel realizzare un tubo che sia capace di resistere almeno per un decennio all'erosione dell'acqua e che sia al contempo abbastanza lungo e largo da portare in superficie la giusta quantità di acqua. Gli ingegneri stanno già lavorando alla sua realizzazione, ma al momento i costi richiesti sono davvero esorbitanti, si parla di 1 miliardo di dollari.

www.lockheedmartin.com



SINGOLA E DOPPIA FACCIA CON FORO METALLIZZATO; CON SOLDER E SERIGRAFIE PER UNO STAMPATO DI ALTA QUALITÀ O SOLO PISTE STAGNATE PER UN PROTOTIPO A BASSO COSTO

I NOSTRI CIRCUITI STAMPATI HANNO DEI COMPONENTI UNICI. SU MISURA PER VOI.

velocità

TEMPO DI CONSEGNA GARANTITO:
24 ORE O I CIRCUITI SONO **GRATIS**

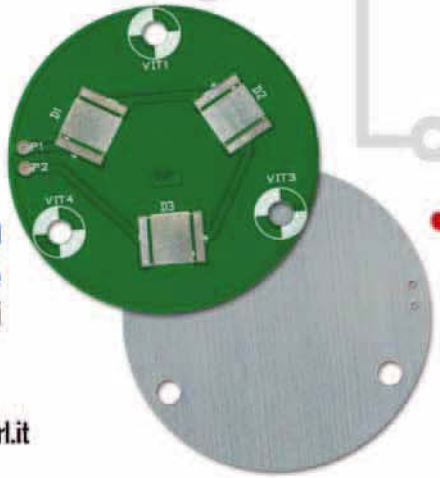
qualità

I CIRCUITI VENGONO SOTTOPOSTI A **TEST ELETTRICO** E ACCOMPAGNATI DAL **CERTIFICATO DI GARANZIA***;
TUTTI I CIRCUITI SONO REALIZZATI SU LAMINATO **Panasonic**

novità

DA OGGI, ALLO STESSO PREZZO E ALLE STESSE CONDIZIONI DEI NOSTRI CIRCUITI DOPPIA FACCIA È DISPONIBILE **IL LAMINATO IN ALLUMINIO** AD ALTISSIMA DISSIPAZIONE TERMICA INDISPENSABILE PER LA **REALIZZAZIONE DI PROGETTI CON LED** AD ALTA LUMINOSITÀ E OVUNQUE SIA RICHIESTO UN ELEVATO GRADO DI DISSIPAZIONE





facilità

PER CHIARIMENTI, DETTAGLI SULLE NOTE TECNICHE, ORDINI:
www.mdsrl.it
PREVENTIVO ANONIMO, GRATUITO E IMMEDIATO

millennium dataware
produciamo circuiti stampati

Parco scientifico e tecnologico
15050 Rivalta Scrivia - Tortona (AL)
tel. 0131 860.254 fax 0131 860157 info@mdsrl.it

*la certificazione sarà rilasciata solo per circuiti con consegna oltre le 48 ore

La Bacheca degli annunci

Vendo:

Da privato "soddisfatti o rimborsati", spedisco pagate alla consegna scontati per urgente realizzo:

- Radiocuffia stereo multifunzione multiaccessoriata Pioneer anche per uso investigativo.
- Microspia americana FM 40,68MHz con auricolare ricevente e certificato di conformità (campionatura).
- Campanello senza fili con funzioni manuali o automatiche (campionatura).
- Proiettore laser a fascio visibile rosso + centralina multieffetti a 3 uscite con portafaretti + lampada di Wood (campionature).
- Accumulatore al gel a tracolla con multiuscite multitensioni ad uso contemporaneo accessoriato tedesco.
- Mini megafono compatto tipo torcia con altoparlante a membrana piatta (campionatura).
- Variatore di velocità professionale fino a 1.320W + variatore universale fino a 1.800W.
- Radiomicrofono con microricevitore a pile.
- Microscopio tascabile tedesco da superficie 100 ingrandimenti con luce per microelettronica, micromeccanica, ecc... (campionatura).
- Vibrografo a battute regolabili + micro incisore multialimentato accessoriato.
- Micro trapano con accessori multialimentato + saldatore ceramico rapido a temperatura regolabile.

Possiedo inoltre utili dispense tecniche collezionabili dal 1900 in poi, che, invecchiandosi, si valorizzano.
Contattare Pietro al numero di telefono 0371-30418.

Eseguo:

- Frontalini in policarbonato anche a pezzi unici senza costi di attrezzatura.

Vendo:

- Kit per la stampa in proprio di mascherine in policarbonato composto da 2 fogli formato A4 di speciale policalicarbonato 250 micron stampabile Ink Jet, 2 fogli di biadesivo formato A4, 1 bomboletta di vernice di fondo, istruzioni dettagliate che vi consentono di avere in 10 minuti le vostre mascherine. ideale per progetti.

Contattare Roberto al 338-4317770.

Vendo:

- Saldatore da 50W con regolazione della temperatura;
- Tester digitale (7 campi di misura e 32 portate);
- Trapanino da 12V con alimentatore per forare le basette;
- Terza mano con lente d'ingrandimento;
- Riviste di elettronica.

Il materiale è nuovo, non è mai stato usato. Il tutto a 50 euro. Contattare Francesco al numero di telefono 347-4133862.

