

Elettronica **In**

Mensile di elettronica applicata, attualità scientifica, novità tecnologiche.

112

Diffusione audio con wireless digitale

Teoria e pratica
delle Smart Card

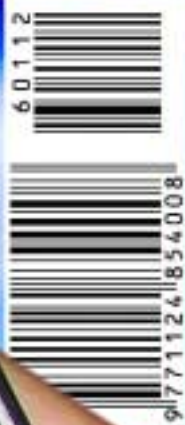
Interfaccia USB per PC: il software

Telecontrollo con cellulare Bluetooth

Tibbo FTP per dati meteo

Robot FOX: lo sguardo all'infrarosso

Piove, tira vento: metti la tenda sotto controllo



ESCLUSIVO
CORSO
CAN-BUS



VELBUS, LA DOMOTICA PER TUTTI

Con il sistema modulare Velbus la domotica entra finalmente in tutte le case. Studiati per l'automazione domestica, i moduli Velbus rendono la vita più semplice e sicura permettendo un controllo completo di tutti i dispositivi elettrici di casa, dall'impianto di illuminazione, alle serrande, dall'antifurto alle tende elettriche. Velbus può essere utilizzato sia in nuove abitazioni che in unità immobiliari già esistenti. Tutti i moduli sono connessi tra loro mediante un cavo a quattro conduttori che trasporta i dati e la tensione di alimentazione (12Vdc). Grazie alla tecnologia digitale è possibile ottenere funzioni che un impianto tradizionale non potrà mai fornire, dalla possibilità di attivare/disattivare con un solo pulsante tutte le luci di casa o alzare tutte le serrande, alla creazione di scenari d'ambiente predefiniti. Ai 10 moduli attualmente disponibili si aggiungeranno altri dispositivi per il controllo della temperatura, sistemi di sicurezza, controllo accessi, ecc.

CARATTERISTICHE PRINCIPALI:

- Apprendimento funzione per ciascun tasto (apprendimento manuale);
- Possibilità di controllare, configurare e monitorare il sistema tramite PC;
- Sicurezza garantita anche in assenza di alimentazione;
- Funzione timer su moduli d'uscita a relé.

Scopri tutti i segreti del CAN-BUS con il Corso a teorico/pratico pubblicato sulla rivista "Elettronica In" e scaricabile dal sito: www.elettronica.in.it.

VELBUS:

- 2 fili sono utilizzati per la trasmissione dati VELBUS e 2 per l'alimentazione;
- Velocità trasferimento dati: 16,6 Kbit/s;
- Protocollo di comunicazione seriale: CAN (Controller Area Network);
- Protezione da corto circuito (verso positivo o negativo dell'alimentazione);
- Auto ripristino da bus error dopo 25 secondi.

MODULO RELÉ



- Contatti relé con soppressore transienti: 5A / 230Vac max.
- 10 differenti modalità di funzionamento.

VMB1RY ~ Euro 90,00

MODULO RELÉ 4 canali



- 4 uscite indipendenti a relé con soppressore: 5A / 230Vac max.
- 10 modalità di funzionamento impostabili per ciascun canale.

VMB4RY ~ Euro 149,00

MODULO di CONTROLLO per SERRANDE



- Utilizzabile per il controllo di serrande, tende da sole motorizzate
- Contatto relé con soppressore transienti: 5A / 230 Vac max.

VMB1BL ~ Euro 80,00

MODULO di CONTROLLO per SERRANDE ~ 2 canali



- Può essere utilizzato per controllare fino a 2 tende da sole motorizzate, serrande ecc.
- Contatto relé con soppressore transienti: 5A / 230Vac max.

VMB2BL ~ Euro 101,00

MODULO DIMMER



Ideale per il controllo di lampade ad incandescenza, alogene a tensione di rete o a bassa tensione con trasformatore controllabile da dimmer.

VMB1DM ~ Euro 92,00

INTERFACCIA SERIALE



- Permette il controllo del sistema VELBUS tramite PC;
- Comunicazione di tipo RS232 Full duplex;
- Isolamento galvanico tra PC e sistema VELBUS.

VMB1RS ~ Euro 44,00

MODULO CON INTERFACCIA USB



- Consente di gestire e controllare, mediante PC, i moduli per l'automazione industriale e la domotica.
- Isolamento galvanico tra PC e sistema VELBUS.

VMB1USB ~ Euro 53,00

MODULO ALIMENTATORE



- Tre uscite a 12 VDC/1A per alimentazione moduli VELBUS;
- Protezione da corto circuiti.

VMB3PS ~ Euro 36,00

MODULO 6 ingressi



Permette di collegare al sistema VELBUS, pulsanti, contatti di porte e finestre, uscite a relé di sensori di movimento, uscite open-collector, ecc.

VMB6IN ~ Euro 80,00

MODULO PULSANTE 8 canali



Permette di collegare qualsiasi tipo di pulsante al sistema VELBUS.

VMB8PB ~ Euro 40,00

Tutti i prezzi si intendono IVA inclusa.

mostra mercato

Expo[®] 2006 Elettronica

ERBA 11-12 novembre
(Como) ore 9/18

 **LARIO
FIERE**
V.le Resegone, 3



organizzazione
BLU NAUTILUS srl
tel. 0541 439573
www.blunautilus.it

dalle ore 10,30
Dimostrazioni e
spiegazioni di utilizzo
di software libero



a cura di GLUX
Gruppo Lecco Utenti linux
www.lecco.linux.it
con il supporto tecnico di yetopen.it

Per ottenere un **INGRESSO
RIDOTTO** presenta questa
inserzione alla cassa



Pag. 48

Pag. 67

16

ROBOT FOX, LO SGUARDO ALL'INFRAROSSO

Mini automa in grado di muoversi liberamente e di evitare gli ostacoli grazie ai sensori di prossimità all'infrarosso di cui è dotato. Dispone anche di sensori che gli permettono di rilevare quando viene a mancare la superficie di appoggio. Grazie ad un modulo radio può essere controllato anche tramite PC.

25

TIBBO FTP PER DATI METEO

Utilizziamo l'opportunità offerta dalla programmabilità dei Device Server della Tibbo per realizzare un sistema in grado di pubblicare via FTP su qualsiasi pagina Web i dati meteorologici forniti dalla WS2300, la più nota centralina meteo della La Crosse Technology. Seconda puntata.

35

INTERFACCIA USB PER PERSONAL COMPUTER, IL SOFTWARE

Rendiamo operativa l'interfaccia USB per PC presentata nel numero precedente utilizzando il software di gestione per PC da noi messo a punto. Analizziamo anche le caratteristiche della DLL da utilizzare per lo sviluppo di applicativi personalizzati in Delphi, Visual Basic e Borland C++.

S o m m a r i o

ELETTRONICA IN

www.elettronica.in
Rivista mensile, anno XII n. 112
OTTOBRE 2006

Direttore responsabile:

Arsenio Spadoni
(Arsenio.Spadoni@elettronica.in)

Redazione:

Gabriele Daghetta, Paolo Gaspari, Boris Landoni, Fabio Riscica,
Mirco Segatello, Alessandro Sottocornola.
(Redazione@elettronica.in)

Grafica:

Alessia Sfulcini
(Alessia.Sfulcini@elettronica.in)

Ufficio Pubblicità:

Monica Premoli (0331-799775).
(Monica.Premoli@elettronica.in)

Ufficio Abbonamenti:

Elisa Guarnerio (0331-799775).
(Elisa.Guarnerio@elettronica.in)

DIREZIONE, REDAZIONE,

PUBBLICITÀ:

VISPA s.n.c. - via Adige 11 - 21013 Gallarate (VA)
Telefono 0331-799775 Fax 0331-778112

Abbonamenti:

Annuo 10 numeri Euro 36,00 Estero 10 numeri Euro 78,00
Le richieste di abbonamento vanno inviate a: VISPA s.n.c.,
via Adige 11, 21013 Gallarate (VA) tel. 0331-799775.

Distribuzione per l'Italia:

SO.D.I.P. Angelo Patuzzi S.p.A.
via Bettola 18 - 20092 Cinisello Balsamo (MI)
Telefono 02-660301 Fax 02-66030320

Stampa:

ROTO 3 srl - 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. Una copia Euro 4,50, arretrati Euro 9,00 (effettuare versamento sul CCP n. 34208207 intestato a VISPA snc).

Poste Italiane Spa - Spedizione in abbonamento Postale - D.L.
353/2003 (conv. in L. 27/02/2004) art.1 comma 1 - DCB Milano.
Impaginazione e fotolito sono realizzati in Desktop Publishing
con programmi Quark XPress 6.1 e Adobe Photoshop 8.0 per
Windows.

Tutti i contenuti della Rivista sono protetti da Copyright. Ne è vietata la riproduzione, anche parziale, la traduzione e più in generale la diffusione con qualsiasi mezzo senza l'autorizzazione scritta da parte dell'Editore. I circuiti, il firmware ed il software descritti sulla Rivista possono essere realizzati solo per uso personale, ne è proibito lo sfruttamento a carattere commerciale e industriale. Tutti possono collaborare con Elettronica In. L'invio di articoli, materiale redazionale, programmi, traduzioni, ecc. implica da parte del Collaboratore l'accettazione dei compensi e delle condizioni stabilite dall'Editore (www.elettronica.in/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. © 1995-2006 VISPA s.n.c.

48

DIFFUSIONE WIRELESS DI AUDIO DIGITALE

Sistema di diffusione sonora wireless a 2,4 GHz realizzato con due nuovi moduli Aurel. La trasmissione avviene in modalità digitale FSK garantendo un'ottima immunità ai disturbi ed una qualità audio sorprendente. Dispone di 8 canali selezionabili tramite pulsante.

57

CORSO DI PROGRAMMAZIONE SMART CARD

Proseguiamo il nostro viaggio nel mondo delle Smart Card concludendo la descrizione della struttura della smartcard ACOS2 con particolare riferimento alla gestione dell'account ed ai relativi meccanismi di sicurezza. Quinta puntata.

67

CONTROLLO BLUETOOTH CON TELEFONO CELLULARE

Utilizziamo un cellulare dotato di interfaccia Bluetooth per controllare gli ingressi e le uscite della demoboard presentata alcuni mesi fa. Il telefono cellulare deve essere in grado di supportare applicativi Java.

75

PIOVE E TIRA VENTO: METTI LA TENDA SOTTO CONTROLLO

Utilizziamo la logica di programmazione a stati, tipica delle FSM (macchine a stati finiti) per realizzare un modulo a microcontrollore in grado di azionare in maniera automatica tende e tapparelle motorizzate. Possibilità di collegare più moduli in cascata.

87

ALLA SCOPERTA DEL CAN-BUS

Nato come protocollo di comunicazione seriale per fare colloquiare tra loro tutti i sistemi elettronici presenti a bordo delle autovetture, si sta affermando anche nell'automazione industriale e nella domotica. In questa e nella puntata successiva presentiamo un sistema in grado di registrare in real-time i messaggi scambiati tra due nodi. Decima puntata.



Mensile associato
all'USPI, Unione Stampa
Periodica Italiana

Iscrizione al Registro Nazionale della Stampa
n. 5136 Vol. 52 Foglio 281 del 7-5-1996
e al ROC n. 3754 del 27/11/2001

16



25



35



48



57



67



75



87



Editoriale

Non solo virus.

Ai tradizionali problemi di virus, spyware e trojan (usati dagli hacker per spiare gli utenti, rubare informazioni, scaricare codici maligni ed ottenere accessi non autorizzati ai computer), negli ultimi mesi si è ulteriormente aggravato il problema dello spam, che ormai ha raggiunto livelli insostenibili. Attualmente 9 messaggi su dieci sono indesiderate, mai richiesti e, spesso, portatori di minacce nascoste. Ormai non è più possibile pubblicare sul proprio sito un indirizzo di posta senza che questo ("letto" da sistemi automatici), venga sommerso da decine e decine di email indesiderate ogni giorno. Tutto ciò rappresenta un fastidio per gli utenti privati ma un costo sempre più elevato per le aziende. Si stima che, solamente in Europa, il tempo perso dal personale per leggere e cancellare le email indesiderate equivalga ad un costo di 3 milioni di Euro. Ma la cosa più grave è sicuramente rappresentata dalla perdita di fiducia da parte dei consumatori che potrebbe minare la futura espansione dei servizi e del commercio basato su Internet.

Visti i risultati finora conseguiti, le azioni intraprese da parte delle Autorità preposte hanno sortito scarsi effetti e gli impegni presi per il futuro, anche in sede di Commissione Europea, rischiano di rimanere carta straccia. A livello europeo solo due nazioni hanno attuato le linee guida preparate dalla commissione, riuscendo peraltro a ridurre drasticamente il flusso di comunicazioni indesiderate a fini commerciali o illegali. Probabilmente accanto ad un'opera di sensibilizzazione delle Amministrazioni dei vari Stati, la risposta definitiva ai problemi di spam potrebbe venire dai grossi player Internet che, alla fine, sono quelli che subiscono i maggiori danni non riuscendo a sviluppare compiutamente il proprio business. Ci riferiamo, ovviamente ai giganti del settore, da Google a eBay, da Amazon a Yahoo!

Perché, dunque, accanto agli Enti governativi, non coinvolgere queste società in quest'opera di bonifica? Siamo certi che con i mezzi tecnologici che questi giganti potrebbero mettere in campo, in poco tempo lo spam diventerebbe un brutto ricordo del passato.

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

[elencoInserzionisti]

Compendio Fiere	Fiera di Vicenza
E.R.F.	GR Elettronica
Expo Elettronica - Blu Nautilus	Mostra Regionale Elettronica Scandiano
Fiera di Genova	RM Elettronica
Fiera di Pescara	Wireless
Fiera di Pordenone	www.mdsrl.it
Futura Elettronica	

La tiratura di questo numero è stata di 22.000 copie.

P
E
S
C
A
R
A

2
0
0
6



ASSOCIAZIONE RADIOAMATORI ITALIANI

Sezione di PESCARA

Via delle Fornaci, 2

Tel 085 4714835 Fax 085 4711930

<http://www.aripescara.org>

e-mail: aripescara@aripescara.org



PROTEZIONE
CIVILE



DXCC
DESK



41^a FIERA MERCATO NAZIONALE DEL RADIOAMATORE DI PESCARA

25 - 26 NOVEMBRE 2006

SILVI MARINA (TE) - FIERA ADRIATICA - S.S.16 (Nazionale Adriatica) - Km. 432

SABATO 9:15 - 19:00 DOMENICA 9:00 - 19:00

AMPIO PARCHEGGIO GRATUITO - RISTORANTE - SELF SERVICE INTERNO

WARNER 2006
Esercitazione di Protezione Civile



con il patrocinio di



Provincia
di Pescara



Comune
di Silvi





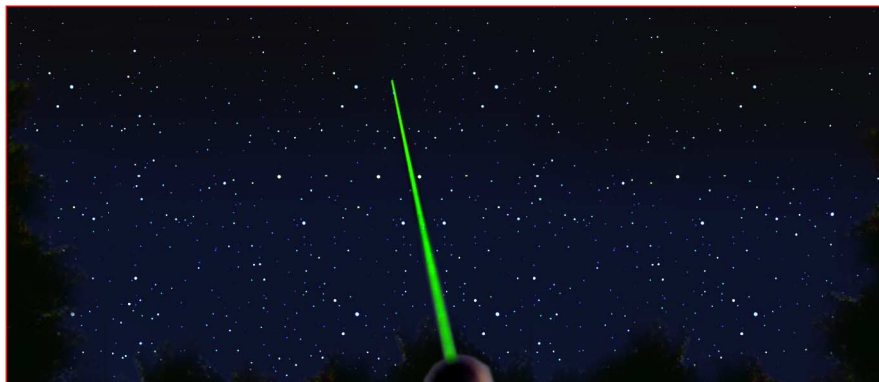
La tecnologia dei laser a luce verde

Ho di recente acquistato un piccolo laser a luce verde che punto verso il cielo per indicare stelle e costellazioni durante le serate con gli amici astrofili. Nonostante la limitata potenza (inferiore a 5 mW) il fascio si vede a chilometri e chilometri di distanza. Come è possibile tutto ciò? Come funzionano questi dispositivi?

Mario Matteoni - Firenze

In effetti chi osserva per la prima volta un puntatore laser a luce verde resta a bocca aperta. L'effetto è incredibile, non solo di notte ma anche di giorno. Ciò dipende dal fatto che la sensibilità relativa dell'occhio umano è massima alla lunghezza d'onda di 555nm, molto vicina a quella di emissione dei puntatori laser a luce verde (532nm). Di giorno il fascio emesso (e la relativa proiezione) viene mascherata dalla luce ambiente per cui, oltre i 200-300 metri, non possiamo più distinguere la luce emessa; di notte, invece, la portata del fascio può raggiungere anche i 10 chilometri il che risulta ideale per dimostrazioni all'aperto riguardanti l'astronomia e la volta celeste. Per questa peculiarità i laser a luce verde sono indispensabili anche come segnalatori di emergenza per quanti vanno per mare o si avventurano in montagna o attraverso i boschi. La diffusione di questi dispositivi è stata frenata dall'e-

vato costo dovuto alla complessità di realizzazione: non esistono infatti laser allo stato solido che emettono direttamente su questa lunghezza d'onda. Solo da poco tempo, grazie alla produzione di massa, i prezzi sono diventati abbordabili. Ad esempio, il puntatore laser verde MP1900G della Futura Elettronica (www.futurashop.it) costa attualmente soltanto 58 Euro. I puntatori oggi in commercio utilizzano la tecnologia DPSS (Diode Pumped Solid State Frequency Doubling technology) che consiste nell'impiego di un diodo laser all'infrarosso da 808nm la cui emissione viene focalizzata su un cristallo tipo NdYVO4 (Neodymium Doped Yttrium Orthovanadate) il quale eleva la lunghezza d'onda fino a 1064nm. Successivamente il fascio viene trattato da un altro cristallo (tipo KTP, Potassium Titanyl Phosphate, KTiOPO4) il quale, avendo una frequenza di risonanza doppia, genera l'emissione verde a 532nm. Un ulteriore filtro elimina la componente residua IR ed effettua una collimazione ottimale del fascio. E' evidente che questo processo risulta molto più complesso rispetto al semplice impiego di un diodo laser a luce rossa con lente in plastica che, ormai, possiamo trovare in commercio a pochi euro. Ricordiamo infine che anche i puntatori laser a luce verde sono pericolosi per l'occhio: non fateli maneggiare ai bambini ed evitate di puntare il fascio direttamente negli occhi di chicchessia, uomini o animali.



Servizio consulenza tecnica

Per ulteriori informazioni sui progetti pubblicati e per qualsiasi problema tecnico relativo agli stessi è disponibile il nostro servizio di consulenza tecnica che risponde allo 0331-245587.

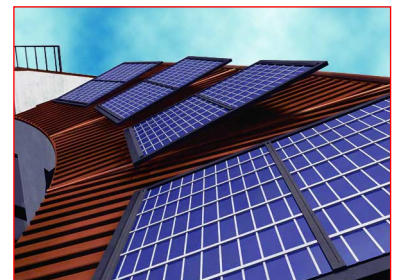
Il servizio è attivo esclusivamente il lunedì e il mercoledì dalle 14.00 alle 17.30.

Metti il sole sotto controllo

Ho collegato un pannello solare direttamente ad una batteria auto: un amico mi ha detto che in questo modo la batteria si può danneggiare, perchè viene ricaricata anche quando non ne ha bisogno...

Fabrizio Parini- Monza

In generale è sempre meglio interporre tra il pannello solare e la batteria un cosiddetto regolatore di carica che verifichi lo stato della batteria che la tensione fornita dal pannello solare. Se la batteria è scarica ed il pannello solare sufficientemente illuminato, il circuito viene chiuso consentendo alla batteria di ricaricarsi. In caso contrario (batteria già carica o pannello scarsamente illuminato), il circuito viene aperto. Naturalmente le caratteristiche del regolatore debbono essere scelte opportunamente; in particolare è necessario verificare la tensione nominale di funzionamento (solitamente 12 o 24 volt) e la massima corrente di ricarica.

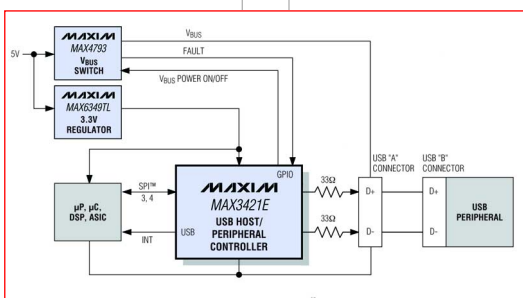


Variatore di velocità per motori in CA

Ho la necessità di aggiungere un'interfaccia USB alla sezione elettronica di controllo di una macchina utensile. Il circuito utilizza un microcontrollore della ST che non supporta questo tipo di interfaccia; non posso sostituire il micro ma posso modificarne il firmware. Cosa mi consigliate di fare?

Andrea Caruso - Avellino

Nell'ultimo anno molti produttori di semiconduttori hanno reso disponibili integrati specifici per questo scopo, ovvero semplici chip che svolgono la funzione di USB Host Controller e che consentono di aggiungere a qualsiasi scheda a microcontrollore, uC, DSP o ASIC questo tipo di interfaccia. Basta modificare poche righe di programma del micro ed il gioco è fatto. Tra i chip più interessanti da questo punto di vista



segnaliamo il MAX3421E della Maxim di cui riportiamo lo schema funzionale. Questo integrato è compatibile con lo standard USB 2.0, dispone di una SPI a 26 MHz per colloquiare col micro mentre per la ricezione e l'invio dei dati ha a disposizione due FIFO da 64 byte. Questo integrato dispone inoltre di 8 ingressi e 8 uscite supplementari di tipo General Purpose.

L'impiego di questo controller consente di aggiungere la funzionalità USB in maniera semplice e veloce a qualsiasi apparecchiatura industriale, strumento di misura e più in generale a qualsiasi circuito a microcontrollore. Per questo dispositivo la Maxim mette anche a disposizione una scheda di valutazione denominata MAX3421EVKIT mentre nel sito potrai trovare una serie di esempi di utilizzo con i riferimenti al codice da implementare nel tuo micro che dovrebbero renderti semplice e veloce il lavoro.

Caccia aperta all'acronimo misterioso

Pur essendo abituato alle terminologie ed alle sigle del mondo dell'elettronica ogni tanto mi trovo spiazzato, i miei allievi mi fanno una domanda e talvolta debbo, con modestia ed onestà, dire loro che farò una ricerca e poi saprò dare una spiegazione.

Matteo Rosi - Napoli

Quotidianamente ci troviamo di fronte nuove sigle, nuovi acronimi, fare confusione è facilissimo, in particolar modo quando siamo abituati a sentirle pronunciare in italiano ed all'improvviso qualcuno inizia a citarle in lingua inglese. Possiamo ben capire quanto sia importante per un insegnante offrire una traccia sintetica, ma efficace perché l'attenzione degli allievi si focalizzi correttamente; noi suggeriamo di visitare la seguente pagina: <http://www.st.com/stonline/it/itgloss.pdf> contenuta nel sito dell ST Miroelectronics dove un file in formato pdf è pronto all'uso e che speriamo sia costantemente aggiornato.



Telecamera di sorveglianza con illuminatore all'infrarosso

Per il controllo di un lungo passaggio anche durante le ore notturne avrei intenzione di installare una telecamera con illuminatori ad infrarosso; osservando le caratteristiche di vari prodotti non ho ben capito sino a che distanza possano lavorare in modo efficace i led all'infrarosso delle unità di video sorveglianza.

Giuseppe Marelli - Padova

L'estensione della distanza di ripresa notturna ottenibile con diodi led all'infrarosso montati in abbinamento a telecamere di sorveglianza può spaziare da 10 a 30 metri. L'efficacia dipende dal numero dei diodi che fanno parte della struttura. Desiderando ampliare il campo di

possono aggiungere dei pannelli illuminatori a led da posizionare in modo opportuno. L'orientamento dei led influisce significativamente sulla resa finale, tant'è che, per ottenere una resa ottimale, i led vengono abitualmente montati in posizione anulare attorno all'obiettivo della telecamera.

Questa soluzione garantisce il perfetto allineamento del fascio luminoso con l'area di ripresa della telecamera.

Se la telecamera di sorveglianza deve essere collocata in una postazione fissa e non si prevede un controllo motorizzato, è più facile trovare la posizione adeguata per eventuali pannelli illuminatori supplementari osservando sul monitor se l'area di lavoro è illuminata correttamente.



La classe dell'amplificatore

Come potete capire dalla domanda sono un principiante: poca esperienza e tanta curiosità. Potreste spiegarmi brevemente la differenza tra amplificatore audio di potenza che lavora in classe A e B.

Franco Boselli - Brescia

Nell'amplificatore classe A il transistor amplifica il segnale per l'intero periodo della forma d'onda, nell'amplificatore in classe B l'amplificazione avviene solo nel semiperiodo positivo o in quello negativo; per amplificare l'intera forma d'onda servono due transistor in Push-Pull: uno conduce nel semiperiodo positivo, l'altro in quello negativo.

www.radioamatore2.it

8ª Fiera del radioamatore, elettronica, home-computer

Radio amatore 2

18-19 novembre 2006

Sabato 09.00-19.00

Domenica 09.00-18.00

200 Aziende,
15.000 mq di esposizione:
e abbiamo detto tutto.

La migliore offerta di
Elettronica e informatica
low cost del Triveneto
Compra qui i tuoi regali di Natale.

Ti aspettiamo!

FRIULCASSA
CASSA DI RISPARMIO REGIONALE
SANDRIOLO



Pordenone Fiere

Fiera dell'Euroregione

PERCHE' ABBONARSI...

Elettronica In propone mensilmente progetti tecnologicamente molto avanzati, sia dal punto di vista hardware che software, cercando di illustrare nella forma più chiara e comprensibile le modalità di funzionamento, le particolarità costruttive e le problematiche software dei circuiti presentati. Se lavorate in questo settore, se state studiando elettronica o informatica, se siete insegnanti oppure semplici appassionati, non potete perdere neppure un fascicolo della nostra rivista! Citiamo, ad esempio, alcuni degli argomenti di cui ci siamo occupati nel corso del 2006:

ECCO ALCUNI VANTAGGI...

L'abbonamento annuo (di 10 numeri) costa € 36,00 anziché € 45,00 con uno sconto 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 risultasse 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 completa disposizione per fornirti tutte le informazioni necessarie riguardanti i progetti pubblicati.

... E INOLTRE AVRAI IN REGALO:

- La Discount Card che ti permette di usufruire di uno sconto del 10% su tutti i prodotti acquistati direttamente presso la ditta FUTURA ELETTRONICA.



- un volume a scelta della collana "L'ELETTRONICA PER TUTTI" (€15,00 cad.).
- Gli abbonati (e solo loro!) potranno scaricare gratuitamente dal sito www.elettronicain.it i file sorgente dei programmi e dei firmware utilizzati in molti dei progetti pubblicati.

TELECONTROLLO GSM CON ANTENNA INTEGRATA

Consente di controllare a distanza, sfruttando la rete cellulare, due carichi di potenza in modalità bistabile o monostabile. Dispone inoltre di due ingressi per l'invio di messaggi di allarme o di stato. Funziona anche da ricevitore per apricancello: basta chiamarlo da uno dei 200 numeri cui può essere abbinato, e con lo scambio di uno dei relé chiude il contatto della centralina.

DEMOBOARD PER BLUETOOTH

Per muovere i primi passi nell'affascinante mondo del Bluetooth, il protocollo che rende possibile la comunicazione tra categorie eterogenee di dispositivi elettronici. Realizziamo una demoboard con la quale prendere confidenza con questa tecnologia e sperimentare il controllo a distanza e la comunicazione vocale mediante un modulo in Classe 1 prodotto dalla Ezurio.

MINI COMBINATORE TELEFONICO GSM CON AUDIO

Economico e ultracompatto combinatore GSM da abbinare a qualsiasi impianto antifurto per casa. Dispone di due canali con messaggi vocali con 8 numeri per canale. Possibilità di invio chiamate vocali o messaggi SMS. Completo di contenitore plastico e antenna integrata su circuito stampato.



CAMPAGNA

ABBONAMENTI 2006

ABBONAMENTO ANNUALE

SOLO € 36,00 + OMAGGIO

COME FARE AD ABBONARSI?

- **On-line tramite Internet**

compilando il modulo riportato nella pagina "Abbonamenti" disponibile sul nostro sito Internet "www.elettronica.in.it".
Se possedete una **carta di credito** potrete 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 piè di pagina.**

Riceverai direttamente a casa tua un bollettino personalizzato di C/C postale.
L'abbonamento decorrerà dal primo numero raggiungibile.
Per il rinnovo attendere il nostro avviso.

L'e-mail è il modo più semplice e veloce per stabilire un contatto con noi. Se possedete una casella di posta elettronica non dimenticate di inserirla nel modulo di richiesta.

Abbonamenti per l'estero

I lettori residenti all'estero potranno richiedere l'abbonamento alla rivista in formato digitale ad un prezzo vantaggioso. Ogni mese sarà disponibile per il download il fascicolo in formato digitale ad alta risoluzione. L'abbonamento estero digitale può essere effettuato solamente on-line con pagamento con carta di credito.

MODULO D'ABBONAMENTO

Sì desidero abbonarmi per un anno alla rivista **Elettronica In.**
Resto in attesa del primo numero e degli omaggi:



Discount Card *Futura Elettronica*;

scegli uno tra questi
volumi della collana
"L'Elettronica per tutti"



**Programmiamo
con i PIC**



**100+1
circuiti elettronici**



**Alla scoperta
della CCTV**

Nome _____ Cognome _____

Via _____ N. _____ Tel. _____

CAP _____ Città _____ Prov. _____

e-mail _____

Data _____ Firma _____

Resto in attesa di vostre disposizioni per il pagamento.

Formula di consenso: il sottoscritto, acquisite le informazioni di cui agli articoli 10 e 11 della legge 675/96, conferisce il proprio consenso alla Vispa s.n.c affinché quest'ultima utilizzi i dati indicati per svolgere azioni correlate all'inoltro dei fascicoli e di materiale promozionale e di comunicarli alle società necessarie all'esecuzione delle sopracitate azioni. È in ogni caso facoltà dell'interessato richiedere la cancellazione dei dati ai sensi della legge 675/96 articolo 163. Spedire in busta chiusa a o mediante fax a:

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



26^o MARC

**mostramercato attrezzature
radioamatoriali & componentistica
hardware • software
ricezione satellitare
editoria specializzata
radio d'epoca**

**Fiera di Genova
16 - 17 Dicembre 2006**

**sabato ore 9 • 18,30
domenica ore 9 • 18**

ENTE ORGANIZZATORE E SEGRETERIA:

STUDIO FULCRO s.a.s.

Piazzale Kennedy, 1 - 16129 Genova

Tel. 010.561111 - Fax 010.590889

www.studio-fulcro.it e-mail: info@studio-fulcro.it

ENTE PATROCINATORE:

A.R.I. - Ass. Radioamatori Italiani

Sezione di Genova

Salita Carbonara 65 b - 16125 Genova

C. P. 1117 - 16121 Genova - Tel./Fax 010.25.51.58

www.arigenova.it

SOBRIO MOBILE IN DISCOTECA

Arriva il cellulare... etilometro. La nuova "funzione" è resa possibile da un programma brevettato a livello mondiale. Il progetto è stato presentato al XVII convegno dell'Associazione Italiana Imprese di Intrattenimenti Danzanti e di Spettacolo (Silb), a Santa Margherita Ligure. La logica di funzionamento è simile a quella delle normali ricariche per schede telefoniche. Il prodotto si chiama "Sobrio mobile" e consiste in una tessera sulla quale è inserito un codice segreto da digitare sul proprio cellu-

lare per scaricare il programma da installare successivamente. Il software propone un questionario in cui vengono richiesti parametri fisici, quanto e che cosa si è bevuto, poi... arriva la risposta: perché con un semplice calcolo, è possibile verificare il tasso alcolemico delle persone. Silb distribuirà gratuitamente migliaia di tessere ai frequentatori di locali notturni.

Maggiori informazioni sono disponibili al seguente indirizzo: <http://www.phoneteam.it/Public/Brochure-SobrioMobile.pdf>

**Non ti "bere" la
PATENTE!
è arrivato**



GRATIS, SOFTWARE PRONTO ALL'USO

Wininizio.it presenta nel proprio sito Wininizio PenSuite, una collezione di software gratuito e open source da copiare su una Pen Drive USB o per uso su Hard Disk. La nuova versione della PenSuite viene presentata nelle edizioni classiche LITE, PREMIUM e GOLD, contenenti software appartenente a tutte le categorie. Il software preparato per le PenSuite è di tipo standard e non installa, non richiede installazione: infatti, tutte le applicazioni possono essere lanciate direttamente dalla Pen Drive e vengono gestite da un pratico menu principale simile all'avvio veloce di Windows. Basta inserire la Pen Drive in una porta USB del computer e immediatamente si potrà disporre di centinaia di software di tutti i generi, da poter utilizzare a seconda delle esigenze.



Sarà quindi possibile avere nel taschino il software necessario in ogni situazione. La PenSuite offre software di qualsiasi categoria, tanto che si potrebbe usare un PC con il solo sistema operativo installato, mentre tutti i programmi potrebbero essere disponibili solo inserendo la Pen Drive in una porta USB. Il materiale delle PenSuite può essere utilizzato benissimo anche su Hard Disk!

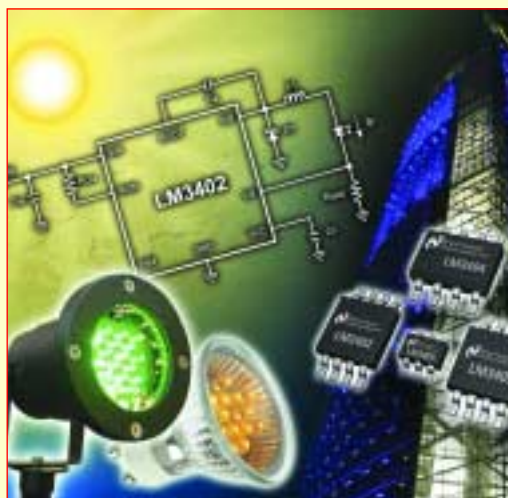
Anche in questo caso non occorre installare i programmi: basterà copiarne il contenuto in una cartella e avviare l'applicazione che occorre, scegliendo fra le centinaia di software suddivisi nelle varie categorie. Per scaricare il software basta accedere al seguente indirizzo: <http://pensuite.wininizio.it/ita/index.html>

DRIVER PER LED DI ALTA POTENZA ED ELEVATA LUMINOSITÀ

National Semiconductor Corporation presenta una nuova famiglia di driver per il pilotaggio di LED progettati per controllare le nuove generazioni di LED da 1W e 5W di elevata luminosità, utilizzati nei settori auto e industria oltre che in applicazioni di illuminazione generalizzate. I LED driver LM3402, LM3404 e LM3405 offrono la più ampia gamma di tensione di ingresso richiesta da questo segmento di mercato attualmente in rapida crescita. Oltre a questa caratteristica i driver per LED di National forniscono una corrente costante per la regolazione della luminosità del LED ed utilizzano una tensione di feedback che permette di minimizzare la dissipazione di potenza. National offre due versioni del driver per LED LM3402 ottimizzato per il controllo di dispositivi LED da 1W, e caratterizzate da una gamma di tensione di ingresso che va da 6V a 42V o da 6V a 75V. L'LM3404 e l'LM3405 sono invece adatti a controllare LED da 3W e 5W grazie alla loro maggiore capacità di pilotaggio. L'LM3402 è un regolatore buck compatto e caratterizzato da una efficienza che arriva al 95%, in grado di erogare una corrente di

525mA. Offerto in un contenitore mini SOIC da 8 pin, l'LM3402 è dotato di un ingresso di abilitazione che può essere utilizzato per la regolazione della luminosità attraverso un segnale PWM (pulse width modulation) e da una tensione di ingresso per la regolazione di corrente di soli 200mV. La tensione di ingresso dell'LM3402 va da 6V a 42V. Per tensioni di alimentazione superiori è disponibile l'LM3402HV che presenta le

stesse caratteristiche, ma opera con una tensione di alimentazione compresa tra 6V e 75V. Disponibili in contenitore SOIC narrow da 8 pin, l'LM3404 e l'LM3404HV presentano la stessa funzionalità dell'LM3402 e dell'LM3402HV, ma hanno una capacità di pilotaggio di 1,2A. Questi prodotti sono adatti per alimentare i LED da 3W e 5W con livelli di corrente rispettivamente di 700mA e 1A.



Realizzato in un contenitore mini SOIC da 8 pin e SOT-23 a 6 pin, il regolatore buck LM3405 eroga una corrente massima di 1A utile a pilotare i LED da 3W e 5W. Con tensione di feedback di soli 200mV per il controllo della corrente e una frequenza operativa di 550kHz o 1,6MHz fissata internamente, l'LM3405 fornisce una semplice ed efficiente soluzione per il pilotaggio di LED. L'LM3405 è caratterizzato da una tensione di ingresso compresa tra 3V e 22V, è compensato internamente ed è dotato di un ingresso per la regolazione della luminosità via PWM. Ulteriori informazioni su questi componenti sono disponibili all'indirizzo: <http://www.national.com/pf/LM>

NOKIA 8800 SIROCCO EDITION

E' composta da Brian Eno, pioniere della "musica ambient", la colonna sonora per il nuovo Nokia 8800 Sirocco Edition: cellulare ispirato alle forze della natura e alla bellezza scultorea del corpo.

Sensuale per natura ed intuitivamente bello nel design, la sua caratteristica chiave è l'impronta per il pollice sul meccanismo scorrevole. L'operatività del nuovo Nokia è tri-band GSM in cinque continenti (reti GSM 900/1800/1900) con un display a matrice attiva (TFT) a colori ad alta risoluzione da 262.144 pixel (208 x 208).

Nell'apparecchio è integrata una fotocamera da 2 megapixel e l'Image engine3D consente risultati di grafica avanzata. Il lettore musicale è compatibile con i formati MP3 ed AAC, il ricevitore radio stereo opera sulla banda FM.

Per quanto riguarda la messaggisti-

ca, il mobile phone gestisce MMS e prevede l'utilizzo di 47 differenti lingue. La posta elettronica supporta i protocolli SMTP, POP3, IMAP4 e APOP. Per l'istant messaging troviamo la funzione "presence" che verifica lo stato del telefono cellulare che si sta per chiamare prima di comporre il numero; purtroppo è un ser-



vizio che al momento non possiamo testare poichè nessun operatore lo ha reso disponibile.

La memoria interna è di 128 MByte e il sistema operativo permette di utilizzare applicazioni Java.

Per la connettività riscontriamo che il sistema Bluetooth corrisponde alla versione 2.0 con EDR (Enhanced Data Rate); la sincronizzazione remota over-the-air (OTA) è possibile fra dispositivi compatibili con SyncML, mentre la sincronizzazione locale verso un PC si effettua con Nokia PC Suite.

Il browser per la navigazione internet si connette con protocollo TCP/IP e supporta XHTML; il trasferimento dati avviene, oltre che in TCP/IP, con modalità EDGE/E-GPRS e GPRS; in entrambi i casi in classe 8. Ultima nota, il prezzo: inferiore ai 1000 Euro. Per ulteriori informazioni: www.nokia.com

WEB AD OCCHI CHIUSI



Anche chi ha grossi problemi di vista potrà navigare su internet e avere qualsiasi tipo di informazione, multimediale e ipertestuale presente in rete grazie ad un dispositivo inventato e brevettato dal Politecnico di Milano e sviluppato congiuntamente all'Istituto dei Ciechi del capoluogo lombardo. Il web "ad occhi chiusi" è un brevetto del Politecnico per disabili visivi che, armati di penna e tavoletta grafica, potranno così viaggiare navigando per il mondo digitale. L'invenzione (spiegano gli esperti del Politecnico di Milano) consiste in un apparato che, durante l'esplorazione mediante touch screen di una schermata web, utilizza i due canali stereo dell'audio per offrire una combinazione di stimoli audio e tattili. L'immagine web di un sito per cui è stata progettata una corretta accessibilità è predisposta in modo da poter generare suoni; in questo caso, attraverso il canale di sinistra si avrà disponibilità di un segnale audio, mentre tramite quello di destra viene generato un impulso tattile percepibile mediante un trasduttore di contatto da indossare sul dito. Attualmente i non vedenti possono avvalersi di software che leggono il contenuto della pagina web a condizione che sia stata correttamente progettata (ossia non completamente realizzata con filmati) ed ascoltare il testo esposto da una voce sintetizzata. Con il dispositivo messo a punto dal Politecnico di Milano si aggiunge la sensazione tattile e quindi la possibilità di leggere con il tatto. Per informazioni: <http://www.istciechimilano.it>, sito dove gli sviluppatori web potranno trovare anche altre interessanti informazioni riguardanti l'accessibilità dei siti agli ipovedenti.

RELÈ STATICO IP20

Alloggiato in un moderno involucro IP20 in formato standard «hockey puck», il relè statico P19176 della Celduc Relais permette di fare variare la tensione applicata al carico proporzionalmente al valore della grandezza analogica applicata all'ingresso. Questa nuova gamma di relè permette, col suo ridotto ingombro, correnti di uscita fino a 125A. I prodotti sono attualmente disponibili con ingresso 4-20 mA ma presto la gamma sarà arricchita da modelli con ingresso 0-10V o con potenziometro. L'uscita è protetta da un varistore, con livello di isolamento di 4KV in conformità alle norme EN60950/VDE0805. Info: <http://www.celduc-relais.com>



X-BOT IL PICCOLO UMANOIDE



Presso il laboratorio didattico di robotica del "Museo delle Scienze e delle Tecnologie" Leonardo da Vinci di Milano è stato presentato X-bot. Si tratta di un robot alto poco meno di 50 centimetri realizzato dalla NuZoo.

La sembianza, come potete vedere dall'immagine, è del tipo umanoide, gli evolutissimi abbinamenti fra elettronica e meccanica hanno consentito di ottenere movimenti degli arti particolarmente accurati e sistemi giroscopici controllati via software permettono al robot di compiere i primi passi. Sì, parliamo di primi passi, perchè i progettisti lo

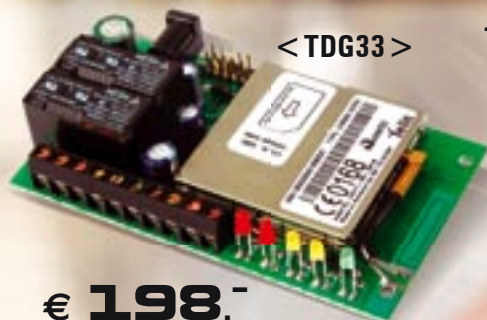
hanno paragonato ad un bimbo di un anno che ogni giorno apprende tramite nuove esperienze. Per ora X-bot necessita ancora di un cavo che lo collega ad un personal computer, ma per le serie successive (X-bot sembra che rimarrà figlio unico) l'autonomia dovrebbe essere decisamente maggiore. Chi desiderasse vedere altre foto e visualizzare filmati del robot in funzione può collegarsi al sito: www.nuzoo.it.



DISPOSITIVI GSM PER CONTROLLI REMOTI

Per controllare, attivare, verificare, ascoltare in modalità remota sfruttando le reti GSM che coprono capillarmente tutto il territorio nazionale. Tutti i dispositivi vengono forniti montati e collaudati.

Tutti i prezzi sono da intendersi IVA inclusa.



< TDG33 >

TELECONTROLLO GSM BIDIREZIONALE CON ANTENNA INTEGRATA

Sistema di controllo remoto bidirezionale che sfrutta la rete GSM per le attivazioni ed i controlli. Configurabile con una semplice telefonata, dispone di due uscite a relè (230Vac/10A) con funzionamento monostabile o bistabile e di due ingressi di allarme optoisolati. Possibilità di memorizzare 8 numeri per l'invio degli allarmi e 200 numeri per la funzionalità apricancello. Tutte le impostazioni avvengono tramite SMS.

Caratteristiche tecniche:

Alimentazione: 5÷32 Vdc - Assorbimento massimo: 500mA - Antenna GSM bibanda integrata - GSM: Dual Band EGSM 900/1800 MHz (compatibile con ETSI GSM Phase 2+Standard) - Dimensioni: 98 x 60 x 24 (L x W x H) mm. Il prodotto viene fornito già montato e collaudato.

€ 198,-

TELECONTROLLO GSM BIDIREZIONALE

Unità di controllo GSM con due ingressi fotoaccoppiati e due uscite a relè. Utilizzabile sia per attivare a distanza qualsiasi apparecchiatura che per ricevere messaggi di allarme. In modalità apricancello è in grado di memorizzare fino ad un massimo di 100 utenti. Ideale per realizzare impianti antifurto per abitazioni e attività commerciali, car alarm, controlli di riscaldamento/condizionamento, attivazioni di pompe e sistemi di irrigazione, apertura cancelli, controllo varchi, circuiti di reset, ecc. Fornito montato e collaudato.

Caratteristiche tecniche:

Frequenza di lavoro: GSM bibanda 900/1.800MHz • Funzione apricancello a costo zero • 2 ingressi optoisolati • 2 uscite a relè (bistabile o astabile) • 5 numeri abbinabili per allarme • 100 numeri abbinabili per apricancello • carico applicabile alle uscite: 230V, 5A • alimentazione: 5÷32V • assorbimento massimo: 550mA.

€ 182,-

< TDG34 >

APRICANCELLO GSM CON ANTENNA INTEGRATA

Dispositivo di controllo GSM da utilizzare in abbinamento al sistema di apertura dei cancelli elettrici.

Il funzionamento è molto semplice: il cancello può essere azionato effettuando una chiamata con il proprio cellulare al numero della SIM Card inserita nell'unità GSM. La chiamata non avrà mai risposta (in questo modo non si consuma neppure uno scatto) ma il dispositivo invierà un comando alla centralina di controllo del cancello che provvederà ad aprirlo o chiuderlo. Gestione degli utenti da remoto mediante SMS (è necessario conoscere la password) oppure in locale tramite PC con apposito software di configurazione. Alimentazione 12÷24 Vdc selezionabile mediante jumper. Fornito già montato e collaudato.

€ 165,-

< TDG37 >

MODEM GSM CON INTERFACCIA USB

Modem particolarmente compatto utilizzabile in tutte le applicazioni nelle quali si ha la necessità di effettuare trasmissioni dati sfruttando la rete mobile GSM. È dotato di porta USB che ne permette l'interfacciamento a qualsiasi PC o Notebook provvisto di tale periferica. L'alimentazione al dispositivo è fornita direttamente dalla connessione USB.

Caratteristiche tecniche:

Modulo bibanda GSM/GPRS Telit: frequenze 900/1800 MHz • potenza RF: 2W (900 MHz), 1W (1800 MHz) • alimentazione: 5V (tramite porta USB) • assorbimento a riposo: 30 mA • assorbimento in connessione: 250mA • interfaccia dati: USB1.1 e USB2.0 • antenna: bibanda, integrata su CS.

< STD32 >



€ 228,-

COMBINATORE TELEFONICO GSM CON AUDIO

Economico e ultra-compatto combinatore GSM da abbinare a qualsiasi impianto antifurto domestico.

Dispone di due canali con messaggi vocali con 8 numeri per canale. Possibilità di invio chiamate vocali o messaggi SMS. Completo di contenitore plastico e antenna integrata su circuito stampato.

€ 210,-

< TDG35 >

Caratteristiche tecniche:

Combinatore telefonico GSM a due canali • 2 messaggi vocali da 10 secondi • 5 cicli di chiamata per canale • 2 ripetizioni del messaggio • invio messaggio vocale o SMS • segnalazione di campo di presenza GSM • blocco allarme da remoto • programmazione dei numeri su SIM • riconoscimento chiamata a buon fine.



**FUTURA
ELETTRONICA**

Via Adige, 11 • 21013 GALLARATE (VA)

Tel. 0331/799775 •

Fax 0331/778112

www.futuranet.it

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

ASCOLTO AMBIENTALE GSM

Dispositivo GSM per ascolto ambientale di tipo professionale caratterizzato da dimensioni estremamente compatte (il contenitore misura appena 56 x 75 x 15mm) e da un'elevata qualità del segnale audio. Indispensabile per l'ascolto ambientale all'interno di veicoli, può essere utilizzato anche in abitazioni e uffici. L'apparecchiatura viene fornita già montata e collaudata completa di microfono, contenitore, antenna bibanda, cavi, manuale operativo e valigetta per il trasporto. Disponibile nella versione a 1 microfono (cod. FT607M1) e a 2 microfoni (cod. FT607M2).

Caratteristiche tecniche:

Frequenza di lavoro: GSM 900/1800 MHz • microfono: Knowles • programmazione e controlli: SMS o DTMF • tensione di alimentazione: 5÷32Vdc; assorbimento a riposo: 20 mA (max.) • assorbimento massimo: 300mA • sensore di movimento • dimensioni: 56 x 75 x 15 mm.

< FT607M1 > € 840,-
< FT607M2 > € 900,-





PARCO CITTA'
*Centro Commerciale
In Vicenza*

5-6-7 gennaio 2007

1° EXPO

*di elettronica, informatica,
telefonia mobile, radiantistica,
componentistica elettronica,
tecnologia solare, hobbistica*

PARCO CITTA'
Via Quadri
Vicenza

Orario 09.00 – 18.00

info line
380/3415800
www.VicenzaEventi.it
info@VicenzaEventi.it

Organizzazione: Vicenza Eventi
Ufficio in Galleria "Parco Città"



PARCO CITTA'

*Centro Commerciale
Vicenza*

Galleria commerciale 3600 mq 



Robot FOX

lo sguardo all'infrarosso

di Mirco Segatello



Mini automa in grado di muoversi liberamente e di evitare gli ostacoli grazie ai sensori di prossimità all'infrarosso di cui è dotato. Dispone anche di sensori che gli permettono di rilevare quando non è più presente una superficie di appoggio. Grazie ad un modulo radio può essere controllato anche tramite PC.

Robot. Quale altra parola ci fa pensare in modo così forte alla tecnologia e al futuro? Non c'è una definizione per la parola robot. In generale si può dire che un robot è una macchina programmabile che imita le azioni e/o l'aspetto di una creatura "intelligente". Per essere definita robot, la macchina deve raccogliere informazioni da quanto la circonda attraverso dei sensori e interagire in modo fisico con l'ambiente. La parola "robot" viene dalla parola ceca "robota" che significa lavoro forzato o ripetitivo. Lo scrittore ceco Karel Capek usò questo termine nel

1920 per la sua opera teatrale R.U.R. (Rossum Universal Robot), nella quale dei robot umanoidi prendevano il controllo del mondo. Sebbene a lungo immaginati, i primi robot sono stati costruiti solo negli anni '50 quando l'avvento di transistor e delle nascenti tecniche di miniaturizzazione permisero l'allestimento di macchine complesse e versatili.

Oggi l'obiettivo della ricerca non è quello di simulare in modo artificiale le caratteristiche dell'uomo, ma di realizzare macchine capaci di inserirsi nella realtà di tutti i giorni con comportamenti autonomi e intenzio-

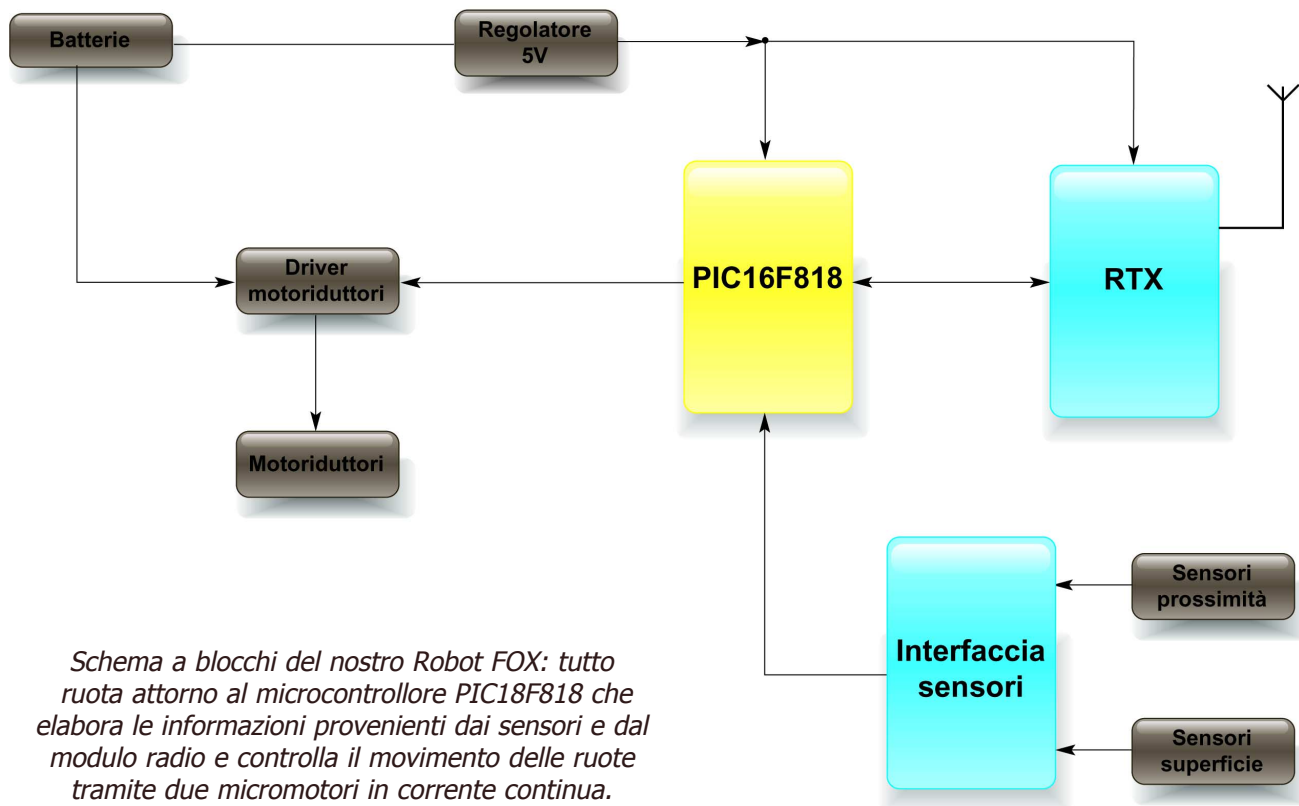


Fig. 1

Specifiche tecniche

- Alimentazione: 8 batterie formato stilo;
- Assorbimento massimo 100mA;
- Movimento con due micromotori in CC;
- Sensori rilevamento oggetti ad infrarosso;
- Sensori rilevamento bordo del tavolo a raggi infrarossi;
- Controllo remoto da PC tramite moduli radio Aurel;
- Gestione a microcontrollore (PIC16F818).

nali. La robotica utilizza idee e contributi provenienti da discipline apparentemente molto distanti quali la meccanica, l'elettronica, l'informatica, la biologia, la psicologia e le neuroscienze. Questo mese il professore Mirco Segatello, ci presenta un altro lavoro sviluppato in ambito scolastico: si tratta del "Robot Fox" costruito da Fabio Penzo, un allievo del quinto anno

dell'Istituto Tecnico Industriale Statale Leonardo da Vinci di Portogruaro. Fabio non si è occupato di neuroscienze, psicologia e biologia, ma ha saputo mettere insieme quanto ha appreso a scuola per quanto concerne elettronica, informatica e meccanica. Ma come possiamo definire l'apparecchio che vedete nelle immagini? Robot? Oppure è più corretto considerarlo

un giocattolo dotato di automatismi? Non abbiamo dubbi è un robot. Molto spesso giocattoli molto evoluti ci traggono in inganno, perché hanno mobilità considerevole, parlano e quando sono sul loro palcoscenico fanno una gran bella figura; ma sono solo giocattoli ripetono un programma che qualcuno ha configurato ed attivato con un radiocomando.

Robot Fox non ha il fascino accattivante di un umanoide come quelli della serie cinematografica "Star Wars" ma, nonostante ciò, si muove sulla superficie di un tavolo ed evita gli ostacoli posti lungo il suo percorso senza problemi. Questo è possibile perché, il "cervello" del circuito, un micro PIC16F818, è stato opportunamente programma- ➤

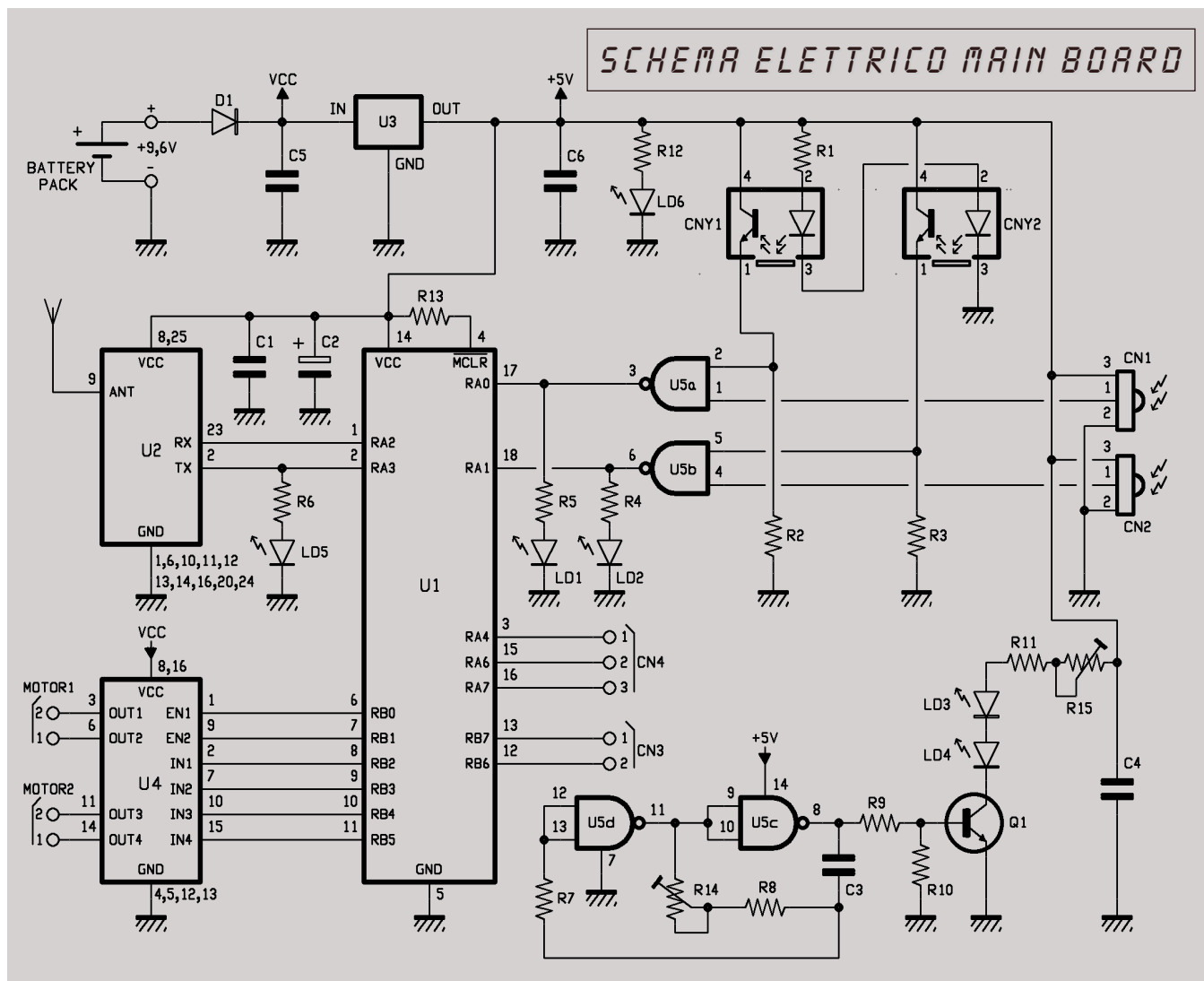


Tabella 1

Sensore prossimità	Sensore bordo tavolo	Segnale al PIC
1 (nessun oggetto)	1 (sul tavolo)	0
0 (presenza oggetto)	1 (sul tavolo)	1
1 (nessun oggetto)	0 (fuori tavolo)	1
0 (presenza oggetto)	0 (fuori tavolo)	1

to per poter gestire i sensori e pilotare, tramite un driver L293D, i motoriduttori che gli consentono di muoversi agilmente. Grazie ai sensori, che gli consentono di esplorare lo spazio circostante, cerca intelligentemente di non cacciarsi nei guai, richiamando al momento giusto le routine adatte.

Nel progetto che il professor Mirco Segatello ci ha presentato nel numero 111 di *Elettronica In*, il robot era in grado di evitare cadute dalla superficie di un tavolo grazie all'impiego di sensori CNY70 e di evitare ostacoli grazie a "baffi" collegati a microswitch. Robot Fox si è tolto i "baffi" ed al suo posto sono comparsi dei più tecnologici sensori all'infrarosso. Inoltre, sulla basetta trova spazio un modulo radio rice-trasmittente tramite il quale il robot può essere comandato a distanza con un personal computer. Per quanto concerne l'aspetto meccanico la novità sta in una terza ruota pivotante che gli consente anche di ruotare su se stesso senza difficoltà.

Analisi dello schema

Soffermiamoci sui diagrammi della pagina accanto per avere un'idea dei rapporti che esistono fra le diverse sezioni dello schema. Il circuito funziona prelevando tensione da un set di 8 batterie a stilo Ni-Cd oppure Ni-MH collegate in serie in modo da erogare circa 9,6V. La corrente richiesta dal circuito è di circa 100 mA, pertanto non è necessario utilizzare batterie di elevata capacità. Consigliamo di prevedere un interruttore tra il circuito e le batterie in modo che queste possano essere

scollegate elettricamente per spegnere il robot. Inoltre se prevedete un plug in parallelo al pacco batterie, potrete ricaricarle con un caricabatterie tipo quello presentato sul N.77 di *Elettronica In*.

Dalle batterie, la tensione di alimentazione arriva al diodo D1 che ha il compito di proteggere l'apparecchio da una accidentale inversione di polarità. Dopo il diodo la tensione giunge al driver L293D, il cui compito è quello di pilotare i moto-

Tabella 2

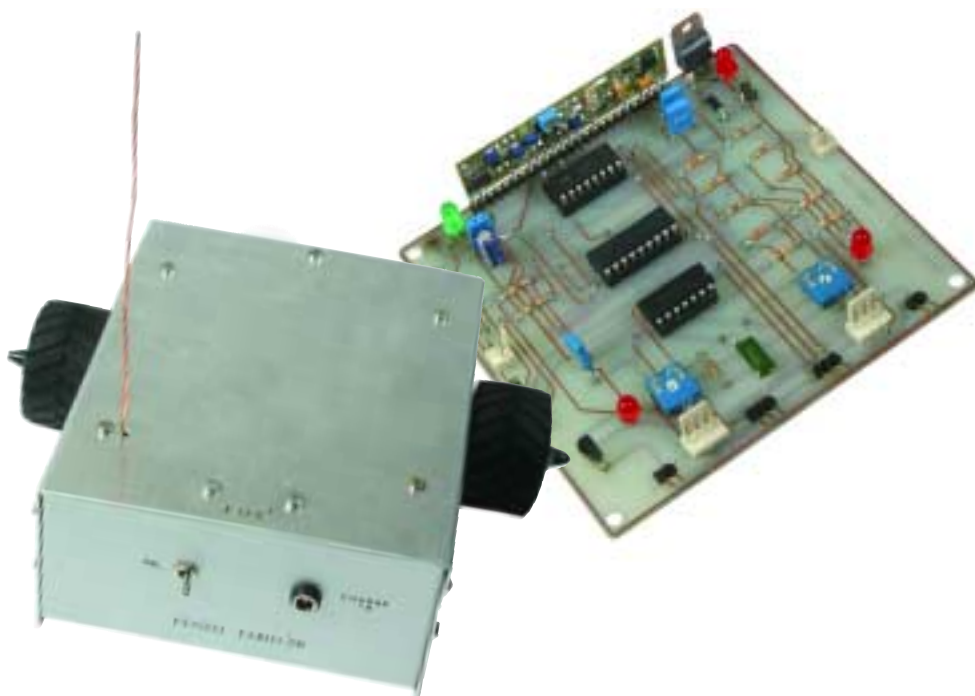
IN1 IN2	IN3 IN4	Movimento robot
0 0	0	Fermo
0 1	1	Avanti
1 0	0	Indietro
1 1	1	Fermo

riduttori, e all'ingresso del regolatore di tensione U3 (un 7805) che si occupa di ridurre e stabilizzare la tensione a 5V utilizzati per alimentare la logica nel circuito.

In particolare la tensione di 5V è portata al piedino 14 del PIC16F818, al ricetrasmittitore ➤

Tabella 3

Pin	Porta	Funzione
1	RA2	TX RTFSAW
2	RA3	RX RTFSAW
3	RA4	I/O opzionale
4	MCLR	Reset
5	GND	Negativo alimentazione
6	RB0	ENABLE1 L293D
7	RB1	ENABLE2 L293D
8	RB2	IN1 L293D
9	RB3	IN2 L293D
10	RB4	IN3 L293D
11	RB5	IN4 L293D
12	RB6	I/O opzionale
13	RB7	Illuminatore a led
14	VCC	Positivo alimentazione
15	RA6	I/O opzionale
16	RA7	I/O opzionale
17	RA0	Sensori DX
18	RA1	Sensori SX



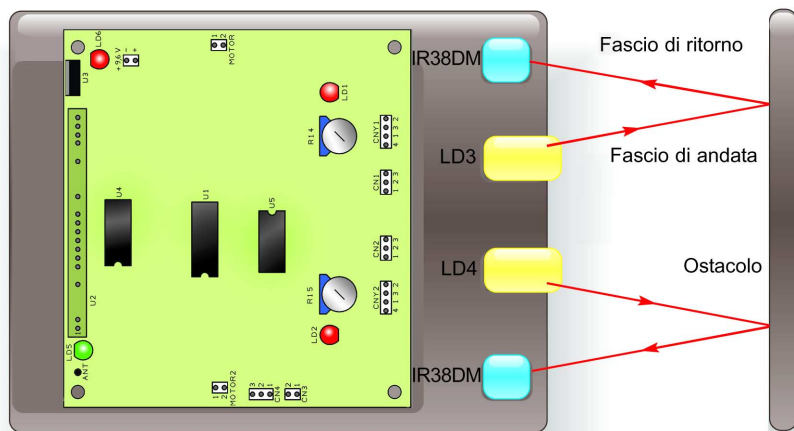


Fig. 2

I sensori di prossimità posti sulla parte anteriore della struttura rilevano gli ostacoli a circa 10 cm; essi funzionano a 38KHz: nello schema generale potete notare che U5d e U5c sono impiegati per generare proprio questa frequenza. La taratura dei sensori si effettua con facilità regolando il trimmer R14 per la frequenza ed R15 per la sensibilità. In basso a destra, un dettaglio dell'IR38DM, usato come ricevitore per infrarossi, con la relativa identificazione dei terminali. Nel disegno sottostante possiamo vedere come funziona il sensore CNY70, una coppia emettitore/ricevitore IR. Nel nostro caso questo dispositivo viene utilizzato per verificare la presenza della superficie di appoggio.