Offro:

- Collaborazione anche a progetto o a tempo determinato a ditte del settore elettronico in tutta Italia (sono tra Milano e Salerno).

Realizzo:

- Schede finite, prevalentemente digitali a microcontrollore (Microchip), su specifiche del cliente. Schema, sbroglio del PCB, sviluppo di firmware in Assembler, part list, assemblaggio anche SMD.

Eseguo:

- Riparazioni ed installazioni elettroniche, tecnico per assistenza.

Vendo:

- Duplicatori di EEPROM per serie 24xxx, 93xxx, MDA2061/2 e NVM3060, 27xxx, 28xxx, 29xxx, 49xxx, PIC12C508, PIC16F84, ecc.

Escluso uso hobbistico.

Contattare Vittorio all'email: elettronica.argento@tiscali.it o al 347-3810787.

Sviluppo:

Firmware per microcontrollori PICXX, STXX e Motorola su specifica e realizzo prototipi. Telefonare a Gianni allo 0376-396743.

Vendo:

- Ricetrasmettitore standard C120 (palmare) per i 144MHz con microfono esterno, antenna extra a stilo, contenitore per batterie stilo, manuale in italiano e inglese (originale) perfettamente funzionante. Euro 70,00;
- Ricetrasmettitore standard SR146 per i 144 MHz a quarzi 5 canali più 14 quarzi per ponti radio microfono esterno, antenna extra in gomma, manuale custodia in cuoio, perfettamente funzionante. Apparato adatto anche per i collezionisti. Euro 50,00;

Contattare Vincenzo al 328-9655192.

Cerco:

Nella zona geografica tra Latina nord e Roma sud, laboratorio elettronico o persona specializzata nella installazione elettronica e funzionamento di macchina CNC (pantografo a portale).
Contattare Francesco allo 06-9853114 o inviare e-mail: fracap@iol.it

Cerco:

- 4 condensatori elettrolitici a secco per radio a valvole (1930) da 8uF - 500Vp -575Vl;
- scatola di montaggio completa di schemi, mobile e materiale elettronico per radio a valvole supereterodina - 2 gamme d'onda.

Rispondere a Leonardo a leopers@libero.it o al 338-2138943.

Questo spazio è aperto gratuitamente a tutti i lettori. La Direzione non si assume alcuna responsabilità in merito al contenuto degli stessi ed alla data di uscita. Gli annunci vanno inviati via **FAX** *al numero* 0331-792287 *oppure tramite* **INTERNET** *connettendosi al sito* www.elettronicain.it *nella sezione "Annunci".*

Per te mai più segreti!

Nuovi dispositivi professionali per misure e analisi precise.



Analizzatore di componenti DCA55

Indispensabile in laboratorio, consente di analizzare una vasta gamma di componenti attivi, identificando la pinout e fornendo informazioni relative ai principali parametri.

€75,00
cod. DCA55



Componenti analizzabili:



Multimetro digitale con USB - 6600 conteggi



€124,00
cod. DVM4200

Strumento professionale con interfaccia USB e display LCD retroilluminato collegabile al PC per registrare, elaborare e stampare i dati misurati. Dispone di selezione manuale/automatica delle portate, misura corrente (10A) e tensione (1000V), resistenza (66Mohm), capacità (66mF), frequenza (66MHz), temperatura (1000°C) ed esegue test su diodi e di continuità elettrica. Completo di software "PC-LINK", custodia e vari accessori.

Disponibile anche nelle versioni:

Multifunzioni 6000 conteggi



Dotato di porta USB, misura tensione (1000V), corrente (10A), resistenza (60Mohm), capacità (300µF), frequenza (999,9kHz) e temperatura (1000°C).

€89,00
cod. DVM4100

Stesse caratteristiche del DVM4100 ma senza porta USB.

€67,00
cod. DVM4000

FUTURA ELETTRONICA
www.futurashop.it
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

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

Prezzi IVA inclusa.

Sistemi di illuminazione a LED

Soluzioni per il risparmio energetico adatte ad ogni ambiente.



Pannello illuminante 112 LED



€ 115,00
cod. P3030-W

- Flusso luminoso 880 lm
- Consumo 12 W
- Dimensioni 29,5 x 29,5 cm

Disponibile anche nella versione a 416 LED, 3000 lm, 45 W, 59,5 x 59,5 cm.
cod. P6060-W **€ 239,00**

Tubo luminoso 56 LED



€ 48,00
cod. T8-HWSMD

- Flusso luminoso 840 lm
- Consumo 10 W

Pannello da incasso 36 LED



€ 72,00
cod. P1717-WW

- Flusso luminoso 880 lm
- Consumo 16 W
- Dimensioni 17 x 17 cm

Faro da incasso 12 LED



€ 96,00
cod. D1202-WW

- Flusso luminoso 1410 lm
- Consumo 28 W
- Diametro 16 cm

Lampada a LED - E27



€ 39,00
cod. G60ME27-9W-WW

- Flusso luminoso 900 lm
- Consumo 9 W

Lampada 72 LED - E27



€ 27,00
cod. CORN72SMD-WW

- Flusso luminoso 900 lm
- Consumo 9,5 W

Prezzi IVA inclusa.

FUTURA ELETTRONICA

www.futurashop.it

Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Maggiori informazioni su questi prodotti e acquisti on-line su *www.futurashop.it*.