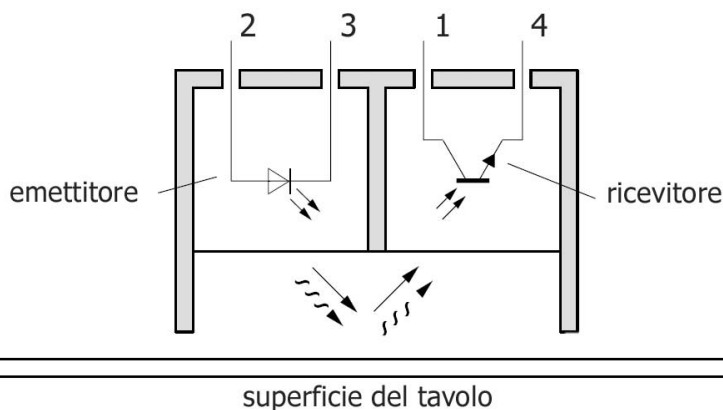


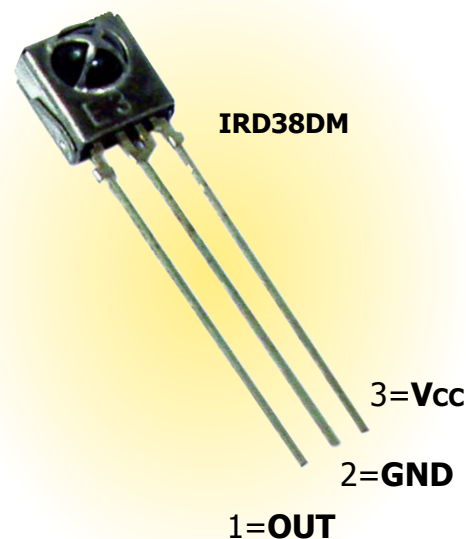
Fig. 3

RTFSAW in corrispondenza dei piedini 8 e 25 ed al pin 14 di U5, un 7400. Con i sensori CNY1 e CNY2 il robot, ovvero il microcontrollore, è in grado di rilevare la presenza del piano su cui è posizionato in modo da richiamare la routine per evitare che Robot Fox possa cadere, per esempio, dal tavolo sul quale è

posto. Dall'abbinamento ottico fra i led trasmettitori all'infrarosso LD3, LD4 ed i ricevitori IRD38DM CN1 e CN2, il microcontrollore è in grado di ricavare le informazioni necessarie per evitare gli ostacoli. I led all'infrarosso non sono collegati direttamente ai 5V, ma vengono pilotati tramite un multivibratore

astabile, la cui frequenza di oscillazione è di circa 38 kHz e che può essere aggiustata tramite il trimmer R14. Questa soluzione è necessaria in quanto il ricevitore IR38DM è in grado di ricevere un segnale ottico modulato alla frequenza di 38kHz restituendo al pin OUT uno stato logico basso o alto a seconda che venga rilevato o meno un segnale all'infrarosso. In questo modo quando il robot incontra un ostacolo, il raggio rimbalza sull'ostacolo e raggiunge il rilevatore IR38DM la cui uscita passa da 5V a 0V. I sensori non sono collegati direttamente al microcontrollore, ma a due porte NAND. Come possiamo osservare nella Tabella 1 quando un sensore rileva un problema (stato logico 0) l'uscita della porta NAND passa al livello logico alto consentendo al microcontrollore di prendere le adeguate contromisure. In figura 4 mostriamo lo schema a blocchi relativo alla routine di rilevazione ostacoli.

Il controllo logico dei motori, è affidato al PIC, mentre l'attuazione ad U4, un driver capace di controllare in maniera ottimale i due motoriduttori. Grazie al driver, il microcontrollore può decidere anche il senso di rotazione dei motori sce-

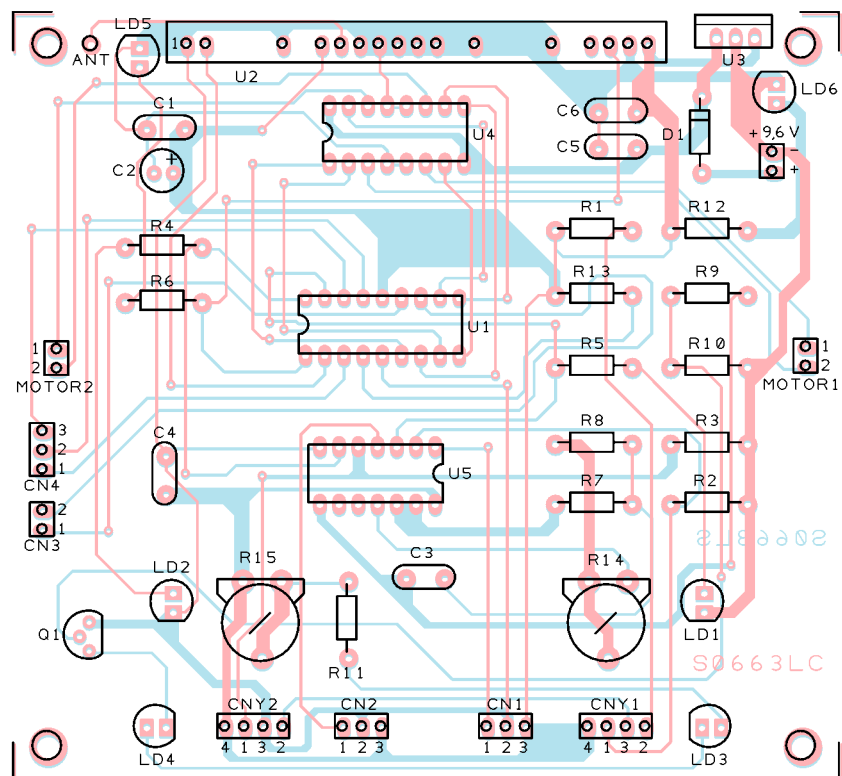
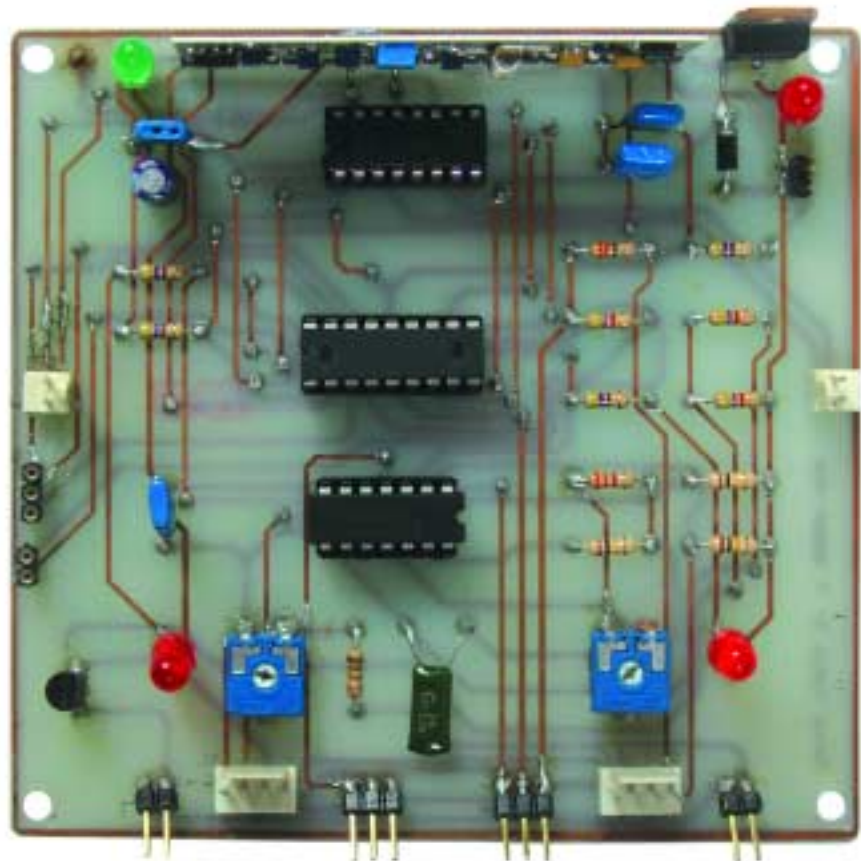
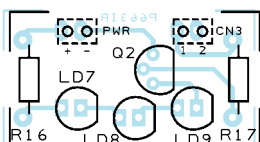


ELENCO COMPONENTI:

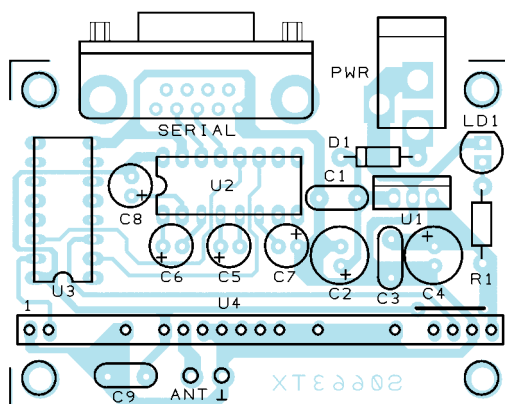
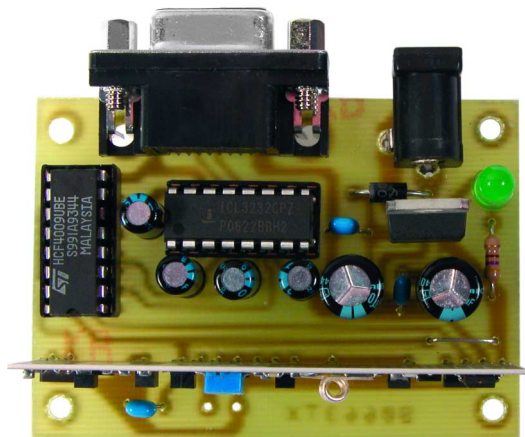
R1: 330 ohm
 R2, R3: 100 kohm
 R4÷R6: 470 ohm
 R7: 100 kohm
 R8, R9: 47 kohm
 R10: 4,7 kohm
 R11: 100 ohm
 R12: 470 ohm
 R13: 4,7 kohm
 R14: trimmer MO 4,7 kohm
 R15: trimmer MO 470 ohm
 R16: 180 ohm
 R17: 12 kohm
 C1: 100 nF multistrato
 C2: 10 µF 63 VL elettrolitico
 C3: 1,8 nF multistrato
 C4÷C6: 100 nF multistrato
 U1: PIC16F818 (MF663)
 U2: RTFSAW
 U3: 7805
 U4: L293D
 U5: 7400
 CN1, CN2: IR38DM
 CN3: interfaccia IR (P663IR)
 CNY1: CNY70
 CNY2: CNY70
 Q1, Q2: BC337
 D1: 1N4007
 LD1÷LD2: led 5 mm rosso
 LD3÷LD4: led 3 mm IR
 LD5: led 5 mm verde
 LD6: led 5 mm rosso
 LD7÷LD9: led 5 mm bianco
 M1÷M2: motoriduttore Micro-Motors
 L149.12.43

Varie:

- Portabatterie per 4 stilo (2 pz.)
- Zoccolo 7+7
- Zoccolo 8+8
- Zoccolo 9+9
- Strip maschio 2 pin (2 pz.)
- Circuiti stampati



L'interfaccia radio per PC

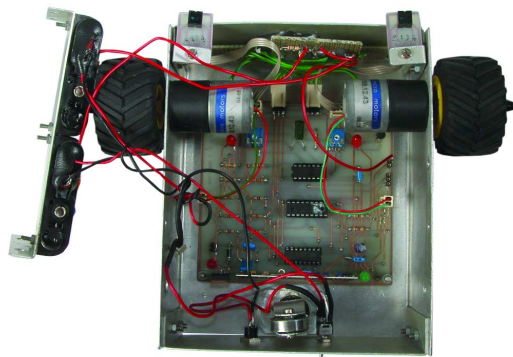


COMPONENTI

R1: 470 ohm
 C1, C3: 100 nF multistrato
 C2, C4: 470 μ F 25 VL
 elettrolitico
 C5÷C8: 1 μ F 63 VL
 elettrolitico
 U1: 7805
 U2: MAX232
 U3: 4009
 U4: RTFSAW
 D1: 1N4007
 LD1: led 5 mm verde

Varie:

- Plug alimentazione
- Zoccolo 8+8 (2 pz.)
- Connettore DB9 femmina



Sulla basetta del robot è montato un modulo ricetrasmittitore RTFSAW. Qualora si intendesse usare il robot solo in modalità automatica si può evitarne il montaggio. In caso contrario, con un secondo modulo RTFSAW, potrete preparare l'unità di controllo radio da abbinare al personal computer e collegata alla porta seriale. In questa pagina trovate anche l'elenco dei componenti necessari per quest'ultima. Nella foto sopra: un dettaglio della ruota pivotante (con i relativi accessori) che permette al robot di girare.

gliando così se far avanzare, ruotare o far tornare indietro il robot, impostando opportunamente lo stato logico degli ingressi IN1, IN2, IN3 e IN4 di U4. Nella Tabella 2 mostriamo come devono essere impostati questi ingressi per far muovere il robot. Anche se collegati al microcontrollore, i pin di enable non sono gestiti. Il modulo U2, un ricetrasmittitore RTFSAW dell'AUREL, permette l'invio e la ricezione di dati ad una velocità massima di 2400 baud. Questo ci consente di controllare il robot via radio da un PC, utilizzando un'apposita scheda da collegare alla porta seriale del computer ed il software scaricabile gratuitamente dal sito www.elettronica.in.it. In Tabella 3 riportiamo il riepilogo delle funzioni dei vari pin del microcontrollore.

La struttura meccanica

Il corpo del robot è stato ricavato

modificando un contenitore Teko 383 dalle dimensioni di 16 x 15 x 7 cm; il movimento è dato da due micromotori collegati a due ruote di trazione. Dal punto di vista meccanico, il pezzo tecnologicamente più complesso è la ruota pivotante, rea-

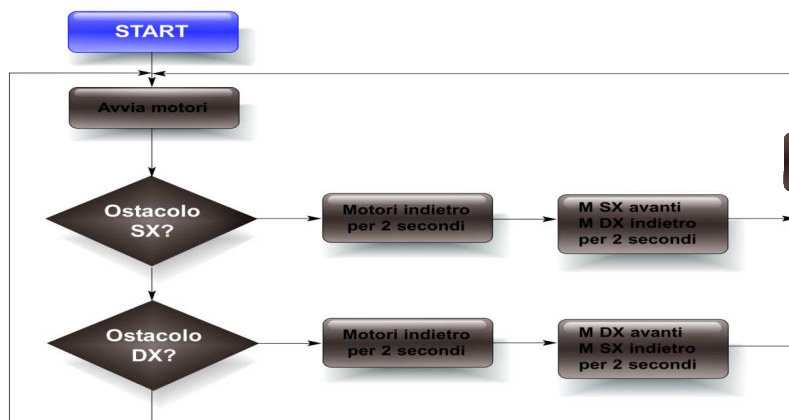


Fig. 4

Il diagramma illustra il comportamento del robot in presenza di ostacoli. Il cambiamento di direzione non è istantaneo, è prevista una pausa di 100 mSec dopo che i motori si sono fermati e prima di riprendere il moto.

LISTATO 1

```

'* Name      : SOFTW3.BAS                                     *
'* Author    : Penzo Fabio                                     *
'* Notice    : ROBOT FOX software vers3                       *
'* Date      : 07/06/2006                                       *
'* Version   : 1.0                                              *
'* Notes     : PIC16F818                                         *
'*****
OSCCON = $60           ' Imposta clock interno a 4MHz
ADCON1 = $06           ' PORTA tutti pin digitali
TRISA = %11111111     ' RA0-RA7 in
TRISB = %00000000     ' RB0-RB7 out
PORTB.6 = 1           ' lampeggio del faretto
PORTB.7 = 1
pause 500
PORTB.6 = 0
PORTB.7 = 0
pause 500
PORTB.6 = 1
PORTB.7 = 1
PORTB.6 = 1
PORTB.7 = 1
pause 500
PORTB.6 = 0
PORTB.7 = 0
pause 500
PORTB.6 = 1
PORTB.7 = 1
pause 500
loop:                  'inizio routine di controllo
If PORTA.0 = 0 and PORTA.1 = 0 then goto avanti              'nessun ostacolo
rilevato
If PORTA.0 = 1 then goto ostdes      'ostacolo rilevato a sinistra
If PORTA.0 = 1 and PORTA.1 = 1 then goto indietro           'ostacolo rile-
vato da entrambi i lati
if PORTA.1 = 1 then goto ostsin      'ostacolo rilevato a destra
pause 200
avanti:                  'inizio ciclo di avanzamento
PORTB.0 = 1             'attivo enable1
PORTB.1 = 1             'attivo enable2
PORTB.2 = 1             'Motor1 avanti
PORTB.3 = 0
PORTB.4 = 1             'Motor2 avanti
PORTB.5 = 0
PAuse 200               'pausa di 1 secondo
goto loop               'torna alla routine di controllo
      ostdes:           'inizio ciclo :rilevato ostacolo a destra
      PORTB.0 = 1       'attivo enable1
      PORTB.1 = 1       'attivo enable2
      PORTB.2 = 0       'Motori stop
      PORTB.3 = 0
      PORTB.4 = 0       'Motori stop
      PORTB.5 = 0
      pause 100         'pausa 0,1 secondi
      PORTB.2 = 0       'Motor1 indietro
      PORTB.3 = 1
      PORTB.4 = 0       'Motor2 indietro

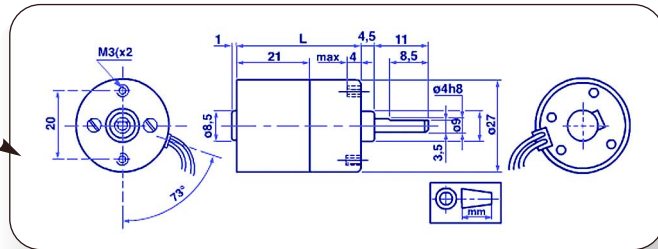
```

lizzata con due cuscinetti che funzionano come le ruote anteriori dei carrelli del supermercato. In queste pagine trovate le foto che mostrano come è stata realizzata e da cui potete prendere spunto.

I sensori all'infrarosso CNY70 vanno fissati in modo da sfiorare la superficie di appoggio: permettono il riconoscimento del bordo del tavolo e sono essenziali per evitare cadute accidentali durante l'avanzamento. I due led IR abbinati ai due ricevitori per infrarosso devono essere posizionati sulla parte frontale del robot avendo l'accortezza di non montare le due coppie troppo vicine. Per completare la parte anteriore, con finalità esclusivamente estetica, troviamo tre led a luce bianca ad alta intensità che si attivano all'accensione del circuito dietro comando del microcontrollore che portando alta la sua porta RB7 manda in saturazione il transistor Q2.

Modalità di funzionamento

La configurazione d'uso del robot è flessibile: in base a come avete intenzione di utilizzarlo, dovete programmare opportunamente il microcontrollore. Tre diversi firmware, ed il programma per il con- ➤



Nel prototipo del Robot FOX sono stati utilizzati dei motoriduttori della Micro-Motors della serie L149; nelle immagini alcuni particolari della loro struttura e nel disegno i dettagli meccanici. Il modello da noi utilizzato presenta una coppia massima di 3,8 Ncm con un rapporto di 43,3:1. Ulteriori informazioni sono disponibili sul sito: <http://www.micromotorssl.com>

(continuazione Listato 1)

```

PORTB.5 = 1
PAUSE 2000
PORTB.2 = 0
PORTB.3 = 0
PORTB.4 = 0
PORTB.5 = 0
PAUSE 100
PORTB.2 = 0
PORTB.3 = 1
PORTB.4 = 1
PORTB.5 = 0
PAUSE 1800
goto loop
    indietro:
entrambi i lati
PORTB.2 = 0
PORTB.3 = 0
PORTB.4 = 0
PORTB.5 = 0
PORTB.0 = 1
PORTB.1 = 1
PORTB.2 = 0
PORTB.3 = 1
PORTB.4 = 0
PORTB.5 = 1
PAUSE 2000
PORTB.2 = 0
PORTB.3 = 1
PORTB.4 = 1
PORTB.5 = 0
pause 1800
goto loop
    ostsin:
PORTB.0 = 1
PORTB.1 = 1
PORTB.2 = 0
PORTB.3 = 0
PORTB.4 = 0
PORTB.5 = 0
pause 100
PORTB.2 = 0
PORTB.3 = 1
PORTB.4 = 0
PORTB.5 = 1
pause 2000
PORTB.2 = 0
PORTB.3 = 0
PORTB.4 = 0
PORTB.5 = 0
PAUSE 100
PORTB.2 = 1
PORTB.3 = 0
PORTB.4 = 0
PORTB.5 = 1
pause 1800
GOTO loop
END

```

```

'pausa 2 secondi
'Motori stop
'Motori stop
'attendi 0,1 secondi
'Motor1 indietro
'Motor2 avanti
'attendi 0,1 secondi
'torna alla routine di controllo
'inizio ciclo :rilevato ostacolo da
'Motori stop
'Motori stop
'attivo enable1
'attivo enable2
'Motor1 indietro
'Motor2 indietro
'attendi 2 secondi
'Motor1 indietro
'Motor2 avanti
'attendi 1,8 secondi
'torna alla routine di controllo
'inizio ciclo :rilevato ostacolo a sinistra
'attivo enable1
'attivo enable2
'Motori stop
'Motori stop
'pausa 0,1 secondi
'Motor1 indietro
'Motor2 indietro
'pausa 2 secondi
'Motori stop
'attendi 0,1 secondi
'Motor1 avanti
'Motor2 indietro
'attendi 1,8 secondi
'torna alla routine di controllo

```



Fig. 5

mente autonomo: evita gli ostacoli che trova lungo il suo percorso e controlla continuamente che la superficie su cui si muove non finisca.

Nel secondo caso il robot viene controllato dal PC: è possibile muoverlo utilizzando l'apposito software, i sensori però non vengono gestiti.

Programmando il microcontrollore con il terzo firmware il robot si muove autonomamente, ma può essere disattivato o riattivato tramite PC.

In queste pagine riportiamo il firmware relativo al funzionamento automatico e, in Figura 5, un dettaglio della schermata video del programma di controllo del robot tramite PC; si noti che non è necessario effettuare alcun tipo di configurazione, basta selezionare la porta seriale di comunicazione (COM1, COM2) a cui è collegato il trasmettitore.

Il montaggio delle basette ed i cablaggi non richiedono particolari attenzioni, mentre per l'aspetto meccanico serve certamente una buona manualità.

trollo via radio tramite PC, sono disponibili all'indirizzo:

www.elettronica.in.it.

I sorgenti scaricabili prevedono tre modi di funzionamento:

a) funzionamento automatico;

b) gestione manuale via radio;

c) modalità automatica con

attivazione via radio.

Nel primo caso il robot è completa-

Per il MATERIALE

Il firmware, il software e tutti i disegni meccanici di Robot FOX sono scaricabili gratuitamente dal sito della rivista (www.elettronica.in.it). I componenti utilizzati in questo progetto sono facilmente reperibili in commercio. Per quanto riguarda il modulo radio, i sensori e tutti gli elementi meccanici è possibile rivolgersi alla ditta Futura Elettronica.

**Il materiale va richiesto a: Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 ~ Fax: 0331-778112 ~ <http://www.futuranet.it>**

Tibbo FTP per dati meteo

di Carlo Tauraso

Utilizziamo l'opportunità offerta dalla programmabilità dei Device Server della Tibbo per realizzare un sistema in grado di pubblicare via FTP su qualsiasi pagina Web i dati meteorologici forniti dalla WS2300, la più nota centralina meteo della La Crosse Technology. Seconda puntata.



Nella precedente puntata abbiamo iniziato a descrivere un interessante progetto che permette di pubblicare in rete i parametri meteo della stazione WS-2300 della La Crosse Technology utilizzando un modulo Tibbo DS-202 opportunamente riprogrammato. Questo dispositivo, nato come Device Server, grazie all'aggiornamento del sistema operativo, può adesso essere utilizzato come controller Ethernet programmabile. Nel precedente articolo abbiamo spiegato sia come predisporre il dispositivo per la programmazione, sia la prima parte del progetto relativa alla illustrazione

del protocollo di comunicazione utilizzato dalla WS-2300. Il codice da noi messo a punto permette di trasferire tutti i parametri meteo via FTP utilizzando un file denominato *meteo.dat*. Abbiamo analizzato i diversi record di questo file in relazione ai corrispondenti parametri monitorati. Ora dobbiamo terminare il discorso affrontando gli ultimi tre record che riguardano la velocità del vento, la sua direzione e la pressione atmosferica. Terminata la descrizione del formato di uscita, procederemo con l'analisi dettagliata del listato sorgente utilizzato per la programmazione del Tibbo DS-202. ➤

Velocità del vento e Direzione (0527h - Sequenza 130, 150, 138, 158, 218)

Il DS-202 richiede 6 byte corrispondenti a 12 locazioni come si vede in Tabella 1. Nel file *meteo.dat* il record è formato da 8 byte. Il primo deve essere pari a 36h (3 + 6 byte da ricevere). Il byte successivo contiene due flag uno relativo alla condizione di overflow ed uno relativo al simbolo visualizzato nel caso la velocità sia inferiore al minimo impostato. La velocità del vento è espressa in metri al secondo (m/s) ed è suddivisa in 3 nibble. Ad esempio, se nella sequenza binaria troviamo il valore 013h = 10011b = 19 la velocità sarà di 1,9 m/s. Per convertirla in Km/h dovete moltiplicare questo valore per 3,6, quindi si ottiene 6,84 Km/h. Le 6 locazioni successive contengono la direzione del vento in gradi multipli di 22,5°. In pratica ciascun nibble deve essere moltiplicato per 22,5° per ottenere l'angolo compreso tra la direzione Nord e quella da cui spira il vento. I gradi partono da 0° corrispondente al Nord e vengono incrementati in senso orario, quindi Est=90° Sud=180° Ovest=270°. Questa è la cosiddetta notazione azimutale utilizzata in ambito meteorologico. Se nel primo byte considera-

Tabella 1

Indirizzo	Descrizione
0527	Overflow Flag se = 0 la misura è corretta
0528	Codice visualizzazione valore velocità inferiore al minimo 0=min 1=___ 2=OFL
0529	Velocità nibble 0 m/s
052A	Velocità nibble 1 m/s
052B	Velocità nibble 2 m/s
052C	Direzione nibble * 22,5°
052D	Direzione 1 misurazione fa
052E	Direzione 2 misurazioni fa
052F	Direzione 3 misurazioni fa
0530	Direzione 4 misurazioni fa
0531	Direzione 5 misurazioni fa
0532	Non assegnato



to troviamo nel nibble più significativo il valore 0100b = 4, l'angolo sarà pari a 4*22,5= 90° quindi la direzione sarà EST. Vediamo a lato come si

presenta la rosa, completa di gradi e nomi dei venti. Si faccia attenzione che per effettuare una misura esatta di direzione e velocità è necessario che l'anemometro sia posizionato a 10 metri di altezza con terreno pianeggiante e privo di ostacoli (campo aperto). Un anemometro fissato su un balcone darà dei risultati sicuramente errati. Per problemi dovuti all'attrito, la velocità del vento varia sensibilmente tra 0 e 10 metri di altezza. Sempre a proposito di come effettuare le misure, ricordiamo che si definisce campo aperto un'area nella quale la distanza tra l'anemometro e qualunque ostacolo sia almeno 10 volte l'altezza dell'ostacolo. Pertanto, misure eseguite ad altezze diverse devono essere corrette per convertire tali dati all'altezza standard. Considerate bene il modo in cui installate il vostro strumento altrimenti finirete per misurare soltanto turbolenze generate in corrispondenza del muro del vostro appartamento. I nibble che troviamo nei byte successivi contengono gli ultimi 5 rilievi effettuati. In Tabella 2 trovate il dettaglio di un rilievo effettuato su bava di vento Levante.

Tabella 2

Byte ricevuto	Nr sequenza	Descrizione
36h	1	3h=Prefisso costante 6h=Nr byte dati da ricevere
00h	2	Simbolo = 0 Overflow Flag = 0
13h	3	Nibble[0]= 3 Nibble[1]=1
40h	4	Nibble[2]=0 pertanto il valore finale è pari a 1,9m/s Direzione corrente = 4 quindi 4*22,5=90° EST
44h	5	Direzione 1 misurazione fa = 4 quindi 4*22,5=90° EST Direzione 2 misurazioni fa = 4 quindi 4*22,5=90° EST
44h	6	Direzione 3 misurazioni fa = 4 quindi 4*22,5=90° EST Direzione 4 misurazioni fa = 4 quindi 4*22,5=90° EST
04h	7	Direzione 5 misurazioni fa = 4 quindi 4*22,5=90° EST Il nibble più significativo non è assegnato
DFh	8	00h + 13h + 40h + 44h + 44h + 04h = DFh

Pressione assoluta (05D8h - Sequenza 130, 150, 182, 162, 206)
Il DS-202 richiede 3 byte corrispon-

Tabella 3

Indirizzo	Descrizione
05D8	hPa decimi
05D9	hPa unità
05DA	hPa decine
05DB	hPa centinaia
05DC	hPa migliaia
05DD	Non assegnato

denti a 6 locazioni (vedi Tabella 3). Nel file *meteo.dat* il record è formato da 5 byte. Il primo deve essere pari a 33h (3 + 3 byte da ricevere). I byte successivi contengono il valore della pressione in hPa (Hecto-Pascal) diviso in 5 nibble che registrano in sequenza, decimi, unità, decine, centinaia, migliaia. L'ultimo byte è il solito checksum. L'Hecto-Pascal è esattamente equivalente al millibar che sicuramente tutti voi avrete sentito nominare. Ricordiamo che questo parametro rileva la pressione atmosferica nel luogo in cui si trova il sensore senza alcun tipo di modifica. Nei bollettini meteo si utilizza generalmente la pressione relativa che vedremo nel prossimo paragrafo. In Tabella 4 possiamo vedere un rilevamento nel dettaglio.

RECORD NR 15 (Lunghezza 5 byte)

Pressione relativa (05E2h - Sequenza 130, 150, 186, 138, 206)

Il DS-202 richiede 3 byte corrispondenti a 6 locazioni (vedi Tabella 5). Nel file *meteo.dat* il record è formato da 5 byte. Il primo deve essere pari a 33h (3 + 3 byte da ricevere). I byte successivi contengono il valore della pressione relativa in hPa (Hecto-Pascal) diviso in 5 nibble che registrano in sequenza, decimi, unità, decine, centinaia, migliaia. L'ultimo

Tabella 4

Byte ricevuto	Nr sequenza	Descrizione
33h	1	3h=Prefisso costante 3h=Nr byte dati da ricevere
69h	2	9 decimi hPa 6 unità hPa
00h	3	0 decine hPa 0 centinaia hPa
31h	4	1 migliaia quindi il valore complessivo è 1006,9 hPa o millibar. Il valore del nibble più significativo non è assegnato.
9Ah	5	69h + 00h + 31h = 9Ah

Tabella 5

Indirizzo	Descrizione
05E2	hPa decimi
05E3	hPa unità
05E4	hPa decine
05E5	hPa centinaia
05E6	hPa migliaia
05E7	Non assegnato

byte è il solito checksum. Questo parametro viene calcolato riferendo la pressione assoluta al livello del mare. Come sicuramente saprete la pressione atmosferica dipende dall'altitudine. Poiché il valore della pressione atmosferica in un punto è legato al peso della colonna d'aria sovrastante, salendo di quota tale valore diminuisce. Quindi se rileviamo la pressione assoluta in una località con un'altitudine di 1.000m dovremo incrementare tale valore per stimare lo stesso parametro a livello del mare. Non esiste una rela-

Tabella 6

Byte ricevuto	Nr sequenza	Descrizione
33h	1	3h=Prefisso costante 3h=Nr byte dati da ricevere
06h	2	6 decimi hPa 6 unità hPa
01h	3	1 decine hPa 0 centinaia hPa
41h	4	1 migliaia quindi il valore complessivo è 1010,6 hPa o millibar. Il valore del nibble più significativo non è assegnato.
48h	5	06h + 01h + 41h = 48h

zione precisa tra la quota e la pressione, poiché altri fattori come la temperatura (una colonna d'aria calda pesa di meno di un'analogica colonna di aria fredda) e l'umidità (una colonna d'aria umida pesa più di una colonna di aria secca) possono influenzare i risultati.

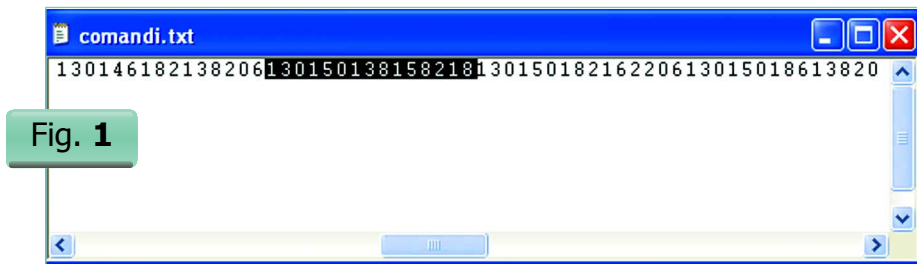
Approssimativamente, alle basse quote si può assumere un gradiente verticale della pressione (cioè una

variazione della pressione con la quota) pari a: 1 hPa ogni 8 metri. Pertanto se la pressione assoluta a 1.000m è pari a 980hPa la pressione relativa sarà pari a: $980 + (1.000/8) = 980 + 125 = 1.105$ hPa. Questa correzione permette chiaramente di riportare i valori rilevati in diverse località ad una base comune per essere comparati. Proprio per questo motivo nei bollettini meteo si utilizza il valore relativo e non quello assoluto. In Tabella 6 trovate un rilievo completo.

Firmware: l'interfaccia seriale

Questa prima sezione del firmware si occupa della comunicazione con la stazione meteorologica. Il Tibbo permette di gestire la porta seriale direttamente attraverso metodi e proprietà dell'oggetto *ser*. Premettiamo che

lo sviluppo di cui stiamo parlando è stato effettuato sulla versione 1.10 del TIOS scaricabile gratuitamente, assieme al TIDE, dal sito della Tibbo. L'ambiente di programmazione Taiko è infatti nato da poco e risulta essere in continua evoluzione: è bene precisare pertanto le condizioni in cui è stato utilizzato. Come abbiamo visto nella scorsa puntata la condizione fondamentale per iniziare a comunicare con la WS-2300 è lo stato logico delle due linee RTS e DTR. In questo caso siamo fortunati perché un DS-202 con TIOS 1.10 mantiene all'avvio RTS positiva e DTR negativa pertanto è necessario soltanto configurare la porta a 2400bps, 8 bit, no parità, 1 bit di stop. Tutta la programmazione dei ➤



Tibbo avviene per eventi, quindi abbiamo raccolto questa prima parte direttamente nella *sys.init* procedura che viene eseguita all'avvio del dispositivo. Ricordatevi sempre che in realtà sui Tibbo il vostro codice girerà in una VM controllata dal processo Master foraggiata da una coda sequenziale di eventi che devono essere gestiti rapidamente. Non state quindi sviluppando l'intero firmware del dispositivo come avviene per un PIC ma utilizzate le funzioni e gli oggetti messi a disposizione dal sistema operativo per effettuare determinate elaborazioni. Se da un lato questa procedura non permette di avere un controllo a basso livello del codice eseguito, è senz'altro molto apprezzabile per l'alta produttività risultante e la grande semplicità con cui è possibile affrontare uno sviluppo di rete. Dopo questa doverosa parentesi destinata a quanti si avvicinano per la prima volta alla programmazione su Tibbo e ai quali consigliamo di leggere la documentazione disponibile sul sito www.tibbo.com continuiamo nell'analisi del codice. Dopo aver stabilito i parametri di comunicazione della porta, definiamo i buffer di ricezione e trasmissione che utilizzeremo per le sequenze che invieremo alla stazione e per registrare le relative risposte. La funzione *sys.buffalloc* comporta un certo tempo di esecuzione perciò è bene inserirla all'avvio del sistema per evitare che tale ritardo possa inficiare la comunicazione. Lo stesso discorso vale anche per i *socket* utilizzati nella trasmissione Ethernet, proprio per evitare che un server con un time-out un po' troppo

spinto possa interrompere la sessione prima che i buffer vengano allocati. La porta seriale viene aperta e il DS-202 inizia ad elaborare il *resource-file comandi.txt*. Questo file contiene le sequenze da inviare al dispositivo divise in gruppi da 5 byte come abbiamo illustrato dettagliatamente nei precedenti paragrafi. Per personalizzare i parametri d'interrogazione è sufficiente quindi entrare nel TID, fare doppio click sul file, rimuovere una sequenza, modificarla o inserirne un'altra. Il file *meteo.dat* in uscita cambierà di conseguenza. Nell'immagine di Figura 1 si vede una fase di editing del file con in evidenza la sequenza che richiede la

velocità e direzione del vento. L'accesso al resource file è possibile attraverso l'utilizzo dell'oggetto *rom-file*. Come si vede chiaramente nel Listato 1 il programma effettua una lettura a gruppi di 3 caratteri che vengono attestati nel vettore *com*. Esso contiene esattamente i 5 byte di un comando completo. L'array viene poi utilizzato dalla procedura WSCOM che è il nucleo di questa sezione. Si faccia attenzione che all'apertura del file il programma si aspetta un gruppo di comandi costituiti ciascuno da 15 caratteri e non viene effettuato un controllo sulla congruenza di questo dato. Se il file non è stato costruito in maniera corretta il programma semplicemente divide la sequenza in gruppi di 15 ignorando tutto ciò che va al di fuori di questa struttura. Naturalmente è possibile inserire un controllo che segnali all'utente questa condizione, noi l'abbiamo ritenuto superfluo. La procedura WSCOM si occupa di inviare ciascun comando alla centra-

LISTATO 1

```
sub on_sys_init
    ser.num = 0
    ser.baudrate = ser.div9600*4
    ser.bits = PL_SER_BB_8
    ser.parity = PL_SER_PR_NONE
    ser.rxbufrrq(1)
    ser.txbufrrq(1)
    sys.buffalloc
    ser.enabled = YES
inizio:
    pat.play("R",PL_PAT_CANINT)
    romfile.open("comandi.txt")
    tot=(romfile.size)/15
    addr=1
    for q= 1 to tot
        for i=0 to 4
            s=romfile.getdata(3)
            com(i)=val(s)
        next i
        WSCOM()
        stor.set(s,addr)
        addr=addr+i
    next q
end sub
```

La porta seriale viene configurata a 2400-8-N-1

Si allocano 256 byte per i buffer di ricezione e trasmissione

La porta seriale viene aperta

Il led rosso viene acceso e il DS-202 inizia a leggere i comandi dal resource-file. Il numero di sequenze da elaborare viene calcolato dividendo per 15 la lunghezza dell'archivio. (1 comando = 5 gruppi da 3 caratteri). Le sequenze vengono registrate nell'array com.

La procedura WSCOM si occupa dell'invio di ciascun comando alla stazione meteo. La risposta registrata nella stringa s viene trasferita nella EEPROM interna del DS-202.

lina meteo gestendo il reset e le eventuali pause indotte dalla WS-2300. Come si vede chiaramente nel listato non viene fatto alcun tipo di interpretazione dei dati in risposta, dati che una volta trasferiti alla stringa *s* vengono direttamente inseriti nella EEPROM interna richiamando il metodo *set* dell'oggetto *stor*. Questo fatto permette di dare una completa libertà all'utente nella personalizzazione dei comandi inviati alla stazione meteo. Il codice si comporta come una sorta di interprete, prende le domande dal file *comandi.txt* le traduce alla stazione e rinvia al file *meteo.dat* le risposte ricevute. Se avessimo inserito delle istruzioni di conversione è evidente che avremmo dovuto modificarle ogni volta che un utente avesse voluto variare il numero di byte da leggere nella memoria della WS-2300. In questo modo, invece, è sufficiente fare riferimento ai paragrafi precedenti ed alle tabelle incluse per interpretare correttamente i dati contenuti in *meteo.dat*. I più curiosi possono dare un'occhiata al file che contiene la mappa completa dei *nibble* contenuti nella memoria della *memory_map_2300.doc*. Tutta la comunicazione con la centralina avviene all'interno della procedura WSCOM. Il primo problema da risolvere è che la centralina meteo non considera come prioritaria la comunicazione attraverso la porta seriale. Pertanto ogni volta che si vuole leggere una sequenza di locazioni bisogna vedere se la stessa è pronta a risponderci altrimenti le nostre richieste cadranno nel vuoto. Questa cosa può succedere anche nel bel mezzo di una sequenza. La procedura migliore per rendere la comunicazione più efficace è quella di inviare un comando di reset (06h) ad ogni richiesta ed attendere di ricevere la risposta corretta corrispondente ad un byte pari a 02h. Inoltre, per ogni byte inviato, bisogna controllare che ci sia un byte di risposta altrimenti è necessario effettuare nuova-

LISTATO 2

```
sub WSCOM ()
reset:
  doevents
  ser.setdata(chr(6))
  ser.send
  w=sys.timercount
  while ser.rxlen=0
  if sys.timercount-w > 2 then goto reset
  doevents
  wend
  s = ser.getdata(255)
  if asc(left(s,1))<> 2 then goto reset
  for i=0 to 3
  ser.setdata(chr(com(i)))
  ser.send
  w=sys.timercount
  while ser.rxlen=0
  if sys.timercount-w > 2 then goto reset
  doevents
  wend
  s = ser.getdata(255)
  next i
  ser.setdata(chr(com(4)))
  ser.send
  i=(com(4)-194)/4 + 2
  w=sys.timercount
  while ser.rxlen<i
  if sys.timercount-w > 2 then goto reset
  doevents
  wend
  s = ser.getdata(255)
end sub
```

Dopo l'invio del reset il DS-202 controlla di continuo il buffer di ricezione per vedere se arriva la risposta da parte della stazione.

Nel caso non si riceva risposta entro un secondo l'esecuzione ritorna al reset. Nel frattempo viene forzata la gestione degli eventi in coda.

Il codice controlla la risposta della WS-2300 se è corretta inizia il ciclo di invio del comando estratto dal file *comandi.txt*.

Durante il primo ciclo vengono inviati i 4 byte iniziali della sequenza.

Dopo la ricezione della risposta al quarto byte si invia il quinto calcola il numero di byte da ricevere nella variabile *i*.

mente il reset. Infine, non si può inviare il byte successivo senza aver ricevuto la risposta a quello precedente. In una situazione del genere utilizzare la gestione eventi del Tibbo non è una cosa immediata, soprattutto perché il gestore dovrebbe essere in grado di capire in che stato di connessione ci si trova. Durante la sessione FTP utilizzeremo proprio questa soluzione per gestire il trasferimento del file *meteo.dat*. Durante tale sessione siamo facilitati dalle risposte ricevute dal server che contengono un numero a tre cifre specifico per ciascuna situazione. Per rendere la cosa più semplice abbiamo quindi concentrato il tutto in un'unica procedura che utilizza il polling della seriale per decidere come pro-

cedere. L'oggetto *ser*, infatti, mette a disposizione una proprietà in sola lettura chiamata "rxlen" che permette proprio di capire se la seriale ha ricevuto qualche carattere. Nell'attesa è possibile valutare il tempo che scorre interrogando un contatore chiamato "timercount" che viene incrementato ogni 500ms. Nel caso in cui la stazione non risponda entro un secondo, il sistema entra in time-out e ricomincia la sequenza inviando il comando di reset. Nel caso in cui il blocco avvenga proprio durante una sequenza, non ricomincia tutto daccapo perché il sistema utilizza l'indice *q* per sapere il comando da inviare. In pratica esso ripete soltanto l'ultimo comando trasmesso finché non riceve risposta ➤

(Listato 2). Si faccia attenzione che il codice della WSCOM() è in realtà costituito da 3 cicli successivi: il primo gestisce il reset, il secondo invia i primi 4 byte e l'ultimo invia il quinto byte attendendo i dati relativi. Per quale motivo abbiamo diviso la sequenza? La spiegazione è molto semplice: i primi quattro byte hanno una risposta costante di un singolo byte mentre il quinto byte informa la stazione del numero di locazioni da leggere pertanto la risposta varia in lunghezza e viene calcolata sulla base dell'ultimo byte trasmesso. Anche la risposta è sottoposta al time-out di un secondo per evitare il blocco in ricezione. Una volta effettuata la registrazione nella EEPROM interna del DS-202, l'esecuzione passa alla seconda sezione che si occupa dell'interfaccia Ethernet.

Firmware: l'interfaccia Ethernet

Questa sezione effettua il trasferimento dei dati registrati nella EEPROM interna del DS-202 verso un Server FTP presente in Internet. Lo sviluppo è stato realizzato prevedendo un architettura di tipo FSM (Finite State Machine) le cui transizioni sono stabilite a seconda delle risposte ricevute dal processo server. L'ambiente di sviluppo presentato da Tibbo rende le cose estremamente semplici visto che esistono due oggetti: *net* e *socket* che attraverso metodi e proprietà permettono di effettuare tutte le operazioni necessarie. Con l'oggetto *net* configuriamo l'interfaccia Ethernet del DS-202 stabilendo l'indirizzo IP e la maschera di sottorete. Con l'oggetto *sock*, invece, è possibile gestire fino a 16 socket differenti. La cosa interessante è che in realtà essi vengono creati direttamente dal TIOS che li rende disponibili ad un livello più alto. Questo permette di tagliare una bella porzione di codice di basso livello permettendo di focalizzare la nostra attenzione sull'implementazione del

protocollo a livello applicativo. Interessanti sono anche le funzionalità di filtro dei messaggi in ingresso, la possibilità di associare agli *stream* la gestione automatica di comandi che però non abbiamo utilizzato in questo progetto iniziale ma che sicuramente riprenderemo in altre occasioni. I più curiosi e chi non vuole aspettare troverà tutte le informazioni necessarie all'interno della documentazione fornita da Tibbo. Già utilizzando la pad "Browser" del Tide è possibile avere una descrizione sintetica di ciascun metodo e proprietà. Ma veniamo al dunque. Durante l'avvio del dispositivo precisiamo la configurazione dell'interfaccia Ethernet e prepariamo i due socket (Canale Dati e Canale Comandi) che serviranno per la sessione FTP. A ciascun socket è possibile associare un buffer

di ricezione ed uno di trasmissione proprio come per la porta seriale. Ciascun socket è indicizzato attraverso il parametro *num* pertanto per accedere allo stesso è sempre necessario definire prima questo indice, soprattutto all'interno di eventi come *on_sock_data_arrival* che modificano il numero del socket corrente. In pratica se valorizziamo *num* a 1, tutte le istruzioni successive si riferiranno al socket numero 1. Nel nostro listato definiamo il socket numero 0 per il canale Comandi e il socket numero 1 per il Canale Dati. Entrambe utilizzano come protocollo di trasporto il TCP ed entrambe hanno associato un buffer di ricezione ed uno di trasmissione in grado di gestire due pagine da 256 byte. In realtà la grandezza deve essere diminuita di 16 byte che sono utilizzati internamente dal

LISTATO 3

```
sub on_sys_init
```

```
net.gatewayip = "192.168.0.1"
net.netmask = "255.255.255.0"
net.ip = "192.168.0.5"
utente = "tibbo"
pwd = "prova"
rit = 180
stato = 0
```

Configurazione dell' interfaccia Ethernet.

Valorizzazione delle variabili relative alle credenziali da utilizzare durante la sessione FTP. Successivamente viene stabilito il numero di tick di sistema (500ms) utilizzati come intervallo tra una sessione e l'altra. La FSM viene gestita attraverso la variabile stato.

```
sock.num = 0
sock.txbufreq(2)
sock.rxbufreq(2)
sys.buffalloc
sock.protocol = PL_SOCKET_PROTOCOL_TCP
sock.targetip = "192.168.0.15"
sock.targetport = 21
```

Socket 0 riservato al canale Comandi. Si allocano i buffer di ricezione e trasmissione. Viene precisato come protocollo di trasporto il TCP e si definiscono come destinazione l'indirizzo IP del server FTP e la well-known port 21. I parametri inconmode e reconmode rimangono quelli di default.

```
sock.num = 1
sock.txbufreq(2)
sock.rxbufreq(2)
sys.buffalloc
sock.protocol = PL_SOCKET_PROTOCOL_TCP
sock.inconmode = PL_SOCKET_INCONMODE_ANY_IP_ANY_PORT
sock.reconmode = PL_SOCKET_RECONMODE_3
sock.localportlist = "1012"
```

Socket 1 riservato al canale Dati. Si dà la massima libertà per quanto riguarda connessioni e riconessioni stabilendo come porta di ascolto la 1012.

```
end sub
```

TIOS. L'allocazione della memoria relativa avviene attraverso la chiamata *sys.buffalloc* che viene eseguita in fase di inizializzazione per evitare problemi durante la sessione FTP. Un ritardo nell'allocazione potrebbe comportare un blocco nella comunicazione, soprattutto se il time-out del server è piuttosto basso. Le proprietà "inconmode" e "reconmode" permettono di stabilire come gestire i messaggi in entrata e gli episodi di riconnessione su un ben determinato socket. Per il collegamento abbiamo utilizzato la configurazione più ampia possibile permettendo l'accesso da qualsiasi IP e da qualsiasi porta. Naturalmente è possibile rendere il sistema più preciso stabilendo che la connessione possa essere effettuata soltanto dall'IP specificato nella proprietà "targetip" o attraverso la porta specificata nella proprietà "targetport". Per quanto riguarda la riconnessione abbiamo sempre utilizzato la metodologia più ampia autorizzando qualsiasi IP e qualsiasi porta. Naturalmente tutto ciò vale per il socket passivo riservato al canale Dati. Per il canale Comandi abbiamo invece utilizzato la configurazione predefinita che non permette connessioni né riconnessioni dall'esterno. Per quanto riguarda le credenziali e lo stato della sessione utilizziamo delle variabili definite a priori e valorizzate all'interno del gestore d'avvio. Successivamente lo stato della sessione verrà incrementato a seconda delle risposte ricevute dal server fino al termine del trasferimento che comporterà il ritorno al valore iniziale. Nel box del Listato 3 si può vedere il codice relativo all'avvio. La connessione verso il server FTP viene aperta attraverso il metodo *connect* richiamato sull'oggetto socket. Il primo trasferimento avviene subito dopo la prima interrogazione della centralina senza attendere il ritardo valorizzato nella *variabile rit*. In questo modo è possibile effettuare un test di comunicazione in ➤

LISTATO 4

```
sub on_sock_data_arrival
  risp = left(sock.getdata(255),3)
  if sock.num = 0 then
    select case risp
      case "220":
        if stato = 1 then
          sock.num = 0
          sock.setdata("USER " + utente + chr(13) + chr(10))
          sock.send
          stato = 2
          end if
        case "331":
          if stato = 2 then
            sock.num = 0
            sock.setdata("PASS " + pwd + chr(13) + chr(10))
            sock.send
            stato = 3
            end if
          case "230":
            if stato = 3 then
              sock.num = 0
              sock.setdata("PORT 192,168,0,5,3,244"+ chr(13) + chr(10))
              sock.send
              stato = 4
              end if
            case "200":
              if stato = 4 then
                sock.num = 0
                sock.setdata("STOR meteo.dat"+ chr(13) + chr(10)) '(PARA) FileName
                sock.send
                stato = 5
                end if
              case "150":
                if stato = 5 then
                  sock.num = 1
                  w=sys.timercount
                  do
                    doevents
                    loop until (sock.statesimple = PL_SSTS_EST) OR (sys.timercount-w > 60)
                    if sock.statesimple = PL_SSTS_EST then
                      sock.notifysent(1)
                      stato = 6
                    else
                      sock.num=1
                      sock.close
                      sock.num=0
                      sock.close
                      stato=0
                    end if
                  case "226":
                    if stato=7 then
                      sock.num=0
                      sock.setdata("QUIT"+ chr(13) + chr(10))
                      sock.send
                      stato = 8
```

Ogni volta che il socket 0 (targetIP=serverFTP + TargetPort= 21) riceve dei dati estrae i primi 3 caratteri corrispondenti al codice necessario per controllare lo stato della sessione.

Codice 220: Canale Comandi aperto, invia il nome utente per il login.

Codice 331: invia la password per il login.

Codice 230: Login effettuato correttamente. Invia al serverFTP, il comando PORT indicando la porta d'ascolto 1012.

L'operazione PORT si è conclusa correttamente. Invia il comando STOR con il nome del file che verrà utilizzato per il trasferimento.

Il sistema attende che avvenga la connessione sul socket 1 porta 1012. A questo punto viene stabilito il limite di scaricamento del buffer di trasmissione attivando l'evento on_sock_data_sent.

Nel caso si superi il time-out di 30 secondi i due socket vengono chiusi e si ritorna allo stato iniziale.

Il trasferimento è avvenuto con successo. Invia il comando di chiusura sessione.

(Segue)

LISTATO 4 (Continuazione)

```

        end if
    case "221" :
        if stato=8 then
            sock.num=0
            sock.close
            stato=0
        end if
    case else :
        sys.reboot
    end select
end if
end sub

sub on_sock_data_sent
    if stato = 6 then
        sock.num = 1
        sock.setdata(stor.get(1,addr))
        sock.send
        stato=7
        sock.notifysent(1)
    else
        sock.num = 1
        sock.close
        stato=7
    end if
end sub

```

La sessione è stata chiusa correttamente quindi il sistema torna allo stato iniziale.

La risposta del server non è nell'elenco, ci limitiamo a riavviare tutto.

Il sistema invia il meteo.dat. Il gestore `on_sock_data_sent` viene eseguito due volte: una all'inizio del download ed una alla fine con la chiusura del socket associato al canale Dati.

maniera rapida. Per tutte le sessioni successive sarà necessario attendere almeno l'intervallo stabilito durante l'inizializzazione del sistema. Si faccia attenzione che a tale periodo deve essere aggiunto il ritardo derivante dall'interrogazione della WS-2300 che è variabile. È la centralina a stabilire quando dare attenzione alla porta seriale, il DS-202 può soltanto attendere e reinviare la propria richiesta. Dai test effettuati il sistema si presta per un campionamento dei parametri meteo che preveda un controllo ogni 3-4 minuti che riteniamo più che sufficiente.

La versione che distribuiamo ha un *rit* valorizzato a 180, corrispondente ad un ritardo di un minuto e mezzo. Una volta effettuata la connessione con il server il sistema gestisce tutta la sessione attraverso due eventi `on_sock_data_arrival` e `on_sock_data_sent`. Il primo viene eseguito nel momento in cui il DS-

202 riceve dei dati su uno qualsiasi dei socket mentre il secondo è relativo al momento in cui il buffer di trasmissione contiene un numero di byte inferiore al livello stabilito attraverso il metodo `notifysent()`.

Nel momento in cui arrivano dei dati sul socket 0 il gestore provvede ad estrarre i primi tre caratteri ricevuti in risposta dal server. Naturalmente il codice non prevede tutte le possibili condizioni ma si limita ad utilizzare quelle corrispondenti ad una sessione corretta. Se utilizzate una password errata il server risponderà con un codice 530. Non abbiamo sviluppato il codice di gestione dell'errore ma ci limitiamo a riavviare il sistema. Naturalmente potete modificare a piacimento il listato fornito che deve essere visto come una sorta di schema di riferimento didattico più che un sistema completo.

Nel gestore `on_sock_data_arrival` utilizziamo uno statement `select`

case per definire le sequenze da eseguire a seconda dello stato di sessione. Le diverse fasi sono:

FASE1: Il DS-202 entra nello stato 1 dopo aver richiesto la connessione al server FTP. Riceve quindi un codice 220 che lo informa della apertura corretta del canale Comandi. Il dispositivo invia il nome utente e passa allo stato 2;

FASE2: Nello stato 2 il DS-202 riceve il codice 331 relativo alla richiesta password dal server. Il dispositivo invia il contenuto della variabile `pwd` e passa allo stato 3;

FASE3: Nello stato 3 il DS-202 riceve il codice 230 corrispondente ad un login corretto. A questo punto viene trasmesso il comando `PORT` che informa il server del numero di porta su cui rimarrà in ascolto il processo per l'attivazione del canale Dati. Il dispositivo passa allo stato 4;

FASE4: Nello stato 4 il DS-202 riceve il codice 200 corrispondente ad una conclusione corretta del comando `PORT`. Viene trasmesso il comando `STOR` per il file *meteo.dat* che contiene tutte le risposte della centralina registrate durante l'interrogazione. Si passa allo stato 5.

FASE5: Dopo aver ricevuto il codice 150 il sistema attende la connessione del server sulla porta 1012.

Da questo momento viene attivato l'evento `on_sock_data_sent` con un limite sul buffer di trasmissione di 8 byte. Il file è molto piccolo pertanto l'evento viene gestito in un'unica passata. Con file più pesanti si deve prevedere l'invio su più pacchetti. Associando al limite precisato con `notifysent()` un contatore di byte trasmessi, è possibile far eseguire il gestore per tutti i pacchetti necessari. Il sistema entra nello stato 6.

FASE6: Nello stato 6 il DS-202 invia i dati del file *meteo.dat* ed entra nello stato 7.

Il limite di attivazione dell'evento *on_sock_data_sent* permette di far richiamare il gestore nel momento in cui la trasmissione è terminata.

FASE7: Quando la trasmissione del file è terminata viene chiuso il socket relativo al canale Dati e si attende il codice 226.

L'elaborazione è conclusa pertanto il DS-202 invia il comando QUIT con il quale si chiude la sessione entrando nello stato 8.

FASE8: Nello stato 8 il DS-202 riceve il codice 221 corrispondente alla corretta chiusura dell'intera sessione di comunicazione. Il sistema torna nello stato 0 iniziale.

Nel momento in cui la macchina a stati finiti si ritrova nello stato iniziale si avvia il ciclo di attesa. Attraverso il *sys.timercount* si controlla lo scorrere del tempo confrontandolo con il ritardo predefinito. Nel medesimo istante in cui si supera il limite, il DS-202 torna ad interrogare la centralina ripetendo il ciclo complessivo. Vediamo nel Listato 4 come si presenta il sorgente relativo al controllo della sessione FTP.

Al termine della sessione FTP il sistema ritorna allo stato iniziale 0 passando l'esecuzione al ciclo di attesa. Si tratta di un *while* molto semplice. Non si fa altro che richiamare la *sys.timercount* fino al raggiungimento del limite stabilito attraverso la variabile *rit*. Al termine il sistema inizia nuovamente l'esecuzione tornando ad interrogare la centralina WS-2300 (Listato 5).

Messa in funzione

Per quanto riguarda il caricamento

dell'applicativo sul DS-202 vi rimandiamo al box pubblicato sul precedente numero nel quale spieghiamo come inserire il TIOS sul dispositivo commercializzato da Tibbo. La cosa più importante da fare è la configurazione dell'interfaccia di rete e delle credenziali da utilizzare nella sessione FTP. In generale, tutte le variabili di base personalizzabili sono state contraddistinte all'interno del listato dall'etichetta (PARA). Naturalmente è possibile agire anche in maniera più radicale visto che il sistema è completamente aperto e manipolabile attraverso il TIDE. Vi consigliamo di effettuare un test di connessione al server FTP che state utilizzando per controllare di aver configurato correttamente il profilo utente relativo. Basta utilizzare la shell di Windows digitando il comando *ftp nome-server* come si vede in Figura 2. Fate attenzione ad assegnare una directory di lavoro con relative autorizzazioni all'utente creato per stabilire il punto esatto a cui punterà lo script che si occuperà della generazione delle pagine Web.

Il codice inserito nel DS-202 infatti non prevede il comando CWD ma trasferisce il file direttamente in radice. Durante l'esecuzione il sistema utilizza i due led rosso e verde presenti sulla cover superiore del DS-202 per segnalare all'utente lo stato della connessione.

- **LED ROSSO ACCESO:** il DS-202 sta interrogando la stazione meteo.
- **LED VERDE ACCESO:** il DS-202 sta comunicando con il server FTP.
- **LED VERDE E ROSSO ACCESI:** il DS-202 è in attesa tra una sessione e l'altra.

Per ragioni diagnostiche e di aggiornamento è prevista la possibilità di bloccare l'esecuzione del codice mantenendo premuto il tasto di setup che si trova a lato della porta Ethernet. Il sistema gestisce la cosa attraverso l'evento "on_but-

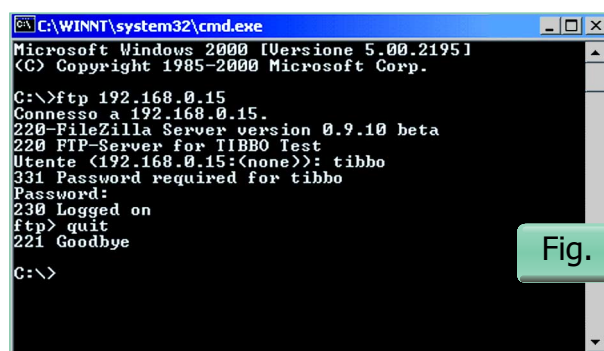


Fig. 2

ton_pressed". A quel punto verrà fatto lampeggiare rapidamente il led verde permettendo le operazioni necessarie.

Conclusioni

Siamo giunti al termine della descrizione di questo progetto che sicuramente potrà darvi una mano per la pubblicazione in rete dei dati meteo della vostra centralina WS-2300. Naturalmente il listato da noi messo a punto può essere modificato a piacere, sperimentando altre soluzioni. Il core di questo nostro sviluppo è stato pubblicato anche come Demo Project sul sito della Tibbo (http://www.tibbo.com/taiko_res.php) che raccoglie anche altri interessanti esempi. Altre informazioni sull'interfacciamento con la WS-2300 sono disponibili su: www.lavrsen.dk/twiki/bin/view/Open2300/WebHome, un twiki-site molto utile.

LISTATO 5

```
w=sys.timercount
pat.play("B",PL_PAT_CANINT)
while sys.timercount-w < rit
doevents
wend
goto inizio
```

Durante il ciclo di attesa si richiama la funzione *doevents* che permette di evitare il blocco della coda di eventi.

le prime fiere di elettronica informatica e radioapparati con solo aziende di qualità garantite da Compendio Fiere

7° edizione - Cerea (VR)

13 - 14 gennaio 2007

25.000 visitatori (nell'edizione invernale) 10.000 MQ coperti
orario 9 - 18 ridotto 5 € intero 7 €



3° edizione - Busto Arsizio (VA)

27 - 28 gennaio 2007

11.000 visitatori (nella seconda edizione) 20.000 MQ coperti
orario 9 - 18 ridotto 5 € intero 7 €

4° edizione - Busto Arsizio (VA)

26 - 27 maggio 2007

11.000 visitatori (nella seconda edizione) 20.000 MQ coperti
orario 9 - 18 ridotto 5 € intero 7 €



8° edizione - Cerea (VR)

25 - 26 agosto 2007

20.000 visitatori (nell'edizione estiva) 10.000 MQ coperti
orario 9 - 19 ridotto 5 € intero 7 €

3° edizione - Reggio Emilia

08 - 09 settembre 2007

12.000 visitatori (nella prima edizione) 18.000 MQ coperti
orario 9 - 18 ridotto 5 € intero 7 €



5° edizione - Busto Arsizio (VA)

22 - 23 settembre 2007

11.000 visitatori (nella seconda edizione) 20.000 MQ coperti
orario 9 - 18 ridotto 5 € intero 7 €

3° edizione - Udine

03 - 04 novembre 2007

10.000 visitatori (nella prima edizione) 11.000 MQ coperti
orario 9 - 18 ridotto 5 € intero 7 €



**www.compendiofiere.it
info 337676719**

Interfaccia USB per Personal Computer, il software

di Arsenio Spadoni

Rendiamo operativa l'interfaccia USB per PC presentata nel numero precedente utilizzando il software di gestione per PC da noi messo a punto. Analizziamo anche le caratteristiche della DLL da utilizzare per lo sviluppo di applicativi personalizzati in Delphi, Visual Basic e Borland C++.

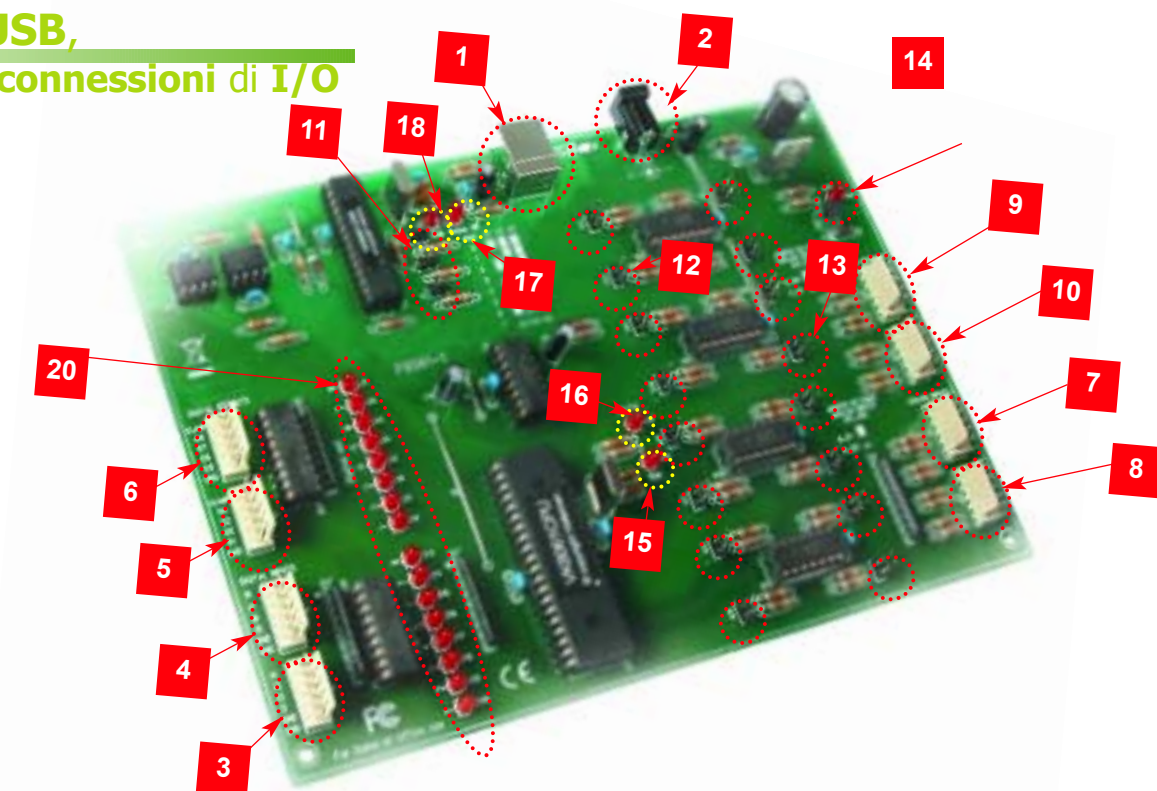


Nel numero precedente abbiamo presentato la struttura hardware dell'interfaccia USB con 33 I/O: un set completo di lavoro adatto per applicazioni professionali e didattiche espandibile modularmente. L'interfaccia K8061 dispone di 8 ingressi e 8 uscite digitali; 8 ingressi ed 8 uscite analogiche; 1 uscita PWM. Qualora il numero dei canali vi sembrasse insufficiente, non disperate, basta connettere altre interfacce (massimo 8) alla porta USB del personal computer ed il gioco è fatto; naturalmente, per evitare un ingorgo di dati, quando si utilizza con lo stesso PC più di una

scheda è necessario impostare per ciascuna interfaccia un indirizzo differente posizionando i jumper A1, A2 ed A3 come riportato nell'apposita tabella. L'indirizzo va impostato prima di connettere la scheda al PC. Ricordiamo che, rispetto al precedente progetto USB presentato sui numeri 90 e 91 della rivista, l'interfaccia proposta in queste pagine è optoisolata e pertanto il suo utilizzo non può in alcun modo danneggiare l'hardware del Personal Computer.

L'interesse suscitato dal progetto è stato notevole e sono arrivate, al nostro servizio tecnico, numerose tele- ➤

L'interfaccia USB, impostazioni e connessioni di I/O



1	Connettore USB	Connettore di collegamento tra scheda K8061 e la porta USB del PC
2	Alimentazione 12VDC	Connettore di alimentazione. Collegare un adattatore di rete con tensione di 12V non regolato / 300mA.
3	Ingressi digitali (1, 2, 3, 4)	Otto ingressi digitali: per essere attivati, devono essere portati a livello "BASSO" (collegare al GND).
4	Ingressi digitali (5, 6, 7, 8)	
5	Uscite digitali (1, 2, 3, 4)	Uscite sono di tipo open collector. Quando sono attive i transistor presenti in IC4 si portano in conduzione collegando a massa l'uscita interessata. Il carico che si vuole collegare (LED, relé, ecc.) deve essere alimentato da una fonte esterna. Collegare il morsetto "CLAMP" al polo positivo (+) della fonte di alimentazione esterna così da proteggere i transistor.
6	Uscite digitali (5, 6, 7, 8)	
7	Ingressi analogici (1, 2, 3, 4)	Ingressi analogici ai quali va applicata una tensione continua di valore compreso tra 0 e 5V o tra 0 e 10V a seconda di come sono stati impostati i jumper AD1 + AD8 (vedi n° 12). ATTENZIONE: l'applicazione di una tensione con valore superiore a 5 o 10 V agli ingressi A/D, potrebbe danneggiare la scheda (IC10/11)!
8	Ingressi analogici (5, 6, 7, 8)	
9	Uscite analogiche (1, 2, 3, 4)	La tensione presente su queste uscite, può essere impostata tramite software. Essa può assumere un valore compreso tra 0 e 5V o tra 0 e 10V a seconda di come sono posizionati i jumper DA1 + DA8. Sul connettore SK9, è anche disponibile un'uscita PWM (pin 2) di tipo open collector, la cui larghezza d'impulso può essere modificata via software.
10	Uscite analogiche (5, 6, 7, 8)	
11	Jumper per impostazione indirizzo	Mediante i jumper A1, A2 e A3, è possibile attribuire un indirizzo univoco per ciascuna scheda connessa. È possibile collegare fino a 8 schede; se ne viene utilizzata solamente una, impostare l'indirizzo "0".
12	Jumper per impostazione ingressi A/D	Mediante i jumper AD1-AD8, è possibile impostare il massimo valore di tensione del relativo ingresso A/D; da 0 a 5V (jumper aperto) o da 0 a 10V (jumper chiuso).
13	Jumper per impostazione uscite D/A	Mediante i jumper DA1-DA8, è possibile impostare la massima tensione ottenibile sulla corrispondente uscita D/A; da 0 a 5V (jumper aperto) o 0 a 10V (jumper chiuso)
14	Led di controllo PWM	Questo LED si illumina quando l'uscita PWM viene attivata. La luminosità del LED è proporzionale al rapporto impulso/pausa.
15	Led CPU "run mode"	Si illumina quando il microprocessore IC6 montato sulla scheda funziona correttamente.
16	Led CPU RX/TX	Questo LED si illumina quando vi è uno scambio di dati tra il micro IC6 ed il controller USB IC3. Se tale LED non si illumina quando la scheda è alimentata, verificare che il controller USB (IC3) o gli optoisolatori (IC1 & 2) non siano guasti.
17	Led POWER ON	Si illumina quando è presente la tensione di alimentazione (5 V) del controller USB. Attenzione: il K8061 è alimentato tramite la porta USB che non fornisce tensione di alimentazione alle sezioni di input/output della scheda.
18	Led USB	Lampeggia durante la connessione USB e si illumina a collegamento riuscito tra il PC e la scheda K8061.
19	Led attivazione ingressi digitali	Si spengono quando il corrispondente ingresso passa ad un livello "BASSO" (collegamento dell'ingresso con GND) mediante un contatto esterno o tramite un'uscita open collector.
20	Led attivazione uscite digitali	Si illuminano quando la corrispondente uscita è attiva, ovvero quando il relativo transistor conduce verso massa (uscita open collector).

Tabella 1 - Impostazione dei Jumper A1, A2, A3

A1	A2	A3	INDIRIZZO SCHEDA
ON	ON	ON	0
OFF	ON	ON	1
ON	OFF	ON	2
OFF	OFF	ON	3
ON	ON	OFF	4
OFF	ON	OFF	5
ON	OFF	OFF	6
OFF	OFF	OFF	7

fonate di lettori che chiedevano informazioni in merito alle procedure software necessarie per usare la K8061. È proprio di questo che ci occupiamo questo mese descrivendo innanzitutto i due programmi forniti a corredo della scheda ma soprattutto presentando le routine di comunicazione contenute nella DLL (Dynamic Link Library) denominata K8061.DLL utilizzata nei nostri programmi e indispensabile per scrivere software personalizzati.

Richiamando le funzioni e le procedure esportate dalla DLL, si potranno scrivere applicazioni personalizzate Windows (98SE, 2000, Me, XP) in Delphi, Visual Basic, C++ Builder o con qualsiasi altro strumento di sviluppo di applicazioni Windows a 32 bit che supporti chiamate ad una DLL. Presentiamo anche alcuni esempi su come costruire i propri programmi applicativi; negli esempi, scritti in Delphi, Visual Basic e C++ Builder, vi sono dichiarazioni complete sulle funzioni e procedure DLL.

Gli esempi riportati nella descrizione della DLL sono invece scritti unicamente per Delphi.

Panoramica delle procedure e delle funzioni della K8061.DLL

Procedure generali

OpenDevice

Apri la comunicazione con la scheda K8061

CloseDevices

Chiude la connessione a tutti i dispositivi della K8061

PowerGood

Controlla il corretto funzionamento dell'integrato IC6
Connected

Controlla che la connessione USB risulti operativa

ReadVersion(CardAddress, Buffer)

Identifica la versione software di IC3 ed IC6

Procedure del convertitore Analogico-Digitale

ReadAnalogChannel(CardAddress, Channelno)

Legge lo stato di un canale di input analogico

Parametri: indirizzo di scheda, numero del canale

ReadAllAnalog(CardAddress, Buffer)

Legge lo stato di tutti i canali di input analogici

Parametri: indirizzo di scheda, buffer dati

Procedure di conversione da Analogico a Digitale

OutputAnalogChannel(CardAddress, Channel, Data)

Configura un canale analogico

Parametri: indirizzo di scheda, canale e dati

OutputAllAnalog(CardAddress, Buffer)

Configura tutti i canali analogici

Parametri: indirizzo di scheda, dati

ClearAnalogChannel

Configura al valore minimo l'uscita analogica indicata

Parametri: indirizzo di scheda e canale

ClearAllAnalog

Configura al valore minimo tutte le uscite analogiche

Parametri: indirizzo di scheda

SetAnalogChannel

Configura al valore massimo l'uscita analogica indicata

Parametri: indirizzo di scheda e canale

SetAllAnalog

Configura al valore massimo tutte le uscite analogiche

Parametri: indirizzo di scheda

Procedure per output PWM

OutputPWM(CardAddress, Data)

Imposta il canale di uscita PWM in funzione dei dati

Parametri: indirizzo di scheda, dati

Procedure uscite digitali

OutputAllDigital(CardAddress, Data)

Imposta tutte le uscite digitali in funzione dei dati

Parametri: indirizzo di scheda, dati

ClearDigitalChannel(CardAddress, Channel)

Azzera il canale di uscita digitale selezionato

Parametri: indirizzo di scheda, canale

ClearAllDigital(CardAddress)

Azzera tutti i canali di uscita digitali

Parametri: indirizzo di scheda

SetDigitalChannel(CardAddress, Channel)

Imposta il canale digitale selezionato

Parametri: indirizzo di scheda, canale

SetAllDigital(CardAddress)

Imposta tutti i canali digitali

Parametri: indirizzo di scheda

Input digitali, procedure e funzioni

ReadDigitalChannel(CardAddress, Channel)

Legge lo stato del canale digitale di ingresso selezionato

Parametri: indirizzo di scheda, canale

ReadAllDigital(CardAddress, Buffer)

Legge lo stato di tutti i canali digitali di ingresso

Parametri: indirizzo di scheda

Un esempio in Delphi

In questo esempio scritto in Delphi illustriamo come usare le procedure e le funzioni della K8061.DLL ed in particolare come usare due delle più importanti chiamate alla DLL: OpenDevice e CloseDevices.

```
unit K8061dmo;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, ExtCtrls, ComCtrls, Math, Buttons;

type
  TForm1 = class(TForm)
    GroupBox1: TGroupBox;
    A2: TCheckBox;
    A1: TCheckBox;
    A3: TCheckBox;
    Timer1: TTimer;
    Button3: TButton;
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure Button3Click(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

{$R *.DFM}
function OpenDevice: Longint; stdcall; external 'K8061.dll';
procedure CloseDevices; stdcall; external 'K8061.dll';
function ReadAnalogChannel(CardAddress: Longint; Channel: Longint): Longint; stdcall; external
'K8061.dll';
procedure ReadAllAnalog(CardAddress: Longint; Buffer: Pointer); stdcall; external 'K8061.dll';
procedure OutputAnalogChannel(CardAddress: Longint; Channel: Longint; Data: Longint); stdcall;
external 'K8061.dll';
procedure OutputAllAnalog(CardAddress: Longint; Buffer: Pointer); stdcall; external
'K8061.dll';
procedure ClearAnalogChannel(CardAddress: Longint; Channel: Longint); stdcall; external
'K8061.dll';
procedure ClearAllAnalog(CardAddress: Longint); stdcall; external 'K8061.dll';
procedure SetAnalogChannel(CardAddress: Longint; Channel: Longint); stdcall; external
'K8061.dll'; procedure SetAllAnalog(CardAddress: Longint); stdcall; external 'K8061.dll';
procedure OutputAllDigital(CardAddress: Longint; Data: Longint); stdcall; external
'K8061.dll';
procedure ClearDigitalChannel(CardAddress: Longint; Channel: Longint); stdcall; external
'K8061.dll';
procedure ClearAllDigital(CardAddress: Longint); stdcall; external 'K8061.dll';
procedure SetDigitalChannel(CardAddress: Longint; Channel: Longint); stdcall; external
'K8061.dll';
procedure SetAllDigital(CardAddress: Longint); stdcall; external 'K8061.dll';
function ReadDigitalChannel(CardAddress: Longint; Channel: Longint): Boolean; stdcall;
external 'K8061.dll';
function ReadAllDigital(CardAddress: Longint): Longint; stdcall; external 'K8061.dll';
procedure OutputPWM(CardAddress: Longint; Data: Longint); stdcall; external 'K8061.dll';
function PowerGood(CardAddress: Longint): Boolean; stdcall; external 'K8061.dll';
function Connected(CardAddress: Longint): Boolean; stdcall; external 'K8061.dll';
procedure ReadVersion(CardAddress: Longint; Buffer: Pointer); stdcall; external 'K8061.dll';

procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  CloseDevices;
end;

procedure TForm1.Button3Click(Sender: TObject);
var h: Longint;
begin
  h:= OpenDevice;
  case h of
    0..7: begin
      Label12.Caption:='Card '+ inttostr(h)+' connected';
      RadioGroup3.Items[h]:=inttostr(h);
      RadioGroup3.Enabled:=true;
      RadioGroup3.ItemIndex:=h;
      Timer1.Enabled:=true;
      CardAddress:=h;
    end;
    -1: Label12.Caption:='All cards connected.';
    -2: Label12.Caption:='Card not found.';
  end;
end;
end;
```

Procedure e funzioni della K8061.DLL

OpenDevice

Attiva il collegamento con la scheda e carica i driver necessari a comunicare con la porta USB. La procedura deve essere eseguita prima di ogni tentativo di comunicazione con la scheda. Se al computer sono connesse più schede la procedura deve essere ripetuta per ciascuna di loro fin tanto che tutte risultino operative. Il risultato è il seguente: in caso positivo, in presenza di una sola scheda, ne restituisce l'indirizzo. Con più schede, la risposta è -1 quando tutte le schede sono connesse e -2 quando non trova la scheda.

Sintassi:

FUNCTION OpenDevice: Longint;

Il parametro da utilizzare è Longint, esso corrisponde all'indirizzo della scheda e può pertanto essere compreso fra 0 e 7; l'indirizzo corrisponde ovviamente all'impostazione dei dip sulla scheda.

Esempio:

```
var CardAddress: longint;  
BEGIN
```

```
CardAddress:=OpenDevice;
```

```
// Apre il link verso la scheda e ne restituisce  
l'indirizzo assegnato
```

```
END;
```

CloseDevices

Il comando è l'ordine necessario per chiudere una sessione di lavoro: disattiva le routine di comunicazione della scheda K8061 e quelle dei driver necessari per la connessione USB.

Sintassi:

PROCEDURE CloseDevices;

Esempio:

```
BEGIN
```

```
CloseDevices; // chiude la comunicazione con i  
dispositivi K8061
```

```
END;
```

ReadAnalogChannel

Il convertitore analogico digitale legge la tensione d'ingresso del corrispondente canale. La tensione presente viene convertita in un valore compreso fra 0 e 1023.

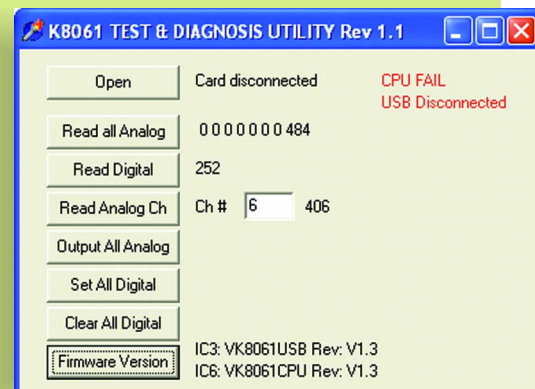
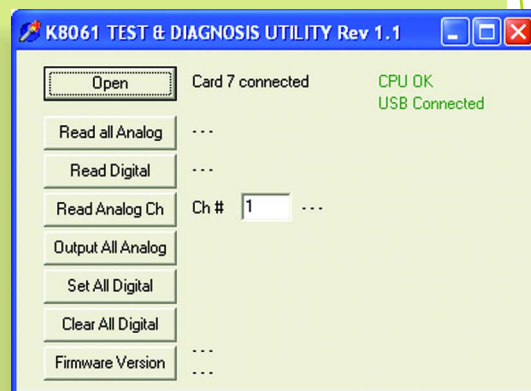
Sintassi:

Il programma di test

Questo semplice programma consente di verificare il corretto funzionamento della scheda una volta ultimato il montaggio. Con una veste grafica semplice ed essenziale, questo software permette di controllare tutte le funzionalità dell'interfaccia USB identificando facilmente eventuali anomalie di funzionamento. In questo riquadro riportiamo alcune schermate che evidenziano come opera il programma. I "bottoni" presenti sulla sinistra consentono, nell'ordine, di aprire il collegamento con la scheda verificando l'indirizzo della

stessa nonché il buon funzionamento del microcontrollore e del link dati. Il secondo bottone permette di leggere tutti gli ingressi analogici: il risultato appare immediatamente sulla destra. Il terzo pulsante consente di leggere lo stato di tutti gli ingressi digitali mentre il quarto permette di leggere singolarmente lo stato degli ingressi analogici. Il canale relativo va indicato nell'apposita fine-

stra. I successivi tre pulsanti consentono di settare (ovvero attivare) o resettare lo stato di tutti gli ingressi e le uscite digitali nonché di tutte le uscite analogiche. L'ultimo pulsante in basso permette di verificare la versione del firmware con il quale sono stati programmati i due microcontrollori montati sulla scheda. Ricordiamo che la verifica delle varie funzioni può essere effettuata solamente dopo che è stato instaurato il collegamento tra il PC e la scheda mediante l'attivazione del pulsante "Open".



Un esempio in Visual Basic

Riportiamo un listato che permette di eseguire i medesimi compiti di quello che abbiamo proposto in Delphi. In questo caso il linguaggio è il Visual Basic della Microsoft. Perchè possa funzionare correttamente è fondamentale copiare il file K8061.DLL nella cartella System32 che si trova all'interno della directory Windows.

```
Option Explicit
Dim DoNothing As Boolean
Dim n As Integer
Dim CardAddress As Long

Private Declare Function OpenDevice Lib "k8061.dll" () As Long
Private Declare Sub CloseDevices Lib "k8061.dll" ()
Private Declare Function ReadAnalogChannel Lib "k8061.dll" (ByVal CardAddress As Long, ByVal Channel As Long) As Long
Private Declare Function PowerGood Lib "k8061.dll" (ByVal CardAddress As Long) As Boolean
Private Declare Function Connected Lib "k8061.dll" (ByVal CardAddress As Long) As Boolean
Private Declare Sub ReadVersion Lib "k8061.dll" (ByVal CardAddress As Long, Buffer As Long)
Private Declare Sub ReadAllAnalog Lib "k8061.dll" (ByVal CardAddress As Long, Buffer As Long)
Private Declare Sub OutputAnalogChannel Lib "k8061.dll" (ByVal CardAddress As Long, ByVal Channel As Long, ByVal Data As Long)
Private Declare Sub OutputAllAnalog Lib "k8061.dll" (ByVal CardAddress As Long, Buffer As Long)
Private Declare Sub ClearAnalogChannel Lib "k8061.dll" (ByVal CardAddress As Long, ByVal Channel As Long)
Private Declare Sub SetAllAnalog Lib "k8061.dll" (ByVal CardAddress As Long)
Private Declare Sub ClearAllAnalog Lib "k8061.dll" (ByVal CardAddress As Long)
Private Declare Sub SetAnalogChannel Lib "k8061.dll" (ByVal CardAddress As Long, ByVal Channel As Long)
Private Declare Sub OutputAllDigital Lib "k8061.dll" (ByVal CardAddress As Long, ByVal Data As Long)
Private Declare Sub ClearDigitalChannel Lib "k8061.dll" (ByVal CardAddress As Long, ByVal Channel As Long)
Private Declare Sub ClearAllDigital Lib "k8061.dll" (ByVal CardAddress As Long)
Private Declare Sub SetDigitalChannel Lib "k8061.dll" (ByVal CardAddress As Long, ByVal Channel As Long)
Private Declare Sub SetAllDigital Lib "k8061.dll" (ByVal CardAddress As Long)
Private Declare Function ReadDigitalChannel Lib "k8061.dll" (ByVal CardAddress As Long, ByVal Channel As Long) As Boolean
Private Declare Function ReadAllDigital Lib "k8061.dll" (ByVal CardAddress As Long) As Long
Private Declare Sub OutputPWM Lib "k8061.dll" (ByVal CardAddress As Long, ByVal Data As Long)

Private Sub Connect_Click()
    Dim h As Long
    h = OpenDevice
    Select Case h
        Case 0, 1, 2, 3, 4, 5, 6, 7
            Label1.Caption = "Card " + Str(h) + " connected"
    Option1(h).Enabled = True
        Option1(h).Value = True
        Timer1.Enabled = True
        Case -2
            Label1.Caption = "Card not found"
        Case -1
            Label1.Caption = "All cards opened"
    End Select
End Sub

Private Sub Form_Terminate()
    CloseDevices
End Sub
```

FUNCTION *ReadAnalogChannel* (*CardAddress: Longint; Channel: Longint*): *Longint*;

Parametri:

CardAddress: indirizzo della scheda in uso.

Channel: canale AD da 1 a 8 che deve essere letto.

Esempio:

var data: longint;

BEGIN

data := ReadAnalogChannel(0, 1);

// Nella variabile data viene inserito il valore del convertitore AD numero 1

END;

ReadAllAnalog

Lo stato di tutti i convertitori analogico/digitali è leggibile in un insieme (array) di interi lunghi (integer long) Sintassi:

PROCEDURE *ReadAllAnalog*(*CardAddress: Longint; Buffer: Pointer*);

Parametri:

CardAddress: indirizzo della scheda in uso.

Buffer: permette di stabilire l'insieme di riferimento dell'intero lungo.

Esempio:

var Buffer: Array[0..7] of Longint;

begin

```
ReadAllAnalog(0, @Buffer[0]);  
// Legge tutti i dati analogici provenienti dalla  
// scheda corrispondenti all'indirizzo.
```

END;

OutputAnalogChannel

Consente di modificare il valore della tensione d'uscita del convertitore D/A specificato in funzione dei nuovi dati. Il valore 0 corrisponde alla minima tensione in uscita (0 Volt), mentre 255 corrisponde alla massima tensione d'uscita (+5 oppure +10 Volt in funzione della configurazione dei jumper).

Sintassi:

```
PROCEDURE OutputAnalogChannel(CardAddress:  
Longint; Channel: Longint; Data: Longint);
```

Parametri:

CardAddress: indirizzo della scheda in uso.

Channel: valore compreso fra 1 e 8 che corrisponde al numero del canale selezionato.

Data: valore compreso fra 0 e 255 che determina il livello del segnale di uscita del convertitore.

Esempio:

BEGIN

```
OutputAnalogChannel (0, 1, 127);  
// il canale D/A 1 della scheda 0 è configurato a 2,5 V  
// oppure a 5 Volt in funzione della posizione del  
// jumper del D/A
```

END;

OutputAllAnalog

Tutte le tensioni delle uscite degli 8 convertitori D/A vengono modificate contemporaneamente. Come nel caso precedente il valore 0 corrisponde al minimo della tensione di uscita, mentre 255 al massimo. Il limite superiore raggiungibile può essere di 5 o 10 Volt in funzione dell'impostazione dei jumper.

Sintassi:

```
PROCEDURE OutputAllAnalog (CardAddress:  
Longint; Buffer: Pointer);
```

Parametri:

CardAddress: indirizzo della scheda in uso.

Buffer: permette di stabilire l'insieme di riferimento dell'intero lungo che contiene i dati.

Esempio:

```
var Buffer: Array[0..7] of Longint;
```

```
i: integer;
```

begin

```
for i:=0 to 7 do Buffer[i]:=10*i;
```

```
OutputAllAnalog(0, @Buffer[0]);
```

```
// grazie al ciclo di for/next la scheda agisce su  
// tutte le uscite analogiche.
```

END;

ClearAnalogChannel

Consente di azzerare l'uscita del canale D/A selezionato portando il valore di uscita a 0 Volt.

Sintassi:

```
PROCEDURE ClearAnalogChannel (CardAddress:  
Longint; Channel: Longint);
```

Parametri:

CardAddress: indirizzo della scheda in uso.

Channel: valore compreso fra 1 e 8 che corrisponde al numero del canale i cui dati debbono essere azzerati.

Esempio:

BEGIN

```
ClearAnalogChannel (0, 1);
```

```
// il valore della tensione d'uscita del convertitore  
// D/A del canale 1 della scheda 0 viene azzerato.
```

END;

ClearAllAnalog

Tutti i canali dei convertitori D/A(8) vengono impostati per la minima tensione d'uscita (0 Volt).

Sintassi:

```
PROCEDURE ClearAllAnalog (CardAddress: Longint);
```

Parametri:

CardAddress: indirizzo della scheda in uso.

Esempio:

BEGIN

```
ClearAllAnalog(0);
```

```
// tutti i canali D/A della scheda 0 sono posti a 0 Volt.
```

END;

SetAnalogChannel

L'uscita del canale A/D selezionato viene impostata per la massima tensione d'uscita (+5 oppure + 10 Volt in funzione dell'impostazione dei jumper).

Sintassi:

```
PROCEDURE SetAnalogChannel (CardAddress:  
Longint; Channel: Longint);
```

Parametri:

CardAddress: indirizzo della scheda in uso.

Esempio:

BEGIN

```
SetAnalogChannel(0, 1);
```

```
// il canale 1 del convertitore D/A della scheda 0  
// è posto a +5V
```

```
// oppure a +10V in funzione del posizionamento  
// dei jumper
```

END;

SetAllAnalog

Tutte le uscite dei canali A/D vengono impostate per la massima tensione d'uscita (+5 oppure + 10 Volt) in funzione dell'impostazione dei jumper).

Sintassi:



PROCEDURE SetAllAnalog (CardAddress: Longint);

Parametri:

CardAddress: indirizzo della scheda in uso.

Esempio:

BEGIN

SetAllAnalog(0);

*// i canali da 1 a 8 dei convertitori DA della scheda
0 sono posti a +5 Volt*

*// oppure a +10 Volt in funzione della configurazione
dei jumper*

END;

OutputAllDigital

Con un solo comando vengono settate le uscite digitali che assumono lo stato alto o basso (tutte alte, tutte basse o alcune a 1 e altre a 0). Il livello 1 significa che l'uscita del microcontrollore IC1 è attiva, il livello 0 che l'uscita è a massa.

Sintassi:

PROCEDURE OutputAllDigital (CardAddress: Longint; Data: Longint);

Parametri:

CardAddress: indirizzo della scheda in uso.

Data: l'intervallo dei dati inviabili in uscita spazia fra 0 e 255.

Esempio:

BEGIN

OutputAllDigital(0, 7);

*// le uscite dei canali 1...3 della scheda 0 sono attive
ed i canali 4...8 della stessa scheda sono off in
quanto 7 in binario vale 00000111 ed il bit più
significativo, quello a sinistra corrisponde
all'uscita 8 ed il meno significativo, quello di destra,
all'uscita 1*

END;

ClearDigitalChannel

Il canale selezionato dell'uscita digitale della scheda attiva viene posto a 0

Sintassi:

PROCEDURE ClearDigitalChannel (CardAddress: Longint; Channel: Longint);

Parametri:

CardAddress: indirizzo della scheda in uso.

Channel: i valori compresi fra 1 e 8 corrispondono all'uscita su cui si desidera intervenire.

Esempio:

BEGIN

ClearIOchannel(0, 4);

// l'uscita digitale 4 della scheda 0 è off

END;

ClearAllDigital

Tutti i canali delle uscite digitali vengono azzerati.

Sintassi:

PROCEDURE ClearAllDigital (CardAddress: Longint);

Parametri:

CardAddress: indirizzo della scheda in uso.

Esempio:

BEGIN

ClearAllDigital(0);

// tutti i canali, dal numero 1 al numero 8, sono off

END;

SetDigitalChannel

Il canale digitale selezionato della scheda in uso viene reso attivo.

Sintassi:

PROCEDURE SetDigitalChannel (CardAddress: Longint; Channel: Longint);

Parametri:

CardAddress: indirizzo della scheda in uso.

Channel: il canale selezionato, compreso fra 1 e 8 viene attivato.

Esempio:

BEGIN

SetDigitalChannel(0, 1);

*// L'uscita digitale del canale 1 della scheda con
indirizzo 0 è ON*

// Digital output channel 3 of card address 0 is ON

END;

SetAllDigital

Tutte le uscite digitali della scheda selezionata vengono rese attive.

Sintassi:

PROCEDURE SetAllDigital (CardAddress: Longint);

Parametri:

CardAddress: indirizzo della scheda in uso.

Esempio:

BEGIN

SetAllDigital(0);

// tutti i canali della scheda 0 devono essere ON

END;

ReadDigitalChannel

Legge lo stato dell'ingresso digitale selezionato. Il risultato che si ottiene è di tipo Booleano, VERO (1) significa che la tensione d'ingresso al canale digitale selezionato è alta; FALSO (0) indica che il valore di input è basso.

Sintassi:

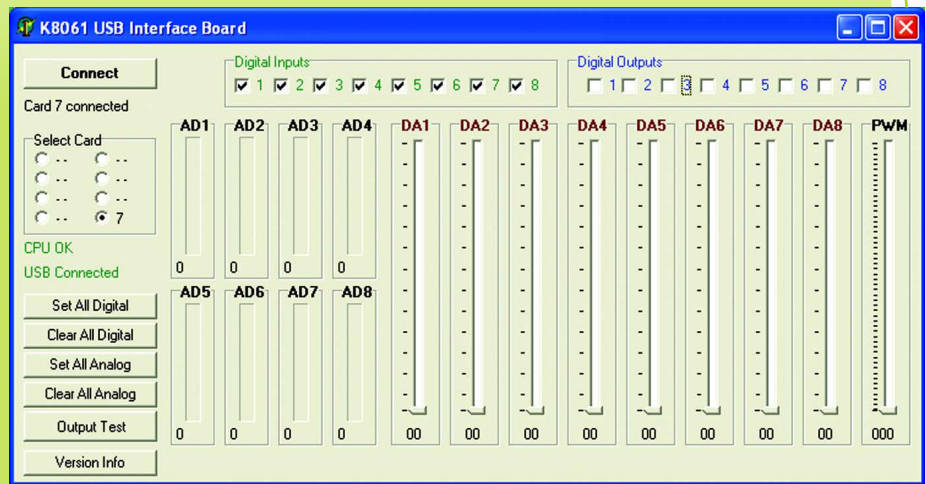
FUNCTION ReadDigitalChannel (CardAddress: Longint; Channel: Longint): Boolean;

Parametri:

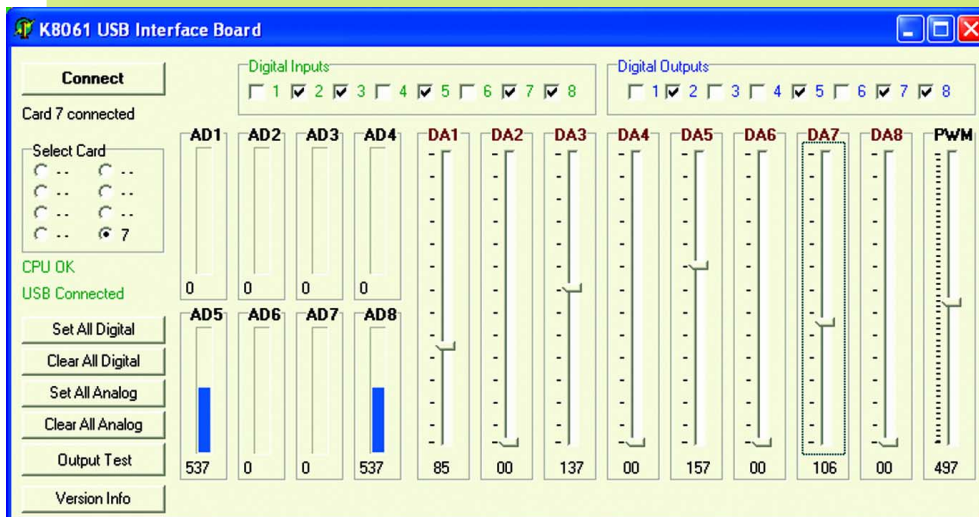
CardAddress: indirizzo della scheda in uso.

Il software di gestione

Quanti non hanno particolare dimestichezza con la programmazione e vogliono semplicemente utilizzare l'interfaccia USB per controllare dispositivi esterni potranno utilizzare il programma da noi messo a disposizione e scaricabile gratuitamente dal sito della rivista. In questo riquadro pubblichiamo due schermate tra le più significative, schermate che evidenziano chiaramente tutte le potenzialità del software. Il programma consente ovviamente di controllare tutte le funzionalità della scheda, gli ingressi, le uscite e di effettuare con un solo click quelle operazioni che controllano contemporaneamente tutti gli ingressi e tutte le uscite. Osservando le immagini potete notare che grazie all'area Select Card si imposta la scheda da testare. Per ogni scheda abbiamo modo di controllare tutte le funzionalità, infatti, in basso a sinistra delle immagini, troviamo i tasti



che corrispondono alle procedure fondamentali per la configurazione dei canali di Input/Output analogici, digitali e del canale PWM. Per ogni canale del dispositivo interveniamo selezionandone l'operatività (applicando il segno di spunta alla relativa check box) e poi possiamo agire sui parametri dei convertitori digitali/analogici, sulle uscite digitali e sul rapporto pausa-impulso del canale dedicato al PWM. Per quanto riguarda gli ingressi analogici abbiamo previsto - oltre al valore numerico - un'indicazione a barra che consente di apprezzarne con immediatezza il livello. Per gli ingressi digitali sono semplicemente presenti dei check box.



Channel: il valore compreso fra 1 e 8 corrisponde al canale da leggere.

Esempio:

```
var status: boolean;
```

```
BEGIN
```

```
status := ReadIOchannel(0, 2);
```

```
// legge l'input del canale 2 della scheda 0
```

```
END;
```

ReadAllDigital

Legge lo stato di tutti gli ingressi digitali della scheda. Il risultato ci viene offerto come intero lungo e gli 8 bit meno significativi corrispondono alla condizione di ingresso di ogni canale; 1=ingresso alto, 0=ingresso basso.

Sintassi:

FUNCTION ReadAllDigital (CardAddress: Longint): Longint;

Parametri:

CardAddress: indirizzo della scheda in uso.

Esempio:

```
var status: longint;
```

```
BEGIN
```

```
status := ReadAllDigital(0);
```

```
// Legge lo stato dei canali di Input digitali della  
scheda con indirizzo 0
```

```
END;
```

OutputPWM

Il duty cycle dell'uscita PWM presenta un valore proporzionale al dato impostato.

Sintassi:

Un esempio in C++

Affrontiamo le medesime problematiche con un terzo esempio d'uso della DLL. In questo caso il linguaggio di programmazione è Borland C++Builder e quello che segue è il listato proposto. Per un corretto funzionamento è necessario includere K8061.lib al progetto. Gli esperti di C++ non avranno certo problema a farlo, ma ci consentano comunque di ricordare che l'operazione si compie, dopo aver selezionato il progetto, accedendo al menu delle opzioni e scegliendo Add To Project, che ci permette di aggiungere all'applicativo creato il file K8061.lib.

```
//Listing K8061.h
#ifdef __cplusplus
extern "C" {
#endif

#define FUNCTION __declspec(dllimport)

FUNCTION long __stdcall OpenDevice();
FUNCTION __stdcall CloseDevices();
FUNCTION long __stdcall ReadAnalogChannel(long CardAddress, long Channel);
FUNCTION __stdcall ReadAllAnalog(long CardAddress, long *Buffer);
FUNCTION __stdcall OutputAnalogChannel(long CardAddress, long Channel, long Data);
FUNCTION __stdcall OutputAllAnalog(long CardAddress, long *Buffer);
FUNCTION __stdcall ClearAnalogChannel(long CardAddress, long Channel);
FUNCTION __stdcall ClearAllAnalog(long CardAddress);
FUNCTION __stdcall SetAnalogChannel(long CardAddress, long Channel);
FUNCTION __stdcall SetAllAnalog(long CardAddress);
FUNCTION __stdcall OutputAllDigital(long CardAddress, long Data);
FUNCTION __stdcall ClearDigitalChannel(long CardAddress, long Channel);
FUNCTION __stdcall ClearAllDigital(long CardAddress);
FUNCTION __stdcall SetDigitalChannel(long CardAddress, long Channel);
FUNCTION __stdcall SetAllDigital(long CardAddress);
FUNCTION bool __stdcall ReadDigitalChannel(long CardAddress, long Channel);
FUNCTION long __stdcall ReadAllDigital(long CardAddress);
FUNCTION __stdcall OutputPWM(long CardAddress, long Data);
FUNCTION bool __stdcall PowerGood(long CardAddress);
FUNCTION bool __stdcall Connected(long CardAddress);
FUNCTION __stdcall ReadVersion(long CardAddress, long *Buffer);

#ifdef __cplusplus
}
#endif
```

```
//Listing Unit1.cpp
//-----
#include <vcl.h>
#pragma hdrstop

#include "Unit1.h"
#include "K8061.h"
//-----
#pragma package (smart init)
#pragma resource "*.dfm"
TForm1 *Form1;
bool DisableOtherFunctionCall = false;
long CardAddr;
int n=8;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Connect1Click(TObject *Sender)
{
    int h = OpenDevice();
    switch (h) {
        case 0:
        case 1:
        case 2:
        case 3:
        case 4:
        case 5:
        case 6:
        case 7:
            Label1->Caption = "Card " + IntToStr(h) + " connected";
            RadioGroup1->Items->Strings[h] = IntToStr(h);
            RadioGroup1->Enabled = true;
            RadioGroup1->ItemIndex = h;
            Timer1->Enabled = true;
            break;
        case -1 :
            Label1->Caption = "All cards connected.";
            break;
        case -2 :
            Label1->Caption = "Card not found";
    }
}
//-----
void __fastcall TForm1::FormClose(TObject *Sender, TCloseAction &Action)
{
    CloseDevices();
}
//-----
```

*PROCEDURE OutputPWM (CardAddress: Longint;
Data: Longint);*

Parametri:

CardAddress: indirizzo della scheda in uso.

Data: All'uscita PWM della scheda può essere assegnato un valore compreso fra 0 e 1023.

Esempio:

BEGIN

OutputPWM (0, 255);

// Il duty cycle dell'uscita PWM è configurato al 25%

END;

PowerGood

Procedura diagnostica che controlla il buon funzionamento dei due micro (IC3 e IC6) verificando anche che il link dati sia attivo.

La risposta che si ottiene è di tipo Booleano: 1 (vero) indica che IC6 (PIC16F871) funziona correttamente e che il data-link fra IC3 ed IC6 è operativo; 0 (falso), significa che IC6 non funziona, oppure che il data-link fra i due integrati non è corretto.

Sintassi:

*FUNCTION PowerGood (CardAddress: Longint):
Boolean;*

Parametri:

CardAddress: indirizzo della scheda in uso.

Esempio:

*if PowerGood(CardAddress) then
begin*

Label19.Caption:='CPU OK';

Label19.Font.Color:=ClGreen;

END

else

begin

Label19.Caption:='CPU FAIL';

Label19.Font.Color:=ClRed;

END;



Connected

Controlla il funzionamento della connessione USB. La risposta che si ottiene è di tipo Booleano: 1 (vero), la connessione USB è operativa e funzionante; 0 (falso), la connessione USB non è operativa oppure il cavo è disconnesso.

Sintassi:

*FUNCTION Connected (CardAddress: Longint):
Boolean;*

Parametri:

CardAddress: indirizzo della scheda in uso.

Esempio:

*if Connected(CardAddress) then
begin*

Label20.Caption:='USB Connected';

Label20.Font.Color:=ClGreen;

END

else

begin

Label20.Caption:='USB Disconnected';

Label20.Font.Color:=ClRed;

Label12.caption:='Card disconnected';

END;

ReadVersion

Procedura che consente di leggere la versione del firmware con la quale sono stati programmati i microcontrol- ➤

Per il

MATERIALE

Il progetto descritto in queste pagine è disponibile in scatola di montaggio (cod. K8061) al prezzo di 98,00 Euro. Il kit comprende tutti i componenti, la basetta forata e serigrafata, i due microcontrollori già programmati, il cavo di connessione USB, le prese di ingresso/uscita ed un CD con due programmi completi nonché la DLL per realizzare programmi personalizzati in vari linguaggi come descritto nel presente articolo. tutti i prezzi sono da intendersi IVA compresa.

**Il materiale va richiesto a: Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 ~ Fax: 0331-778112 ~ <http://www.futuranet.it>**

3ª FIERA DELL' ELETTRONICA
SCANDIANO (RE)
28/29 OTTOBRE 2006

- TELEFONIA
- COMPONENTISTICA
- COMPUTER ■ HI-FI CAR
- RADIANTISMO CB e OM
- VIDEOREGISTRAZIONE
- MERCATINO DELLE PULCI
- RADIOAMATORIALI

Orari: sabato 9 - 18,30 / domenica 9 - 18

INFO: 0522.857136 www.fierascandiano.it con il patrocinio di A.B.I. sez. Scandiano

lori IC3 ed IC6. Per la lettura si posiziona il puntatore sull'array che contiene le informazioni ed estraggono i dati. L'occupazione minima del buffer è di un intero lungo da 200 bytes (valore 50).

Sintassi:

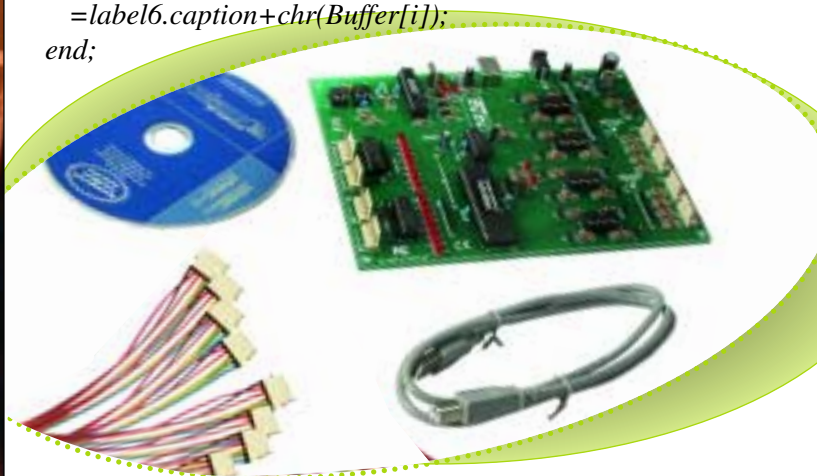
PROCEDURE ReadVersion(CardAddress: Longint; Buffer: Pointer);

Parametri:

CardAddress: identificatore della scheda in uso.

Esempio:

```
var Buffer: array[0..50] of longint;
    i: integer;
begin
    ReadVersion(cardnumber, @Buffer[0]);
    label6.caption:="";
    for i:=0 to 49 do label6.caption:
        =label6.caption+chr(Buffer[i]);
end;
```



Ultimata l'analisi della DLL proponiamo alcuni esempi riguardanti la realizzazione di programmi applicativi. Il primo è scritto in Delphi: in questo esempio troviamo le dichiarazioni delle procedure e le funzioni K8061.DLL e un esempio su come utilizzare le due chiamate alle funzione DLL più importanti: OpenDevice e CloseDevice. Anche nell'esempio in Visual Basic sono presenti le dichiarazioni delle procedure e funzioni K8061.DLL e un esempio su come utilizzare le due chiamate OpenDevice e CloseDevice. In questo caso bisogna verificare che il file K8061.DLL venga copiato nella cartella Windows SYSTEM32. Molto simile è anche l'esempio scritto in C++ Builder con, anche in questo caso, le due chiamate alle funzioni DLL più importanti: OpenDevice e CloseDevice. In questo articolo presentiamo anche (descritti negli appositi riquadri) due completi programmi: il primo consente di verificare il buon funzionamento della scheda alla fine del montaggio mentre il secondo è un software di uso generico che consente di utilizzare tutte le risorse disponibili sull'interfaccia. Si tratta di un programma completo in ogni sua parte che potrà essere utilizzato per comandare tutte le uscite della scheda - sia analogiche che digitali - e per leggere i valori d'ingresso.



MARCHEFIERE

ENTE REGIONALE PER LE MANIFESTAZIONI FIERISTICHE

**Quartiere Fieristico
CIVITANOVA MARCHE (MC)**

8-9-10 dicembre 2006

22^a Mostra Mercato Nazionale Radiantistica Elettronica

**Materiale radiantistico per C.B. e radioamatori
Apparecchiature per telecomunicazioni - Surplus
Telefonia - Computers
Antenne e Parabole per radioamatori e TV sat
Radio d'epoca - Editoria specializzata**

“2^o Mercatino del Radioamatore”

**Libero scambio tra privati
di apparecchiature amatoriali**



**In collaborazione con
Associazione Radioamatori Italiani
Sezione di Civitanova Marche**

DISCO

**Mostra mercato
del disco usato in vinile
e CD da collezione**

**Orario:
9,00-13,00
15,00-19,30**

**ERF • ENTE REGIONALE PER LE MANIFESTAZIONI FIERISTICHE
Quartiere Fieristico di Civitanova Marche • Tel. 0733 780811 • Fax 0733 780820**

www.erf.it e-mail: civitanova@erf.it

Diffusione wireless di audio digitale

di *Franco Tagliabue*

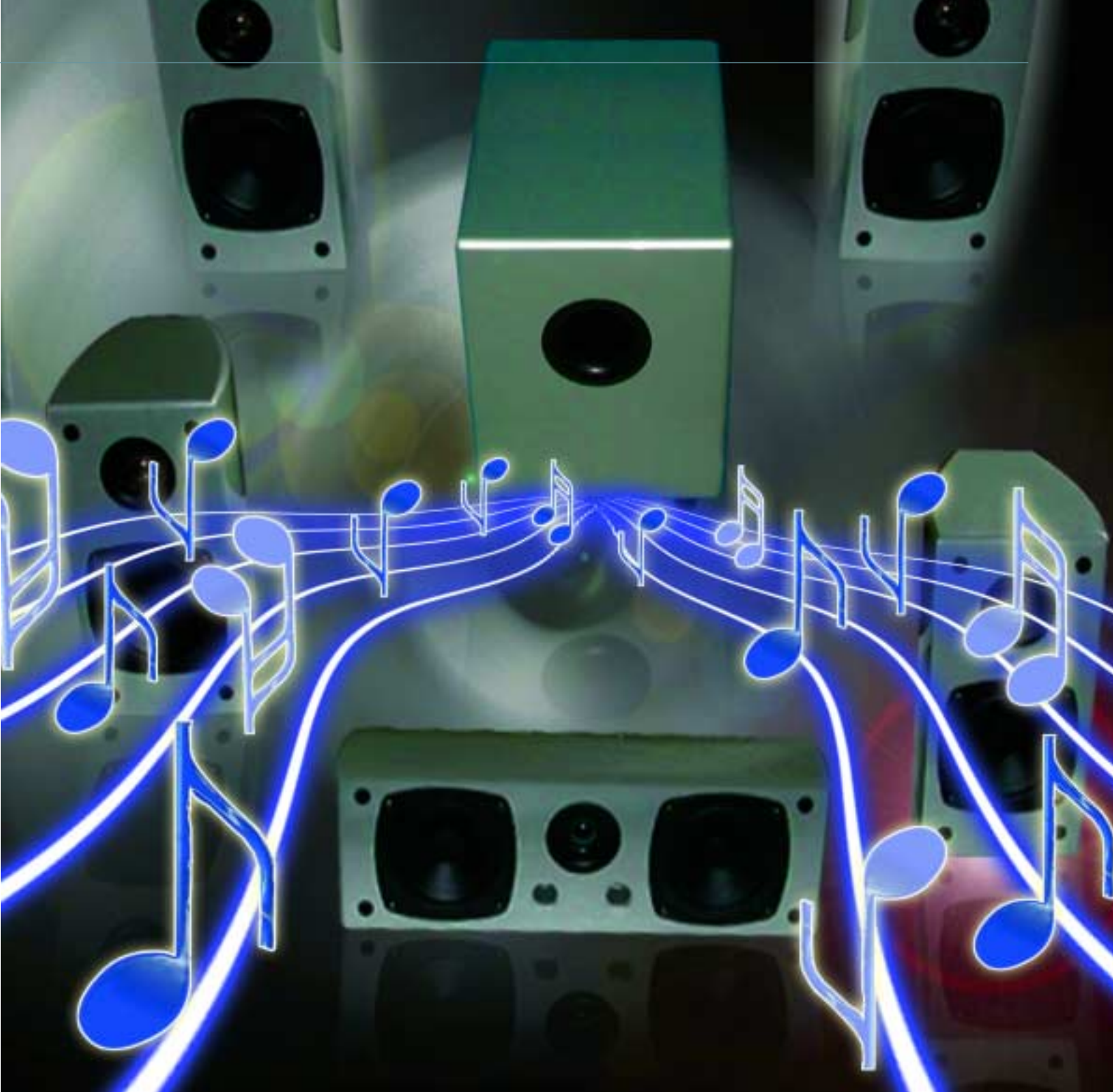
Sistema di diffusione sonora wireless a 2,4 GHz realizzato con due nuovi moduli Aurel.

La trasmissione avviene in modalità digitale FSK garantendo un'ottima immunità ai disturbi ed una qualità audio sorprendente. Dispone di 8 canali selezionabili tramite pulsante.



Ci sono situazioni in cui diffondere musica, o più in generale un segnale audio, in spazi particolarmente ampi, o non progettati per tale scopo, appare veramente difficile. Pensiamo ad esempio ad eventi sporadici come una manifestazione o una fiera, in questi casi la posa di cavi, oltre a richiedere molto tempo, si rivela costosa e determina spesso problemi di sicurezza. In ambito domestico i problemi non sono minori. Per esempio un buon impianto Home Theatre, richiede l'utilizzo, oltre che di casse frontali, di casse posizionate posteriormente allo spettatore. Il collegamento

delle prime solitamente non è un problema poichè vengono poste vicino al televisore e quindi il cavo non è eccessivamente lungo. Ma per gli altoparlanti posteriori, a meno di aver previsto apposite canaline dove far passare i fili, il collegamento è più scomodo. In questi casi la soluzione migliore è quella di utilizzare un dispositivo wireless che ha il vantaggio di essere semplicissimo da installare e a volte più economico di una soluzione via cavo. In questo articolo presentiamo un trasmettitore e un ricevitore, in grado di risolvere molti di questi problemi; il sistema utilizza due nuovissimi



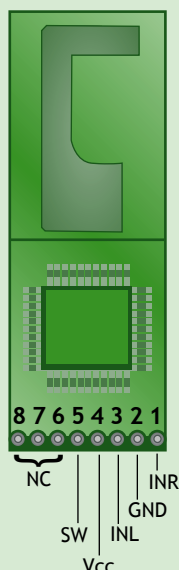
moduli Aurel denominati TX-AUDIO-24 e RX-AUDIO-24. Si tratta di due moduli che utilizzano una portante a 2,4 GHz modulata in FSK; il segnale stereo di ingresso viene digitalizzato da un convertitore A/D a 16 bit e la portante radio viene modulata dalla stringa di dati digitali di uscita. La trasmissione può avvenire su uno degli 8 canali disponibili, consentendo di utilizza-

re più coppie di moduli contemporaneamente senza che queste interferiscano tra loro, ovvero utilizzando più ricevitori sintonizzati sullo stesso trasmettitore. La modulazione con segnale digitale in FSK garantisce un'elevata immunità ai disturbi. Il campionamento del segnale audio avviene a 44,1 KHz con una risoluzione a 16 bit (qualità CD): la banda passante è notevoli

e la soppressione dei disturbi particolarmente efficace. Sempre riguardo all'audio, rammentiamo che il segnale in uscita viene automaticamente annullato in assenza di portante valida in modo da evitare fastidiosi fruscii.

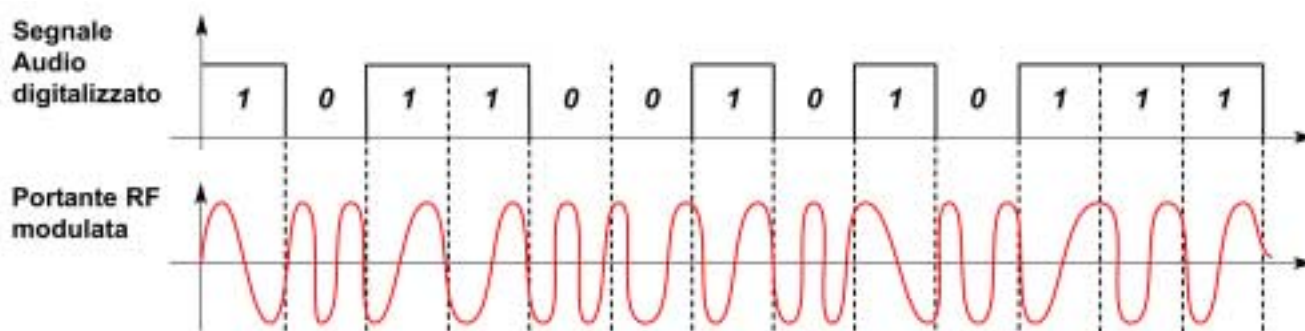
Nel riquadro di pagina 50 riportiamo le principali caratteristiche del modulo trasmettente TX-AUDIO-24: il segnale stereo presente in ►

Il modulo trasmittente TX-AUDIO-24



Descrizione	Valore
Tensione di alimentazione	3,6-5Vdc
Consumo di corrente	115mA (Tipico)
Temperatura operativa	-10 ~ +60°C
Gamma di frequenza RF	2400 ÷ 2486,5 MHz
Modulazione	FSK (Modulazione Index 0,5)
Numero di canali	8
Spazio canali	9MHz
Frequenza canale	2410, 2419 ~ 2473 MHz
Stabilità in frequenza	±100 kHz
Potenza TX	EIRP +10dBm
Impedenza d'ingresso BF	>10 Kohm
Livello d'ingresso BF	4Vp-p (Max)
Risposta in frequenza BF	20 ~ 20 kHz, -1dB
Dynamic range	92dB (Tipico)
Separazione	80dB (Tipico)
Rapporto S/N	87dB (Tipico)
THD	0,1% (Tipico)

Il modulo Aurel TX-AUDIO-24 lavora sulla banda dei 2,4 GHz, dispone di 8 canali ed effettua una digitalizzazione del segnale audio (stereo) d'ingresso mediante un A/D a 16 bit. Il campionamento avviene a 44,1 kHz consentendo di ottenere la cosiddetta "qualità CD" con una banda passante del segnale audio compresa tra 20 Hz e 20 kHz. La portante radio ha una potenza di +10 dBm (10 mW) e la modulazione è di tipo FSK. In basso è raffigurato il principio di funzionamento della modulazione di tipo FSK (Frequency-shift keying): due portanti a frequenze diverse vengono abbinate ai due valori logici 1 e 0 di cui è composta la stringa digitale che, nel nostro caso, corrisponde al segnale audio digitalizzato. Questo particolare tipo di modulazione ha il vantaggio di presentare un'elevata immunità ai disturbi in quanto è sempre presente un segnale trasmesso, a differenza della modulazione ASK (Amplitude-Shift Keying) dove il segnale digitale va a modulare una portante sinusoidale facendone variare l'ampiezza in modo tale da far corrispondere all'uno logico la portante stessa e, allo zero logico, l'assenza della portante. In quest'ultimo caso un disturbo può mettere in crisi il ricevitore che non riesce a discriminare se si tratta di un dato valido o meno.



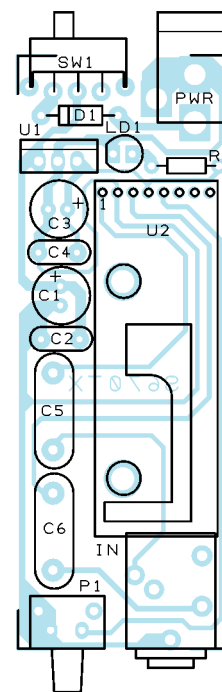
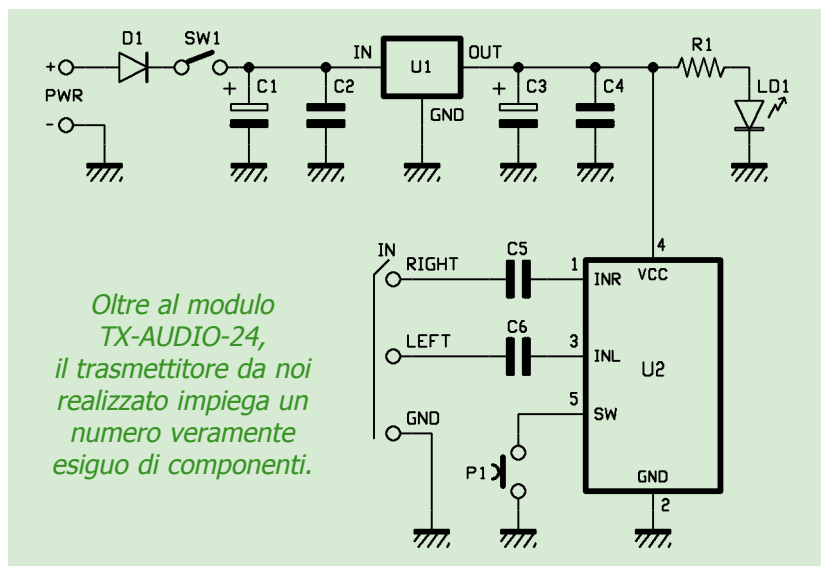
ingresso (INL e INR) viene campionato a 44,1 KHz, convertito da analogico a digitale e utilizzato per modulare in FSK la portante a 2,4 GHz. La portante radio viene successivamente amplificata e filtrata per poi arrivare all'antenna da stampato per la diffusione. Il modulo

trasmittente dispone di 8 pin di cui 5 utilizzati: due per gli ingressi audio, uno per la massa, uno per il positivo di alimentazione e l'altro per la selezione dei canali. Il modulo accetta un'alimentazione compresa tra 3,6 e 5 volt; ciò significa che il dispositivo è adatto ad impie-

ghi portatili con alimentazione mediante pile al litio o Ni-Cd a 3,6 volt. Ulteriori dettagli sono disponibili nel riquadro in alto. Il segnale irradiato verrà successivamente captato dal modulo RX-AUDIO-24, demodulato e riconvertito da digitale ad analogico fino ad ottene-

re un segnale stereofonico del tutto simile a quello utilizzato per pilotare il modulo trasmettente. Vediamo ora come abbiamo utilizzato il modulo TX-AUDIO-24 per realizzare un completo trasmettitore. Lo schema elettrico è riportato in questa stessa pagina. Il positivo di alimentazione è connesso al diodo D1 che crea la necessaria protezione contro le inversioni di polarità; C1, C2 a monte del regolatore di tensione e C3, C4 - a valle - filtrano eventuali disturbi. Il regolatore di tensione U1 è un 7805 in grado di fornire una tensione perfettamente stabilizzata di 5 V necessaria al corretto funzionamento del modulo. Dalle prove effettuate abbiamo riscontrato che l'assorbimento di corrente del circuito è leggermente superiore a quanto dichiarato nelle specifiche del modulo; suggeriamo pertanto di aggiungere, in caso di funzionamento particolarmente prolungato, un piccolo dissipatore termico al regolatore di tensione. I condensatori C5 e C6 servono per disaccoppiare l'ingresso audio mentre P1 consente di selezionare il canale di trasmissione fra gli 8 disponibili. Per cambiare il canale è sufficiente agire sul pulsante P1, ad ogni pressione del tasto si passa al canale successivo. A proposito di questo comando segnaliamo che il modulo TX-AUDIO-24 dispone di una memoria non volatile nella quale vengono salvate le impostazioni dei canali per cui, in caso di spegnimento e successiva riaccensione, la frequenza di lavoro non cambierà rispetto al precedente utilizzo. Per concludere, possiamo aggiungere che SW1 permette di accendere o spegnere il circuito mentre LD1 segnala se l'apparecchio è alimentato o meno.

Veniamo ora al sistema di ricezione che utilizza un nuovo modulo Aurel denominato RX-AUDIO-24. Come si vede nell'apposito riquadro di pagina 52, si tratta di un modulo ➤



ELENCO COMPONENTI:

R1: 470 ohm
C1: 220 μ F 25 VL elettrolitico
C2: 100 nF multistrato
C3: 220 μ F 25 VL elettrolitico
C4: 100 nF multistrato
C5: 1 μ F 100 VL poliestere
C6: 1 μ F 100 VL poliestere
U1: 7805
U2: TX-AUDIO-24
D1: 1N4007

LD1: led 3 mm verde
SW1: Deviatore slitta 90°
P1: Micro Switch 90°

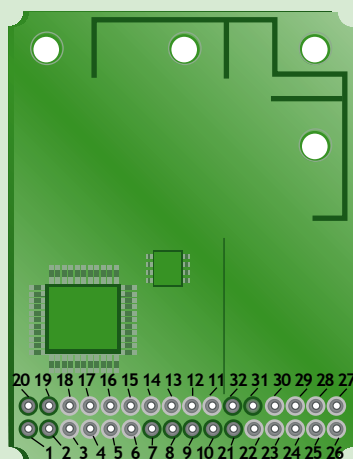
Varie:

- Plug alimentazione
- Presa jack 3,5 mm
- Strip femmina 8 pin passo 2 mm
- Circuito stampato 0670TX

Descrizione	Valore
Sezione Alimentazione	
Tensione alimentazione	5 ± 0,1 Vdc
Consumo di corrente	70mA @ 5Vdc (Tipico)
Sezione Audio	
Risposta in frequenza	20 ~ 20kHz, -1dB
Dynamic range	90dB (Tipico)
Separazione	80dB (Tipico)
Rapporto S/N	90dB (Tipico)
THD	-75dB (Tipico)
Livello d'uscita	3,4Vpp (Max)
Impedenza d'uscita	<1kohm
Sezione RF	
Gamma di frequenza	2400 ~ 2483,5 MHz
Demodulazione	FSK
Numero canali	8
Spazio canale	9MHz
Frequenze canali	2410, 2419 ~ 2473 MHz
Stabilità in frequenza	±100kHz
Sensibilità RX	-85dBm (Tipico)
Temperatura operativa	-10 ~ +60°C



PIN	Descrizione
1	PWR_ON
2	MUTE
3	USER_BIT
4	FORMAT
5	OB
6	TACT_SW
7	VCC
8	DAC_L
9	GND
10	DAC_R
11	ID2
12	ID3
13	SW0
14	ID3
15	ID2
16	-
17	ID0
18	CH_MODE
19	TACT_SCAN
20	CTINU
21	GND
22	DC_IN
23	GND
24	CH_R
25	GND
26	CH_L
27	DAC_L
28	-
29	AMP_R
30	AMP_L
31	GND
32	TACT_SW



Questo modulo è un completo ricevitore audio stereo multicanale PLL sulla banda dei 2,4 GHz con demodulatore FM e convertitore D/A a 16 bit ed interfaccia I2S che garantisce una banda passante del segnale audio compresa tra 20 Hz e 20 kHz. Esso rappresenta l'ideale complemento del trasmettitore TX-AUDIO-24 in quanto dispone di otto canali che utilizzano le medesime frequenze di lavoro ed un sistema di conversione D/A complementare rispetto a quello utilizzato dal TX. Anche in questo caso l'antenna è integrata nel circuito stampato. Nel presente riquadro riportiamo le caratteristiche principali di questo circuito la cui basetta misura esattamente 32 x 44 millimetri. La tensione nominale di funzionamento è di 5 volt con un assorbimento di corrente di 70 mA. Il circuito interno utilizza una media frequenza a 95 MHz, garantisce una sensibilità di -85 dbm e presenta una stabilità in frequenza di ±100kHz. La canalizzazione è di 9 MHz, identica a quella del modulo trasmettente. Numerose sono le modalità di funzionamento, specie per quanto riguarda la selezione dei canali che può essere manuale o automatica.

sicuramente più complesso, non foss'altro per il numero di pin di I/O. Come nel modulo TX, anche in questo caso l'antenna è integrata sullo stampato ed i canali a disposizione sono otto sulla banda dei 2,4 GHz. L'alimentazione è fissa ed è di 5 volt. Con questo modulo abbiamo realizzato il ricevitore descritto a pagina 53. La sezione di alimentazione è identica a quella del trasmettitore: la corrente passa attra-

verso il diodo D1 che protegge il circuito dalle inversioni di polarità; i 5V necessari per il funzionamento del modulo vengono ottenuti tramite il regolatore U1 e filtrati dai condensatori C1, C2, C3 e C4. Il modulo U2 (RX-AUDIO-24) richiede qualche componente in più rispetto al trasmettitore per poter funzionare. Osserviamo in particolare quanto accade ai pin 1, 2, 20 e 19. Tra i pin 1 e 2 sono collegati il diodo D2

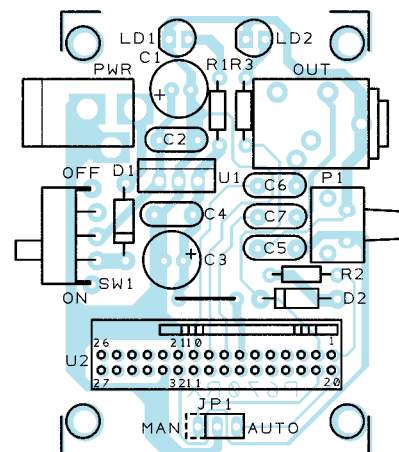
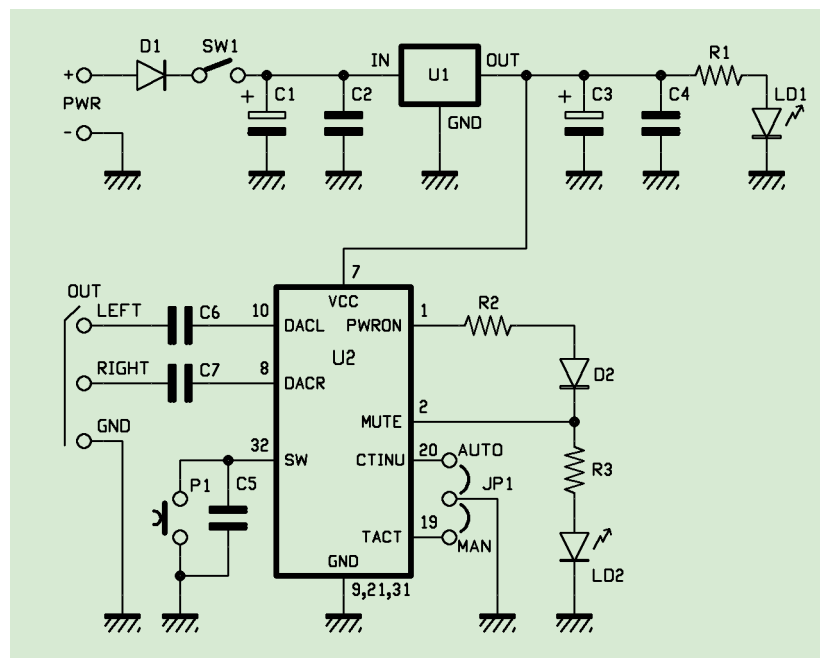
e la resistenza R2 mentre il led rosso LD2 è connesso in modo tale da illuminarsi quando è presente la portante radio. Questo ci permette di sapere quando siamo sintonizzati su un canale dove è in corso una trasmissione e allo stesso tempo ci avverte (spegnendosi) se la trasmissione è stata interrotta e quindi se il modulo è passato automaticamente in stato di mute, eliminando così eventuali disturbi. Tra i pin 19 e 20

è previsto un jumper che permette di portare un pin o l'altro a massa. Portando il pin 20 a 0V, viene selezionata la ricerca automatica di un canale, pertanto premendo brevemente P1 il ricevitore è in grado di trovare automaticamente un canale di trasmissione: se nel raggio di azione ci fossero altre unità trasmettenti verrebbe automaticamente sintonizzato il primo canale trovato. Portando il pin 19 a massa si seleziona la ricerca manuale: ad ogni pressione del pulsante P1 il modulo si sintonizza sul canale successivo. Le uscite audio sono disaccoppiate tramite i condensatori C6 e C7.

Poniamo ora l'attenzione sui livelli previsti per i segnali audio in ingresso, per il trasmettitore ed in uscita per il ricevitore. La banda passante del trasmettitore è compresa tra 20 e 20.000 Hz mentre l'ampiezza massima del segnale di ingresso non deve superare i 4 volt di picco, pena una forte distorsione; per quanto riguarda l'impedenza, il valore nominale è di 10 kohm. Tutto questo, in sintesi, significa che è possibile collegare una sorgente preamplificata come potrebbe essere l'uscita di un mixer o l'uscita di un impianto Home Theatre senza alcun problema. Il segnale in bassa frequenza che troviamo all'uscita del ricevitore può pilotare direttamente delle casse acustiche amplificate oppure può essere inviato ad un amplificatore di potenza di un impianto preesistente.

Procedure di montaggio

I componenti necessari per utilizzare i moduli sono veramente pochi e, nonostante l'elevato contenuto tecnologico dei componenti Aurel, la realizzazione è sicuramente alla portata di tutti: non ci sono punti di taratura e nemmeno servono strumenti sofisticati. Iniziamo dal trasmettitore. La basetta è particolarmente compatta, i pezzi sono pochi: quello che serve è solo un po' di ➤



ELENCO COMPONENTI:

R1: 470 ohm
 R2: 100 kohm
 R3: 1 kohm
 C1: 470 µF 25 VL elettrolitico
 C2: 100 nF multistrato
 C3: 470 µF 25 VL elettrolitico
 C4: 100 nF multistrato
 C5: 1 nF 100 VL poliestere
 C6: 10 nF 100 VL poliestere
 C7: 10 nF 100 VL poliestere
 U1: 7805
 U2: RX-AUDIO-24
 D1: 1N4007

LD1: led 3 mm verde
 LD2: led 3 mm rosso
 SW1: Deviatore slitta 90°
 P1: Micro Switch 90°

Varie:

- Plug alimentazione
- Presa jack 3,5 mm
- Strip maschio 3 pin
- Jumper
- Strip femmina 16x2 passo 2 mm
- Circuito stampato 0670RX

Il prototipo del ricevitore digitale a 2,4 GHz a montaggio ultimato.



attenzione ed un saldatore efficiente. Montate per primi C2, C4, R1, D1 e lo strip femmina ad 8 pin passo 2 mm; successivamente possiamo occuparci di tutti gli altri elementi, ovvero C1, C3 e U1: tutti componenti polarizzati per i quali bisogna prestare attenzione al corretto orientamento dei terminali. Per ultimo va inserito il modulo Aurel TX-AUDIO-24. Con estrema delicatezza inseriamo i suoi terminali nell'apposito connettore a strip precedentemente saldato ed il gioco è fatto.

La procedura di montaggio per il ricevitore è la stessa del TX: saldate per primi i componenti a più basso profilo e, poi, tutti gli altri facendo attenzione al verso di inserimento dei componenti polarizzati.

Per ultimo inserite il modulo Aurel utilizzando l'apposito connettore da stampato. Quando anche il ricevitore è pronto per le prove bisogna procurarsi gli alimentatori.

L'alimentazione

Durante le prove possiamo alimentare i due circuiti con qualsiasi sorgente di alimentazione, anche con batterie; nella installazione definitiva è invece consigliabile fare uso di due adattatori da rete in grado di erogare 12 Vdc con una corrente di 200-300 mA max.

Evitate di porre i due

circuiti molto vicini tra loro, ciò al fine di evitare interferenze.

Collegate una sorgente audio (può andare benissimo l'uscita di una scheda audio di un PC) all'ingresso del trasmettitore e delle casse amplificate (per esempio altoparlanti per computer) all'uscita audio del ricevitore. Posizionate il jumper del ricevitore in posizione ricerca automatica del canale (aiutatevi guardando la serigrafia) e alimentate i due circuiti: tutto deve funzionare al primo colpo e dovrete sentire l'audio trasmesso nelle casse collegate. Agendo sul pulsante del trasmetti-

tore provate a cambiare canale; il ricevitore avrà perso la portante quindi in uscita non dovrete sentire nulla; premete ora il pulsante sul ricevitore per fare ripartire la procedura di ricerca automatica del canale. Dopo il primo test eseguiamone un altro per controllare se la commutazione dei canali in modo manuale viene eseguita correttamente. Spegnete tutto e posizionate il jumper del ricevitore in posizione "sintonizzazione manuale".



Altrettanto semplice è il trasmettitore (nell'immagine il nostro prototipo).

Riaccendete il sistema e selezionate un canale per il TX e poi, premendo il tasto del ricevitore, provate a cercarlo. Quando il led rosso si illumina, significa che il ricevitore ha captato la portante e, di conseguenza, le casse riprodurranno ciò che viene trasmesso.

Consigli per l'uso

L'utilizzo in ambiente domestico del nostro sistema audio wireless non richiede particolari accorgi-

Per il MATERIALE

Tutti i componenti utilizzati in questo sistema wireless sono facilmente reperibili in commercio. Fanno eccezione i due moduli Aurel a 2,4 GHz che possono essere acquistati presso la ditta Futura Elettronica. Il modulo trasmettitore (cod. TX-AUDIO-24) costa 32,00 Euro mentre il modulo ricevitore (cod. RX-AUDIO-24) costa 44,00 Euro. I prezzi si intendono IVA compresa.

**Il materiale va richiesto a: Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 ~ Fax: 0331-778112 ~ <http://www.futuranet.it>**

menti se non piccole attenzioni. Ipotizziamo di prelevare il segnale audio per destinarlo ad una coppia di casse attive opportunamente posizionate. In questo caso ci serve un trasmettitore ed un ricevitore, se possiamo collegare l'uscita del ricevitore ad entrambe le casse attive, oppure due ricevitori se ci troviamo nell'impossibilità di collegare via filo le due casse.

Il segnale del medesimo trasmettitore può essere captato da più ricevitori sintonizzati sullo stesso canale. Se quindi desideriamo portare la musica in tutti gli ambienti di casa è sufficiente fare uso di tanti moduli quante sono le stanze, collegando all'uscita di ciascun modulo un diffusore acustico amplificato.

E se il nostro vicino di casa rimasto strabiliato dal nostro nuovo impianto ne preparasse uno uguale? Nessun problema, la portata massima di 50/100 metri si raggiunge all'aperto mentre in un condominio le mura e altri ostacoli riducono la possibilità di interferenze ai minimi termini; ma se il disturbo dovesse proprio esserci ricordiamoci che disponiamo di 8 canali, basta non usare lo stesso e le interferenze scompaiono, anche perchè i canali sono ben separati fra loro.

La possibilità di poter scegliere tra 8 canali di trasmissione, consente di utilizzare i moduli anche in altro modo: se disponete di più fonti sonore che volete diffondere via

radio, potete prevedere più trasmettitori configurati su canali differenti. Scegliendo la modalità manuale sul ricevitore e premendo più volte il pulsante P1 potrete selezionare la fonte audio preferita.

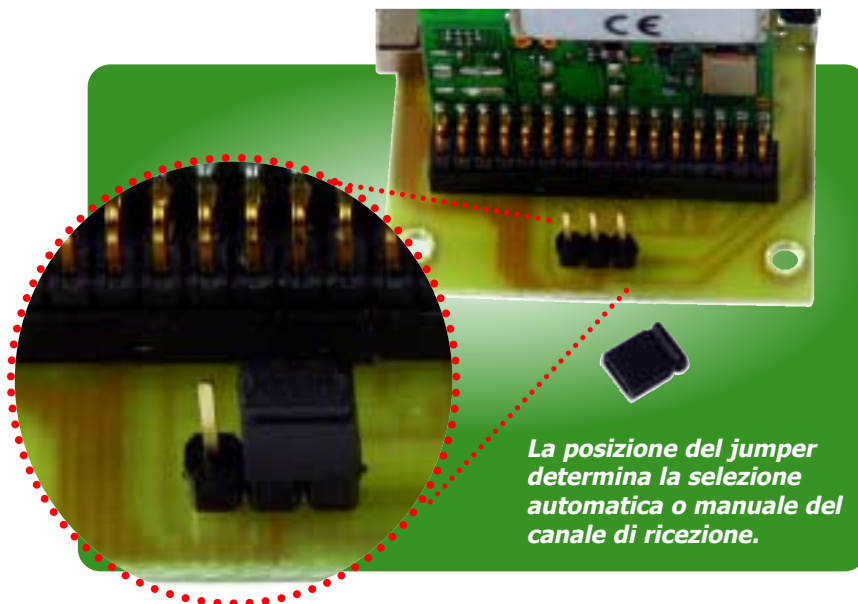
Questa soluzione è l'ideale per conferenze e convegni, in quanto consente agli ascoltatori di poter seguire il discorso nella propria lingua. Trasmettitori canalizzati in modo differente trasmettono lingue diverse, l'ascoltatore dotato di ricevitore può selezionare la propria lingua scegliendo il relativo canale.

Per ottenere il massimo delle prestazioni dall'impianto suggeriamo solo alcuni elementari accorgimenti: naturalmente non racchiudete il trasmettitore in un contenitore metallico che si comporterebbe

come uno schermo nei confronti dell'antenna presente sul circuito stampato, anche in considerazione del fatto che il nostro sistema utilizza una frequenza di funzionamento di 2,4 GHz. Lo stesso discorso vale ovviamente anche per il ricevitore. I contenitori, quindi, dovranno essere di materiale plastico.

Le notevoli prestazioni dei nuovi moduli Aurel, in particolare l'elevata qualità del segnale trasmesso e l'assenza di disturbi o interferenze, rendono questi dispositivi adatti ad un impiego professionale. Proprio per questo motivo nei prossimi mesi presenteremo altri interessanti progetti in campo audio con questi moduli.

Non perdetevi, dunque, i prossimi numeri di *ElettronicaIn!*



La posizione del jumper determina la selezione automatica o manuale del canale di ricezione.

Vendita componenti elettronici specializzati in forniture di enti pubblici e istituti professionali

Elettronica s.a.s

Via R. Grazzoli Lante, 22 - 00195 ROMA - Tel. 06 372 8112 - Fax. 06 3751 5380
www.elettronicaroma.it - email: elyda@tiscali.net

con il patrocinio del
Comune di Faenza

Fxp^o® Elettronica

mostra mercato

FAENZA 21-22 ottobre 2006

 **FAENZA FIERE - V.le Risorgimento, 1 ore 9/18**

**Elettronica per tutto,
Elettronica per tutti!**

Solo Sabato
21 ottobre

**RADIO:
EXPO'**

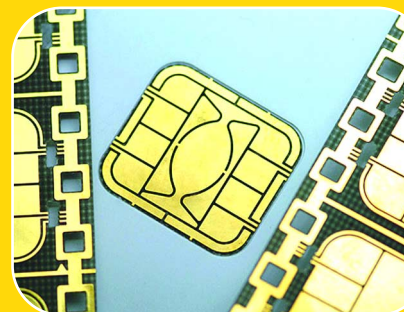
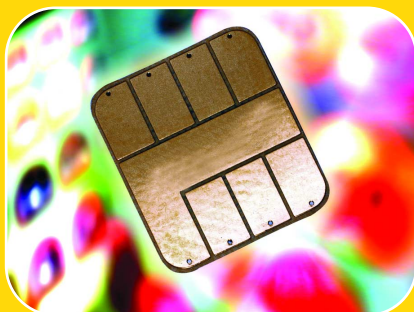
> organizzazione
BLU NAUTILUS srl
tel. 0541 439573
www.blunautilus.it

MERCATINO delle RADIOCOMUNICAZIONI
Apparecchi per radioamatori, Radio d'Epoca e da collezione,
Radio militari, Surplus, Valvole, Accessori, Ricambi, Riviste

Per ottenere un **INGRESSO RIDOTTO** presenta questa inserzione alla cassa

Corso di programmazione: **SMART CARD**

a cura di Fabio Riscica



Proseguiamo il nostro viaggio nel mondo delle Smart Card concludendo la descrizione della struttura della smartcard ACOS2 con particolare riferimento alla gestione dell'account ed ai relativi meccanismi di sicurezza. Quinta puntata.

5

In questa puntata concludiamo la descrizione della struttura della smartcard ACOS2, facendo riferimento in particolare alla gestione dell'account ed ai relativi meccanismi di sicurezza. Continuando il discorso del numero scorso, completiamo l'esame in dettaglio dell'area di memorizzazione interna.

Security File

Il Security File (ID FF03H) memorizza le seguenti informazioni:

- la coppia di chiavi utilizzata per la mutua autenticazione fra carta e terminale;
- i cinque Application Codes (AC1 .. AC5) utilizzati per il controllo dell'accesso ai files;
- l'Issuer Code (IC);
- il PIN;
- i contatori di errore che hanno la funzione di limitare il numero di verifiche errate delle pre-

cedenti chiavi (se uno di questi contatori raggiunge il valore 8, la corrispondente chiave è bloccata in modo irreversibile);

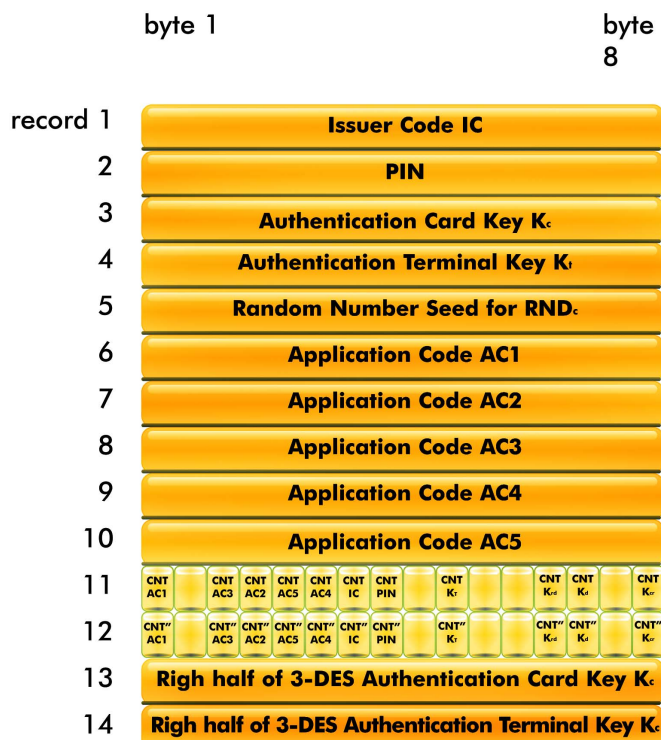
- il seme utilizzato per la generazione dei numeri random durante il processo di mutua autenticazione.

Ricordiamo che il Security File può essere letto soltanto durante la fase di Manufacturing e Personalization dietro presentazione dell'Issuer Code; la lettura di tale file non è più possibile nello User Stage.

La scrittura è invece sempre possibile dietro presentazione dell'IC (cioè nell'Issuer Mode).

Il Security File comprende 14 records di 8 bytes ciascuno ed è organizzato come mostrato in Figura 1.

I record 11 e 12 memorizzano i contatori di errore delle chiavi e la relativa copia di backup, mantenu- ➤

Security File:**Fig. 1**

definito nel Personalization File, che specifica il numero di file di dati utente allocati nella smartcard).

Come già accennato nel numero scorso, i record dello User File Management File possono essere liberamente letti e scritti (quest'ultima operazione deve essere preceduta dalla presentazione dell'IC) in ogni fase del ciclo di vita della carta, incluso lo User Stage. Ovviamente il numero di record è fissato dal parametro N_OF_FILE definito in fase di personalizzazione. La sequenza in cui i File Definition Blocks sono allocati nello User File Management File è irrilevante; nel momento in cui un file viene selezionato, il sistema operativo della carta scandisce i record alla ricerca dell'identificativo del file a cui

ta per sicurezza nel caso in cui si verificasse una perdita di alimentazione o un reset durante l'aggiornamento di questi valori. Si faccia attenzione al fatto che la scrittura di un valore

errato in questi record può bloccare in modo irreversibile la carta e renderla inutilizzabile; i contatori di errore vengono infatti gestiti dal sistema operativo della ACOS2 e non risulta necessario forzarne manualmente il valore (ogni contatore si azzerava automaticamente nel momento in cui viene presentato il relativo codice corretto). I record 13 e 14, presenti soltanto nella versione 3.0 della carta, memorizzano i bytes meno significativi delle chiavi di autenticazione 3-DES (se è selezionata l'opzione di DES semplice, questi record non sono utilizzati).

User File Management File

Lo User File Management File (ID FF04H) ha la funzione di specificare la struttura e la modalità di accesso degli User Data Files. Quest'ultimo è costituito da N_OF_FILE record di 6 bytes ciascuno e memorizza in ciascuno di questi il File Definition Block per ogni User Data File allocato (abbiamo già descritto nello scorso numero il meccanismo di accesso ai file utente ed il parametro N_OF_FILE,

**Fig. 2**

si fa riferimento. Per questo motivo il sistema operativo non è in grado di verificare la consistenza dei File Definition Blocks ed errori in questa struttura possono causare il malfunzionamento della carta. In Figura 2 vediamo il File Definition Block, costituito da 6 bytes, che comprendono, per ciascun file, l'identificatore, il numero di record, la lunghezza di ogni record e gli attributi di lettura e scrittura (il cui utilizzo è stato descritto nel precedente numero). La procedura da seguire per l'allocazione degli User Data Files consiste quindi nel definire il parametro N_OF_FILE nel Personalization File e quindi i File Definition Blocks nello User File Management File. Dopo queste operazioni gli User Data Files possono essere letti e scritti, con le condizioni di accesso definite.

Account File

L'Account File (ID FF05H) contiene l'Account Data Structure, che consente di utilizzare la smartcard come un borsellino elettronico. Bisogna

innanzitutto osservare che, nello User Stage, i dati all'interno dell'Account File non possono essere manipolati con le istruzioni di gestione dei files, ma sono disponibili soltanto alcuni particolari comandi che consentono di incrementare o decrementare il credito in modo estremamente sicuro, come vedremo più avanti.

L'Account Data Structure nell'Account File è costituita da 8 record di 4 bytes ciascuno e presenta la struttura descritta in Figura 3. Il primo byte del primo record prende il nome di TRANSTYPE e memorizza il tipo di transazione associata al bilancio corrente della carta. I tipi di transazione possibili sulla carta sono tre: CREDIT (incremento del bilancio secondo un valore fissato) è specificato dal valore 03H, DEBIT (decremento

del bilancio secondo un valore fissato) è specificato dal valore 01H, REVOKE DEBIT (annullamento dell'ultima operazione DEBIT) è specificato dal valore 02H. Il secondo, terzo e quarto byte del primo record costituiscono il bilancio corrente della carta (BALANCE). Si tratta di un numero intero positivo di 24 bit; il massimo valore che può essere rappresentato è quindi $2^{24}-1=16777215$.

Il primo e secondo byte del secondo record forniscono l'Account Transaction Counter (ATC). Si tratta di un contatore che viene incrementato dopo ogni transazione, in modo che possa essere generata una firma univoca per ogni transazione. L'ATC è utilizzato insieme all'Account Identification (AID) per generare l'Account Transaction reference (ATREF) che permette di calcolare il DES cipher-block chained checksum (MAC). Quest'ultimo permette di certificare i comandi relativi all'Account. Quando l'ATC raggiunge il suo valore massimo (FFFFH), il sistema operativo della carta non permette più l'esecuzione di alcuna

transazione. Il terzo byte del secondo record (CHKSUM) fornisce il byte meno significativo del checksum ottenuto sommando i bytes TRANSTYP, BALANCE ed ATC, più uno. Questo dato è gestito dal sistema operativo della

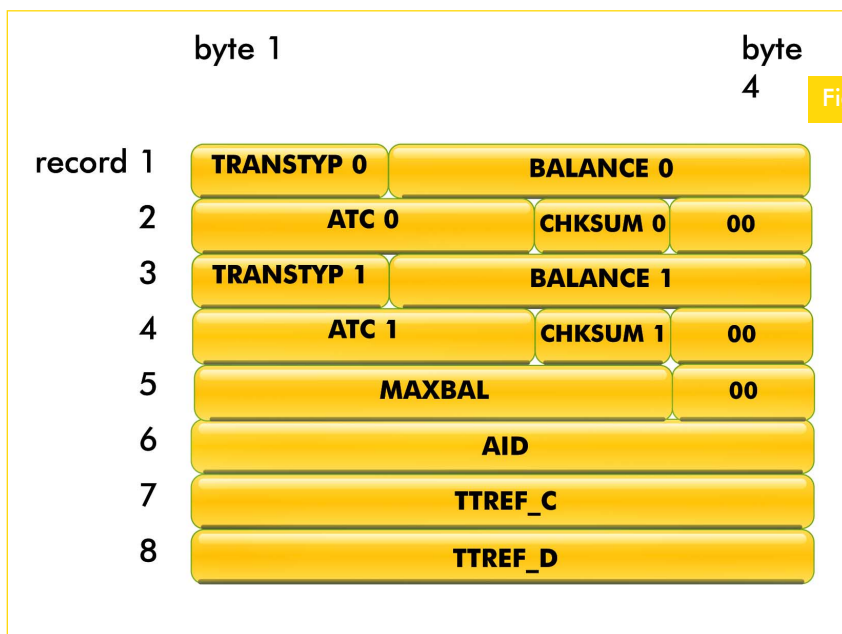
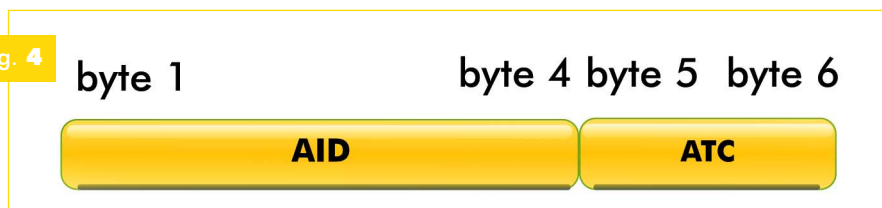


Fig. 3

carta ed ha la funzione di verificare la consistenza dei dati presenti sulla carta. Se il checksum è errato, la carta permette l'esecuzione delle transazioni soltanto nell'Issuer Mode, cioè dopo l'inserimento dell'IC. Il terzo e quarto record rappresentano un backup dei primi due record, anche qui mantenuti per sicurezza nel caso avvenga una mancanza di alimentazione o un reset durante la transazione. Il valore maggiore fra i due ATC permette di riconoscere la transazione più recente. I primi tre bytes del quinto record (MAXBAL) hanno la funzione di limitare l'importo contenuto nella carta. Questo valore viene verificato dal sistema operativo prima di ogni comando CREDIT; se la somma del bilancio corrente e del credito eccede MAXBAL, la carta rifiuta il comando CREDIT. I quattro bytes del sesto record prendono il nome di Account Identification (AID) e sono combinati con l'ATC per ottenere i sei bytes ATREF, il cui utilizzo è già stato accennato in precedenza. L'AID è scritto durante la fase di personalizzazione e non è modificabile (vedi Figura 4).

I quattro bytes del settimo record costituiscono il Terminal Transaction Credit Reference (TTREF- ➤

Fig. 4



Key storage for Single DES

Fig. 5

Record No	Byte 1							Byte 8
1	K _D							
2	K _{CR}							
3	K _{CF}							
4	K _{RD}							

C), dato fornito dal Card Accepting Device, cioè dal terminale esterno, quando è eseguita una transazione di tipo CREDIT. Questo dato non viene interpretato dal sistema operativo dalla carta ma può essere letto ed utilizzato dal terminale, ad esempio, per rifiutare una carta che è stata accreditata da un terminale non autorizzato.

Infine, i quattro bytes dell'ottavo record costituiscono il Terminal Transaction Debit Reference (TTREF-D), dato anch'esso fornito dal Card Accepting Device quando è eseguita una transazione DEBIT od una REVOKE DEBIT. Si osserva che il comando REVOKE DEBIT può essere eseguito soltanto se il TTREF-D fornito con il comando è identico a quello memorizzato. Ciò garantisce che il terminale che richiede il REVOKE DEBIT è lo stesso che ha eseguito il precedente DEBIT.

Account Security File

L'Account Security File (ID FF06H) contiene le chiavi di codifica usate nel calcolo del MAC durante le operazioni relative alla gestione dell'Account. Nel caso in cui sia stata abilitata la crittografia 1-DES, l'Account Security File è costituito da 4 record di 8 bytes ciascuno e contiene le quattro chiavi di 8 bytes utilizzate: la DEBIT KEY, utilizzata dal terminale nel calcolo del MAC per costruire il comando DEBIT da inviare alla carta; la CREDIT KEY, utilizzata dal terminale nel calcolo del MAC per costruire il comando CREDIT

da inviare alla carta; la CERTIFY KEY, utilizzata dalla carta per generare il MAC contenuto nella risposta del comando INQUIRE ACCOUNT da inviare al terminale (interrogazione dello stato dell'account), la REVOKE DEBIT KEY, utilizzata dal terminale nel calcolo del MAC per costruire il comando REVOKE DEBIT da inviare

alla carta (Figura 5). Nel caso in cui invece sia stata abilitata la crittografia 3-DES, l'Account Security File è costituito da 8 record di 8 bytes ciascuno, in quanto in questo caso le chiavi sono ciascuna di 16 bytes di lunghezza.

USER DATA FILES E MODALITA' DI ACCESSO AI DATI

Iniziamo adesso ad esaminare più attentamente la struttura dell'area dei files dati utente, la User Data Files. Quest'area viene allocata nella fase di personalizzazione della carta; i dati memorizzati possono essere letti e scritti con i comandi READ RECORD e WRITE RECORD, a patto che le condizioni d'accesso specificate nei bytes 3 e 4 dello User File Definition Block siano soddisfatte. Dopo la fase di personalizzazione, in cui vengono scritti i records dello User File Managements File, non è più possibile modificare il numero di record di un Data File se questo è stato utilizzato almeno una volta. Infatti, una volta allocato lo spazio di memoria, lo stesso non può essere più liberato anche se i files di dati non sono più utilizzati.

Ogni User Data File può contenere al più 255 record con un massimo di 32 bytes ciascuno. Lo spazio di memoria complessivamente disponibile per l'area utente è teoricamente di 8 KBytes, ma bisogna considerare che una parte di questa viene utilizzata dal sistema operativo. Ad esempio, se nella carta è implementata l'Account Data Structure, bisogna tener conto del fatto che sono

Tabella 1

Size	Account Data Structure not available (ACCOUNT = 0)	Account Data Structure available with 1-DES (3-DES=0 and ACCOUNT = 1)	Account Data Structure available with 3-DES (3DES=1 and ACCOUNT=1)
No of Files even	7964 - N*6	7900 - N*6	7868 - N*6
No of Files odd	7962 - N*6	7898 - N*6	7866 - N*6

riservati a questa 64 bytes (nel caso in cui l'Account Data Structure utilizzi il DES semplice) o 96 bytes (nel caso di 3-DES). Bisogna prestare attenzione a non allocare memoria per i files utente oltre la quantità disponibile: il sistema operativo non effettua alcun controllo sul superamento di tale limite e quindi, in tal caso, potrebbero verificarsi inaspettati malfunzionamenti.

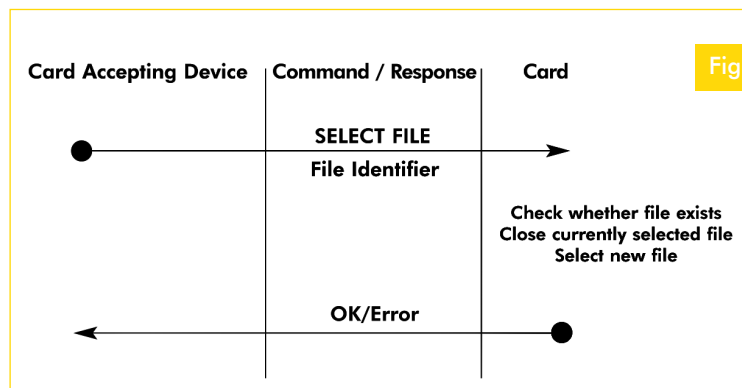
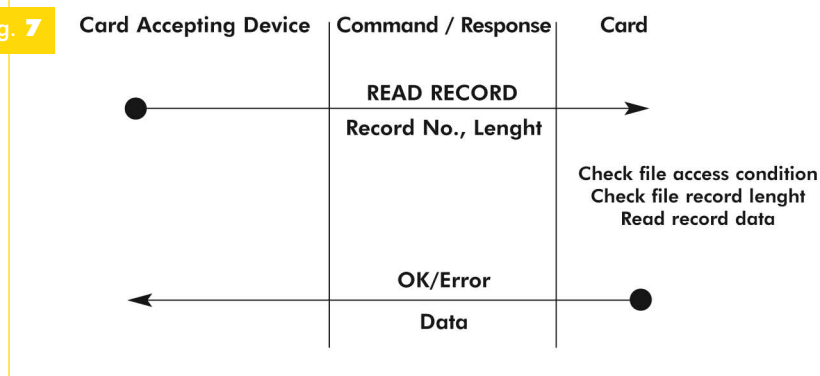


Fig. 7

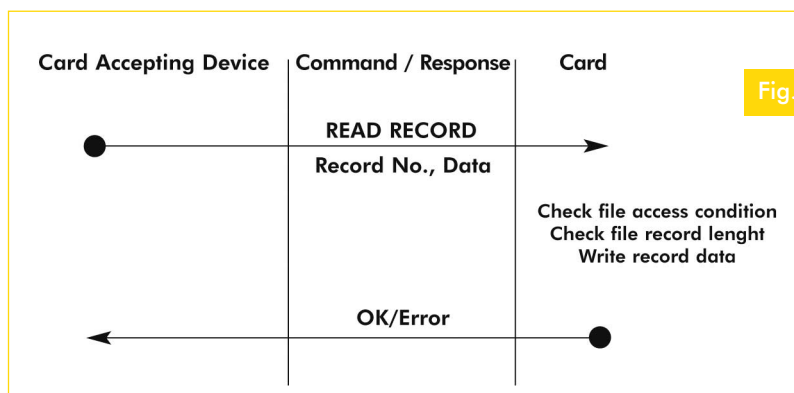


zionato alcun file. La Figura 6 riassume l'esecuzione del comando SELECT FILE. Il terminale deve specificare l'identificativo del file da selezionare; la carta restituisce lo status code che segnala se il comando è andato a buon fine. Dopo aver selezionato il file è possibile eseguire le operazioni di lettura e scrittura dei record dello stesso.

menti della carta. La Tabella 1 permette di calcolare in modo preciso la quantità di memoria disponibile per gli User Data Files. Si osservi che per motivi legati all'architettura della carta ACOS2, i files dati utente occupano uno spazio di memoria sempre multiplo di 4 bytes. Passiamo adesso a descrivere il meccanismo di accesso ai files dati, che è peraltro identico sia per gli User Data Files che per gli Internal Data Files (ad eccezione, in modalità User, dell'Account File).

La prima azione che deve essere necessariamente eseguita per l'accesso ad un generico file dati è la selezione mediante il comando SELECT FILE. Dopo l'esecuzione di questo comando, il file specificato (se esistente) sarà selezionato e l'eventuale file correntemente selezionato sarà chiuso. Se l'identificativo del file specificato dal comando è inesistente, la carta restituisce un codice di errore ed il file correntemente selezionato non cambia. Si osservi che non è necessario specificare le condizioni di accesso al file durante l'esecuzione di questo comando. Inoltre, dopo un reset della carta, non risulta sele-

Il comando di lettura è READ RECORD e, se le condizioni di accesso relative al file non sono state soddisfatte prima dell'esecuzione del comando, quest'ultimo viene rifiutato dalla carta. Ogni comando permette di leggere soltanto un record del file selezionato; il numero di bytes da leggere è

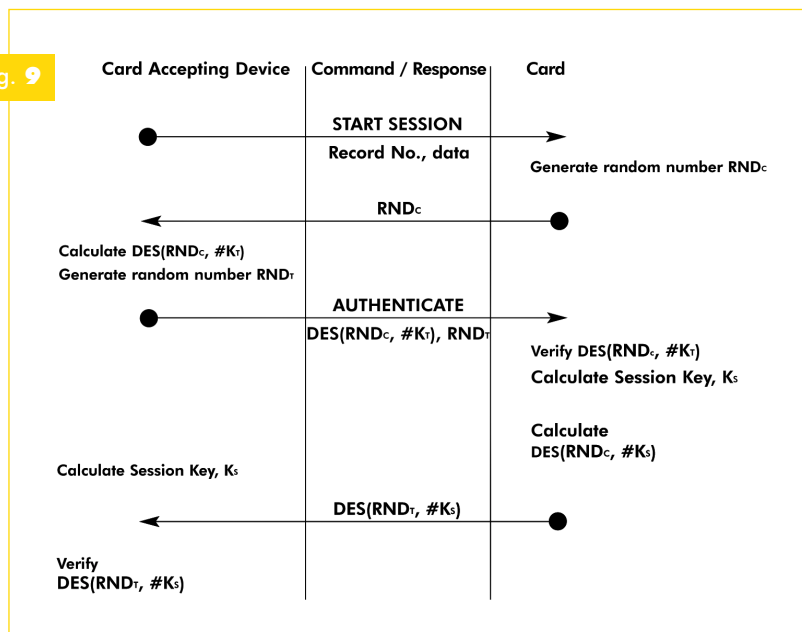


specificato nel comando stesso (ovviamente il massimo numero di bytes che può essere letto è pari alla lunghezza del record).

La Figura 7 riassume l'esecuzione del comando READ RECORD. Il terminale deve specificare l'indice del record (l'indice del primo record è specificato dal bit RECORD_NUMBERING del Manufacturer File ed è generalmente pari a 0) ed il numero di bytes da leggere (al più 32); la carta ➤

restituisce i bytes richiesti seguiti dallo status code. Analogamente a quanto visto per la lettura è possibile scrivere i bytes di un record tramite il comando WRITE RECORD. Dopo aver selezionato il file e verificato le condizioni di accesso, è possibile scrivere un record come mostrato in figura. Il terminale deve specificare l'indice del record ed i bytes da scrivere; la carta restituisce lo status code (Figura 8).

Fig. 9



SICUREZZA E TRANSAZIONI SULL'ACCOUNT

Passiamo adesso ad esaminare tutti i meccanismi di sicurezza messi a disposizione dal sistema operativo della carta ACOS2.

Mutua autenticazione

Il primo meccanismo che può essere abilitato (mediante i bit TRNS_AUT e INQ_AUT dell'Option Register) è quello della mutua autenticazione fra Card Accepting Device (terminale) e carta e si basa sullo scambio e verifica delle chiavi in modo sicuro (cioè non in chiaro ma crittografate secondo l'algoritmo DES o 3-DES, secondo l'opzione selezionata nell'Option Register).

Quando la mutua autenticazione è abilitata mediante il bit TRNS_AUT, ogni transazione sull'account, affinché vada a buon fine, deve essere preceduta da questa operazione; analogamente, se è abilitata mediante il bit INQ_AUT, lo stato dell'account può essere interrogato soltanto dopo che la mutua autenticazione è stata effettuata. Vediamo di riassumere nella figura sottostante il meccanismo della mutua autenticazione. Il terminale inizia inviando alla carta il comando START SESSION, che comunica alla stessa di voler iniziare il processo di mutua autenticazione. La carta risponde generando ed inviando un numero random di 8 bytes necessario per continuare l'operazione, seguito dallo status code. A questo punto il terminale crittografa il numero random ricevuto dalla carta utilizzando la Terminal Key, genera anch'esso un secondo numero random ed invia questi due dati alla carta nel corpo del comando AUTHENTICATE. La carta verifica innanzitutto se il dato inviato dal terminale è corretto, quindi genera una nuova chiave, la Session Key (calcolata in base

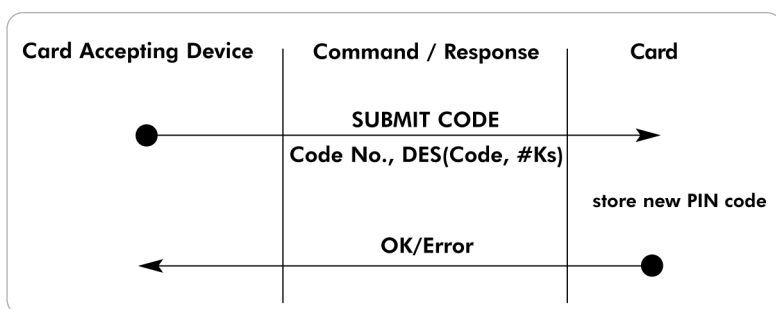
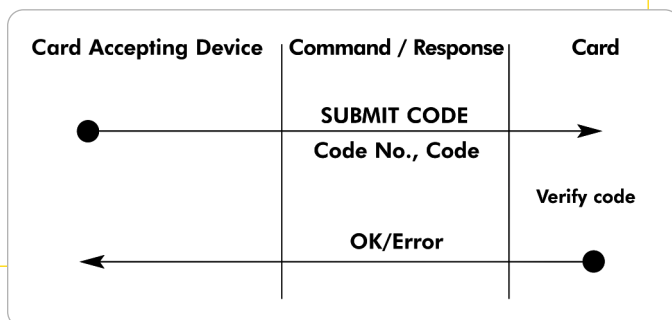


Fig. 10



al valore dei due numeri random, della Terminal Key e della Card Key), con cui crittografa il numero random inviatogli dal terminale. Questo dato viene inviato in risposta al comando AUTHENTICATE, seguito al solito status code. Infine il terminale, che ha calcolato anch'esso la Session Key, verifica la risposta inviatogli dalla carta concludendo la mutua autenticazione (vedi Figura 9). Il risultato finale della mutua autenticazione è la generazione della Session Key, che verrà utilizzata dalla carta per crittografare e decrittografare i dati nel corso della stessa sessione. La mutua autenticazione fra carta e terminale deve essere completata nell'ordine specificato; se nel corso della mutua autenticazione viene inviato alla carta un comando differente da quello aspettato, questa viene interrotta e deve essere nuovamente eseguita dall'inizio con il comando START SESSION. Se il comando di AUTHENTICATE non invia alla carta il dato correttamente crittografato, il contatore della Terminal Key viene incrementato (se viene raggiunto il valore 8 il comando AUTHENTICATE non potrà esse-

re più eseguito dalla carta, che risulterà inutilizzabile). Il contatore della Terminal Key viene

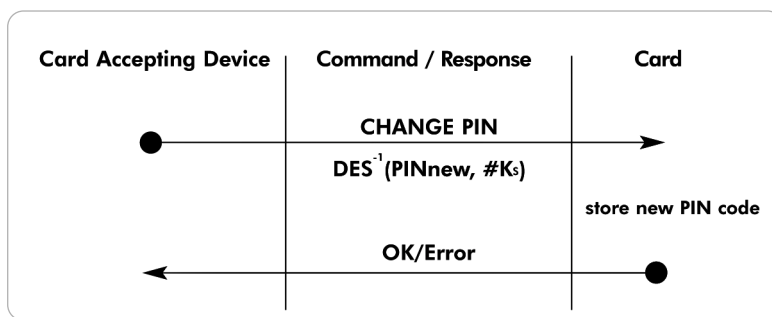
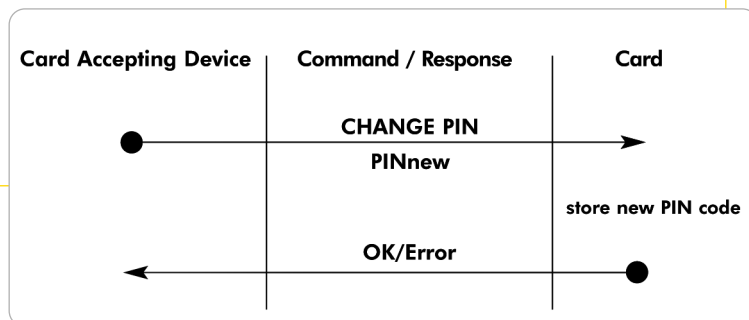


Fig. 11



azzerato (se non bloccato) non appena il comando di AUTHENTICATE va a buon fine.

Verifica e modifica delle chiavi

Abbiamo già parlato delle chiavi della carta, che hanno la funzione di limitare l'accesso ai dati ed alle operazioni che possono essere eseguite. Nella carta ACOS2 possono essere memorizzati cinque Application Codes (AC1, AC2, AC3, AC4, AC5), un Issuer Code (IC) ed un PIN Code (PIN). L'Issuer Code è la chiave di amministrazione della carta e consente di eseguire delle operazioni privilegiate; le altre chiavi vengono generalmente utilizzate per limitare l'accesso ai files dati (inoltre sul

PIN può essere eseguito un particolare comando di modifica di cui parleremo fra poco).

La verifica delle chiavi può essere eseguita in chiaro od in modalità crittografata, in base al valore dei bit del Security Option Register. Il comando che permette di verificare le chiavi è SUBMIT CODE e viene schematizzato nelle immagini di Figura 10. Il terminale deve specificare l'indice della chiave da verificare ed il

valore di 8 bytes, eventualmente crittografato con la Session Key; la carta dopo aver effettuato la verifica risponde con lo status ➤

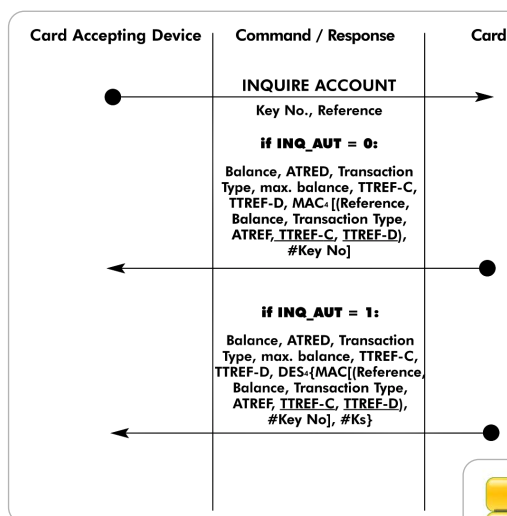


Fig. 12

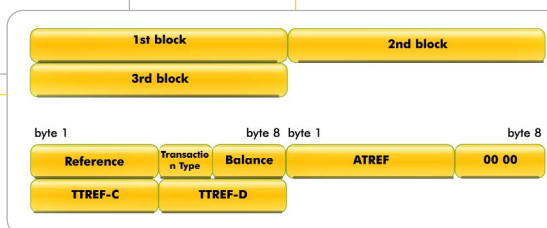
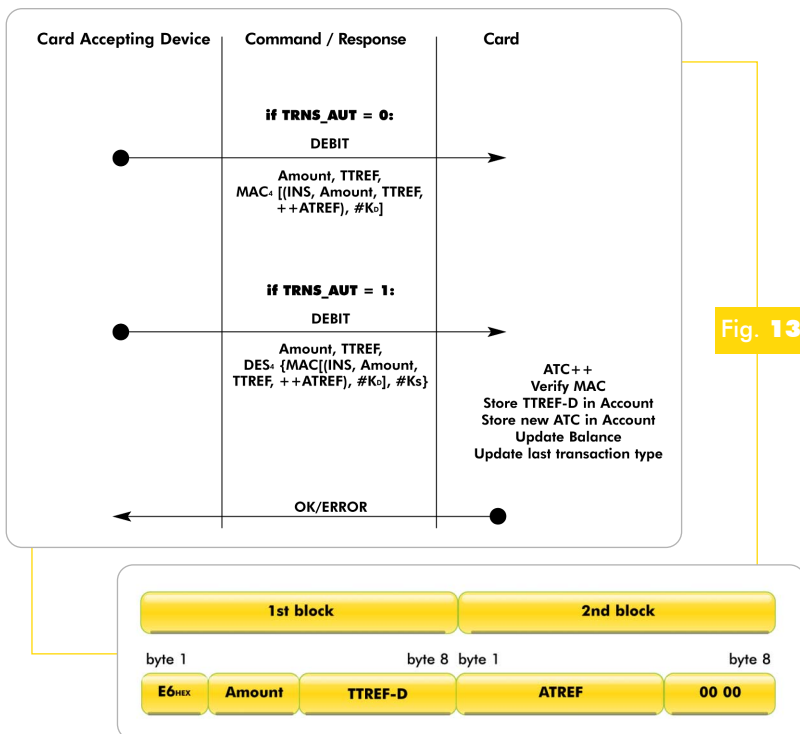


Tabella 2

DEB_MAC	DEB_PIN	Livello di Sicurezza
0	0	Nessuno; il comando DEBIT può essere sempre eseguito.
0	1	Il comando deve essere preceduto dalla verifica del PIN.
1	0	Il calcolo del MAC è richiesto e verrà verificato dalla carta.
1	1	Il comando deve essere preceduto dal PIN ed il calcolo del MAC è richiesto.



code. Gli indici da 1 a 5 identificano i rispettivi Application Codes, il 6 è relativo al PIN ed il 7 identifica l'Issuer Code.

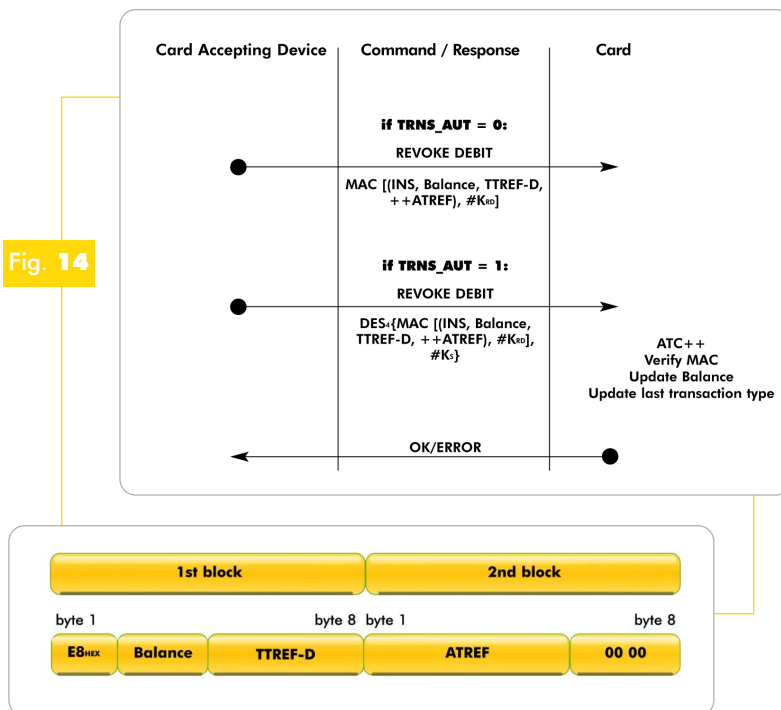
Ricordiamo che la carta memorizza un contatore per ogni chiave e limita ad otto il numero di esecuzioni senza successo del comando **SUBMIT CODE**. Anche per questi contatori valgono le stesse regole viste in precedenza per quello della Terminal Key. Le chiavi possono essere modificate dopo la presentazione dell'Issuer Code con il comando **WRITE RECORD**. Se il bit **PIN_ALT** nell'Option Register è settato, il PIN può invece essere modificato con il comando **CHANGE PIN**. Per motivi di sicurezza questo comando può essere eseguito soltanto dopo che il vecchio PIN è stato verificato ed immediatamente dopo il pro-

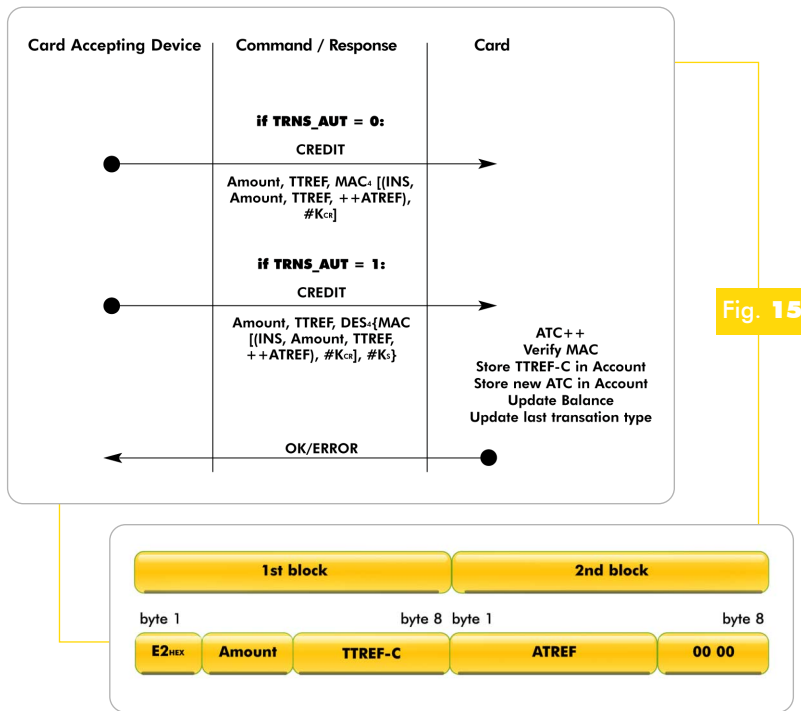
cesso di mutua autenticazione. Anche in questo caso il dato può essere inviato in chiaro o crittografato, in base al valore del bit **PIN_DES** del Security Option Register. Le immagini di Figura 11 schematizzano l'esecuzione del comando **CHANGE PIN**. Il terminale deve specificare il valore del nuovo PIN, eventualmente crittografato con la Session Key; la carta memorizza questo valore e risponde con lo status code.

Transizioni sull'account

Quattro chiavi, memorizzate nell'Account Security File, sono associate alla gestione dell'Account: la Credit Key, la Debit Key, la Certify Key, la Revoke Debit Key. Queste chiavi sono utilizzate nel calcolo e nella verifica del MAC (checksum crittografico), nei comandi relativi alla gestione dell'Account. Ad eccezione della Certify Key, ad ogni chiave è associato un contatore di errore che limita il numero di

Fig. 14





possono essere eseguite quattro differenti tipi di transazione:

- INQUIRE ACCOUNT;
- DEBIT;
- REVOKE DEBIT;
- CREDIT.

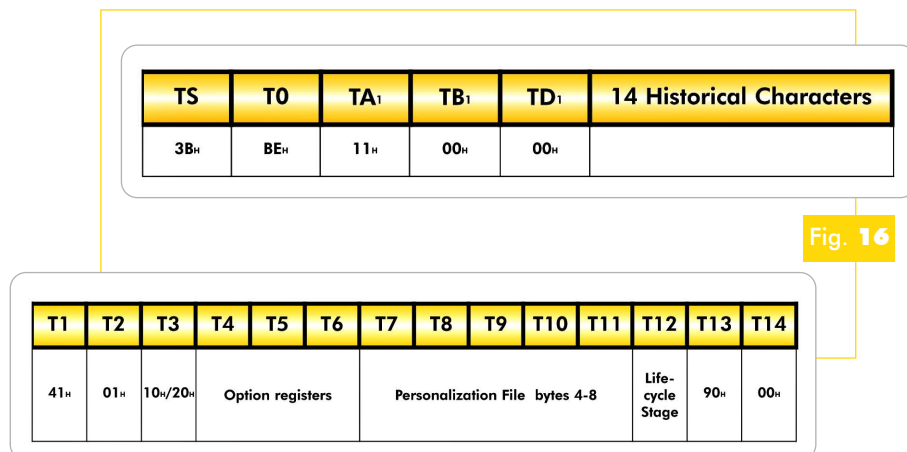
Nella transazione di tipo INQUIRE ACCOUNT la carta restituisce il valore del bilancio corrente seguito da altre informazioni sull'account e dal MAC calcolato sui dati precedenti, che può essere considerato dal terminale come un certificato di autenticità fornito dalla carta. Si ricordi che, se il bit INQ_AUT è settato, il comando INQUIRE ACCOUNT deve essere preceduto dal processo di mutua autenticazione. I disegni di Figura 12 schematizzano l'esecuzione del comando INQUIRE ACCOUNT. Il terminale deve specificare la chiave che la carta utilizzerà per il calcolo del MAC ed un valore di quattro bytes chiamato Reference, che condiziona il valore del dato restituito; la carta restituisce le informazioni sull'account ed il MAC (eventualmente crittografato con la Session Key se INQ_AUT=1) seguito dallo status code. Nella transazione di tipo DEBIT, il bilancio dell'Account è decrementato dell'importo specificato. Il massimo importo che può

essere addebitato è ovviamente pari al valore del bilancio corrente (non sono ammessi valori negativi). A seconda del valore assunto dai bit DEB_MAC e DEB_PIN dell'Option Register, possono essere specificati diversi livelli di sicurezza (vedi Tabella 2). Anche in questo caso, se il bit INQ_AUT è settato, il comando DEBIT deve essere preceduto dal processo di mutua autenticazione (vedi immagini della Figura 13). La transazione di tipo REVOKE DEBIT è possibile soltanto immediatamente dopo una transazione DEBIT per annullare un errato addebito sulla carta; è possibile abilitare e disabilitare tale opzione mediante il bit REV_DEB dell'Option Register. Come al solito, se il bit INQ_AUT è settato, il

comando REVOKE DEBIT deve essere preceduto dal processo di mutua autenticazione (Figura 14). Nella transazione di tipo CREDIT, il bilancio dell'Account è incrementato dell'importo specificato. Il massimo importo che può essere accreditato è limitato dal valore MAXBAL, memorizzato nell'Account Data Structure. La transazione CREDIT può essere eseguita soltanto ad un elevato livello di sicurezza perciò se il bit INQ_AUT è settato, il comando CREDIT deve essere preceduto dal processo di mutua autenticazione (Figura 15).

Struttura dell'ATR

Dopo un reset hardware, la carta trasmette l'ATR secondo lo standard ISO 7816-3. La carta ACOS2 supporta il protocollo T=0 con la convenzione diretta della codifica dei bit. Le immagini riportate in Figura 16 illustrano i dati trasmessi dalla carta.



VIDE
SUMMIT



GOV
2006

IL PRIMO EVENTO DEDICATO ALLA VIDEOSICUREZZA PUBBLICA

CONTROLLO, PROTEZIONE E PREVENZIONE:
UNA PANORAMICA DELLA VIDEOSORVEGLIANZA
IN AMBITO PUBBLICO

19 Ottobre 2006
Palazzo Turati, Milano

CORSI DI FORMAZIONE

- _Videosorveglianza wireless
- _Videosorveglianza professionale su IP
- _Videosorveglianza & privacy

ESCLUSIVO

Proposta formativa gratuita
per le imprese artigiane della
Regione Lombardia

I TEMI DELLA CONFERENZA:

Normativa e privacy
Videosorveglianza IP
Videosorveglianza wireless
Analisi dei flussi video
Telecamere e Video-server
Sicurezza urbana
Esperienze e progetti

Organizzato da

Il Forum della NetEconomy e' istituito da

Segreteria Organizzativa

FORUM NETECONOMY



www.videogov.com

INGRESSO GRATUITO RISERVATO AGLI OPERATORI DEL SETTORE

Controllo Bluetooth con telefono cellulare

di Boris Landoni

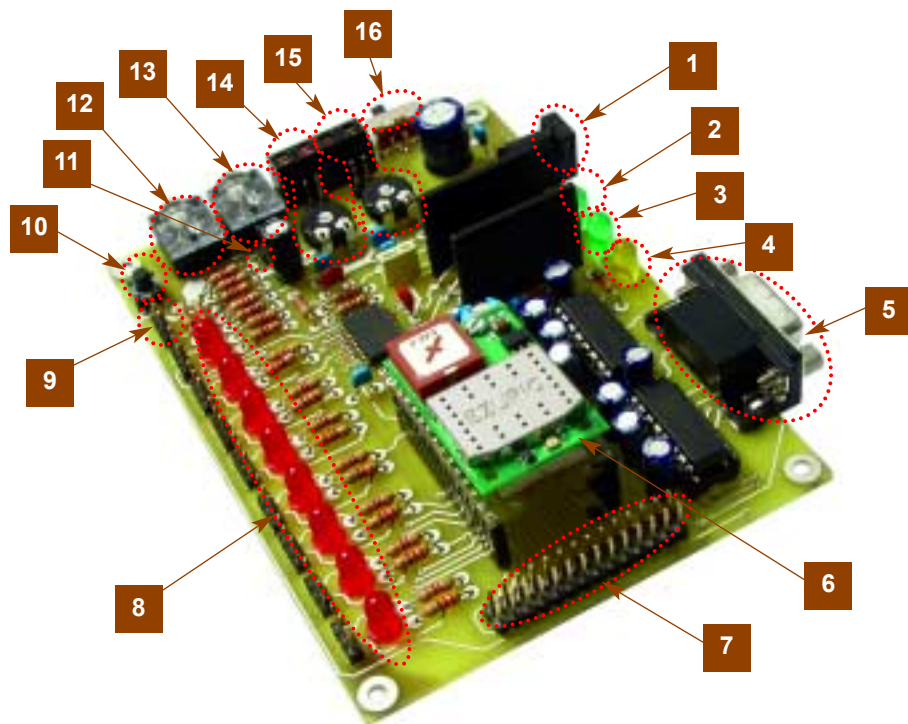
Utilizziamo un cellulare dotato di interfaccia Bluetooth per controllare gli ingressi e le uscite della demoboard presentata alcuni mesi fa. Il telefono cellulare deve essere in grado di supportare applicativi Java.



La DemoBoard Bluetooth presentata nei numeri 106 e 107 della rivista è un progetto che ben si presta a divenire la base per compiere sperimentazioni che portino alla realizzazione di nuovi dispositivi e sistemi basati su questa tecnologia. Molti lettori hanno già utilizzato la DemoBoard avvalendosi del software da installare su PC che gestisce la comunicazione Bluetooth; ora come promesso, è il momento del passo successivo: adoperare la scheda tramite un telefonino Bluetooth che abbia la possibilità di utilizzare i programmi applicativi scritti in linguaggio Java. Questo

passaggio è particolarmente significativo in quanto è nostra intenzione proporre in futuro altri progetti basati sul modulo Bluetooth BISM2 della Ezurio. Per quanto riguarda le prestazioni di detto modulo (che rappresenta il “cuore” della nostra DemoBoard), precisiamo che esso appartiene alla classe 1, pertanto, in assenza di ostacoli, prevede una portata teorica di circa 250 metri. Ciò non significa che la scheda riceverà sicuramente segnali da un telefonino posto a tale distanza, in quanto per valutare la portata reale dobbiamo considerare la potenza del modulo Bluetooth presente all’interno del ➤

Le connessioni della Demo Board



Rif.	Descrizione
1	Alimentazione scheda 12Vdc
2	Led di alimentazione principale
3	Led di alimentazione del modulo
4	Led di connessione
5	Porta seriale
6	Modulo Bluetooth Ezurio
7	Connettore esterno per test moduli
8	Ingressi / uscite digitali

Rif.	Descrizione
9	Filtro passa-alto antironzio
10	Reset del modulo
11	Polarizzazione del microfono
12	Altoparlante
13	Microfono
14	Ingresso analogico 1
15	Ingresso analogico 2
16	Accensione modulo

telefonino e quindi, anche se il modulo Ezurio può operare fino a 250 metri, non è detto che il telefonino sia in grado di fare altrettanto. Quasi sempre, infatti, i cellulari possono operare in modalità Bluetooth solo a qualche decina di metri. In conclusione possiamo dire che la DemoBoard si presta per lo sviluppo di telecontrolli e la gestione di segnali in ambito locale, come

un appartamento,
un negozio o
una piccola
azienda. I
dati trasmessi
tramite il
telefonino
non sono diversi
da quelli inviati



col PC: si tratta sempre di stringhe standard AT, il linguaggio con cui si comunica con i modem; la differenza sostanziale sta nel fatto che, in questo caso, lo facciamo in remoto, dal cellulare, grazie ad un programma in linguaggio Java, di cui forniamo sia il file sorgente che l'eseguibile in modo da poter adattare il programma alle reali esigenze di ciascuno. Ipotizziamo che disponiate della DemoBoard e che il cellulare di cui siete in possesso abbia i requisiti necessari per raggiungere gli obiettivi prefissati.

Il primo passo da affrontare consiste nel preparare il telefonino affinché possa memorizzare il programma scaricabile gratuitamente dal sito della rivista all'indirizzo

www.elettronica.in.it. Per installare il nostro software va utilizzato il pacchetto applicativo per PC fornito assieme al cellulare che consente l'installazione degli applicativi Java. Se non l'avete a portata di mano potete comunque scaricarlo dal sito internet del produttore del telefono.

Per le prove abbiamo utilizzato un cellulare Nokia 6680: con tale dispositivo possiamo fare uso di Nokia PC Suite, un'applicazione che permette di accedere a tutte le funzionalità del telefono da PC nel momento in cui sono reciprocamente connessi. Delle opzioni disponibili, una è fondamentale per ciò che dobbiamo fare: si tratta della procedura per il caricamento dei software "Nokia Application Installer" di cui trovate una videata nella pagina accanto.

Attenendovi alle procedure previste bisogna caricare il programma nella memoria del telefonino.

Abbiamo ritenuto interessante fornire anche il sorgente in quanto le procedure implementate possono essere riutilizzate come fossero i mattoncini fondamentali per realizzare applicativi personalizzati. Per scrivere il nostro programma abbiamo utilizzato un pacchetto di sviluppo Java: Netbeans Mobility Pack. Si tratta di uno dei programmi più validi per scrivere applicativi Java destinati all'impiego "Mobile" ovvero adatti a telefoni cellulari, smartphone, pocket PC, eccetera; oltretutto, è scaricabile gratuitamente all'indirizzo: <http://www.netbeans.org>.

A questo proposito desideriamo sottolineare che il concetto "*sperimentiamo insieme*" è per noi sempre valido. Se quindi qualcuno pensa che si possa fare di meglio e riesce a preparare un software di gestione della DemoBoard più funzionale, non esiti ad inviarci una copia: saremo lieti di pubblicarla e renderla disponibile a tutti i lettori.

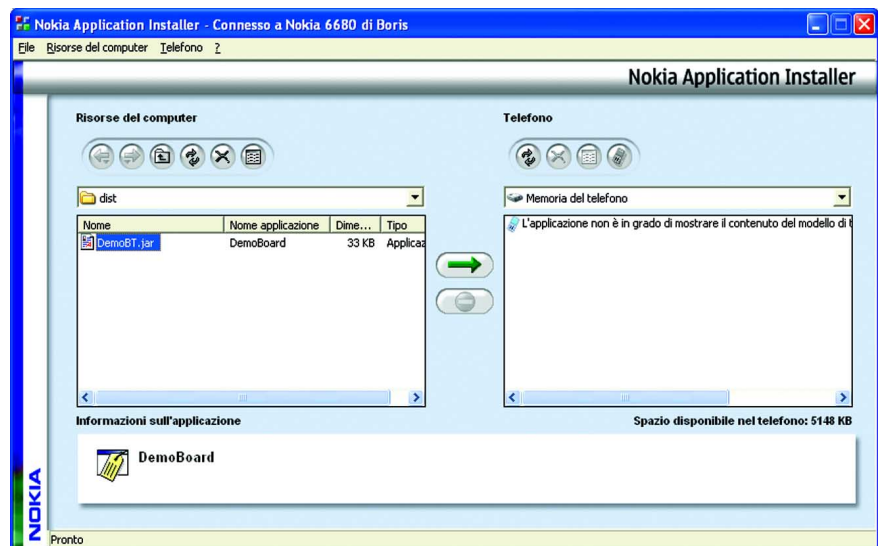


Al centro pagina, la schermata del "Nokia Application Installer", software in dotazione al cellulare col quale abbiamo installato il programma Java per il controllo remoto della Demo Board. In alto a sinistra: schermate iniziali di configurazione; a base pagina, la sequenza dei passaggi necessari perchè il modulo Ezurio Bluetooth possa essere identificato dal telefono cellulare.

Attiviamo il software

Dopo aver caricato il programma, lo mettiamo in esecuzione al fine di verificare che tutto funzioni correttamente e che sul display compaiano i vari menu. In questa prima fase ci limitiamo a controllare se il programma si attiva in quanto non è possibile utilizzare il sistema se prima non provvediamo ad inizializzare la DemoBoard. Sul display del telefonino compaiono infatti solamente i dati del cellulare, vale a dire MAC e nome dell'apparecchio, come appare nell'immagine che trovate in questa stessa pagina.

La procedura di inizializzazione della DemoBoard che consente al modulo Ezurio di essere identificato dal telefonino avviene tramite PC e consiste nell'invio di una serie

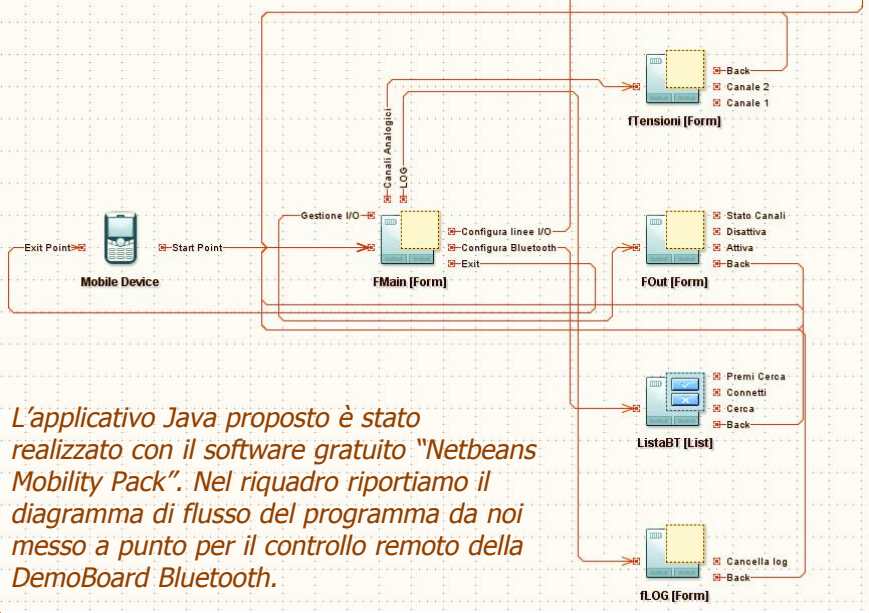


di stringhe di comandi AT con le quali si settano le modalità ed i parametri di funzionamento del modulo Bluetooth. Questa procedura

va effettuata solamente la prima volta in quanto i parametri impostati possono essere memorizzati su EEPROM e pertanto il sistema si ➤



Netbeans Mobility Pack: una soluzione per lo sviluppo di applicazioni in linguaggio Java



L'applicativo Java proposto è stato realizzato con il software gratuito "Netbeans Mobility Pack". Nel riquadro riportiamo il diagramma di flusso del programma da noi messo a punto per il controllo remoto della DemoBoard Bluetooth.

Comandi AT

ATS0=1 (rispondi dopo il primo squillo);

ATS512=4 (rendi il dispositivo identificabile e consenti la connessione);

ATS536=1 (consenti di controllare il modulo tramite i comandi AT da remoto);

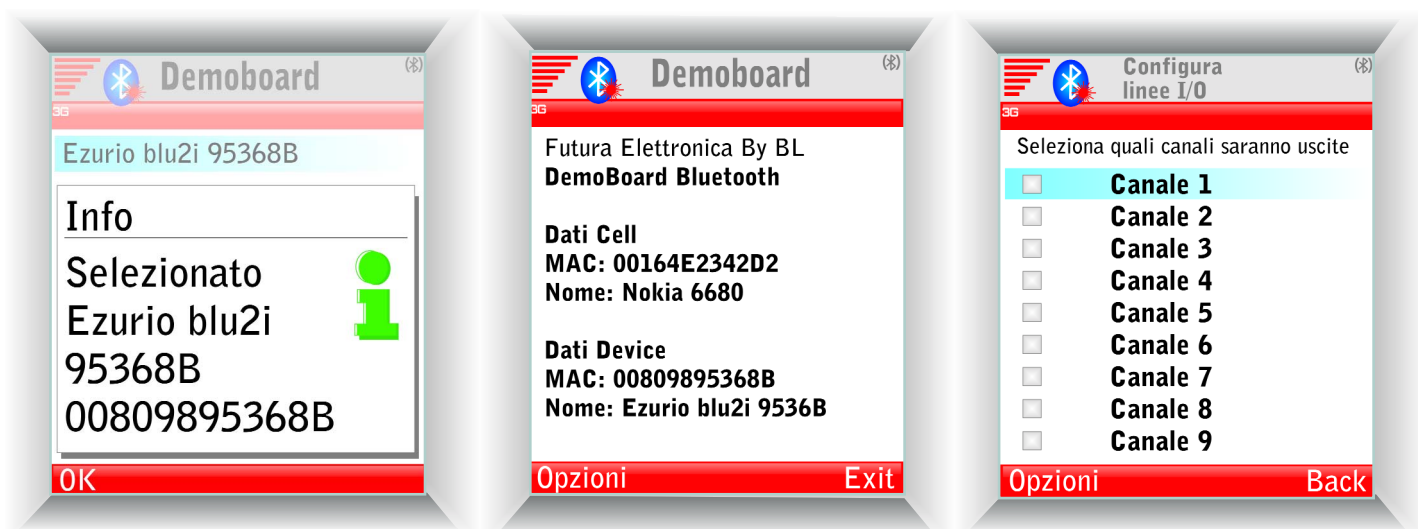
AT&W (salva nella memoria non volatile queste impostazioni);

ATZ (reset del modulo per rendere operativi i precedenti settaggi);

ATI4 (per conoscere l'indirizzo Bluetooth del modulo).

Avendo trasmesso il comando **ATZ** i dati rimangono nella memoria della scheda anche in assenza della tensione di alimentazione.

Scollegiamo dunque la scheda dal PC e proviamo ad utilizzare in



Nella raccolta di immagini del display del telefonino, una sequenza dei punti fondamentali per rendere operativa la DemoBoard e per verificare lo stato dei canali.

In particolare notiamo che cosa si visualizza quando il modulo Ezurio posto sulla DemoBoard risulta connesso e come appare la lista dei canali selezionabili; osservate le check box che confermano lo stato del canale. In particolare vediamo, da sinistra verso destra, la richiesta di informazioni sull'elemento selezionato (Ezurio), la risposta ottenuta e come si presenta l'elenco dei canali su cui impostare la funzionalità I/O.

predisporre nella modalità imposta anche dopo lo spegnimento e la successiva riaccensione della DemoBoard.

Collegiamo dunque la scheda al PC mediante la porta seriale (configurata con 9600 Baud, 8 bit, N1) ed utilizziamo un programma come

Hyper Terminal di Windows, oppure TDK Terminal della Ezurio scaricabile dal sito del produttore all'indirizzo www.ezurio.com.

Riportiamo di seguito le stringhe dei comandi AT da inviare per inizializzare la scheda; ogni passo della procedura è commentato.

modalità stand-alone la DemoBoard cercando di instaurare una connessione in modalità Bluetooth col telefonino.

Come primo passo attiviamo il programma di gestione DemoBoard caricato sul cellulare: il display del telefonino rappresenta ora il nostro ➤



I segni di spunta ai canali 1,3,4,5 e 8 stanno ad indicare che si intende utilizzarli come uscite. Questa impostazione diventa realmente operativa dopo che è stato premuto "invia" del menu "opzioni". Avvalendoci della funzionalità gestione canali controlliamo e determiniamo se la loro uscita è attiva o meno. Allo stesso modo verifichiamo il livello di ingresso dei canali digitali. Anche i due ingressi analogici sono monitorabili. Per questi ultimi possiamo leggere il valore di tensione applicato e visualizzarlo graficamente. Dopo la trasmissione di una stringa di comando riceviamo ogni volta delle risposte e, fra queste, la stringa in codice AT inviata.



ESTRATTO DAL LISTATO IN JAVA

```

public String inviodati(String stringa, Form form) {

    String output = "";
    Date startTime = new Date();
    long begin = startTime.getTime();
    Date actualTime;
    long end = 0;
    int input;

    String connString = "btspp://" + MAC_DEV +
":1;master=false;encrypt=false;authenticate=false";
    if (connString != null)
    {
        try{
            StreamConnection conn =
(StreamConnection) Connector.open(connString);
            OutputStream out =
conn.getOutputStream();
            InputStreamReader in = new
InputStreamReader(conn.openInputStream());
            fLOG.append("Config BT ok\n");

            try {Thread.sleep(1000);}
            catch (InterruptedException ex){ex.printStackTrace();}

            for (int n = 0; n < 3; n++)
            {

                String buff_message = "!";
                out.write(buff_message.getBytes());
                out.flush();
                try
                {
                    Thread.sleep(500);
                } catch (InterruptedException ex)
                {
                    ex.printStackTrace();
                }
            }

            try {Thread.sleep(500);}
            catch (InterruptedException ex){ex.printStackTrace();}
            fLOG.append("Invio !!! ok\n");

            String buff_message="";
            message=stringa;
            buff_message = message + invio;
            out.write(buff_message.getBytes());
            out.flush();
            fLOG.append("Invio dati " + message + "\n");
            try {Thread.sleep(1000);}
            catch (InterruptedException ex){ex.printStackTrace();}
            out.close();

            for (int n = 0; n < 80; n++)
            {
                if((in.ready() == true)
                {
                    input = in.read();
                    output += (char) input;
                    fLOG.append("\nn:" + n + "-in:"+(char) input));
                }
            }
            fLOG.append("\nCaratteri risposta " + output.length());
            fLOG.append("\nRisposta completa(con echo):\n"+output+"\n");

            if (output.length()>=message.length() )
            {
                output = output.substring(1, output.length() - 1);

                output=output.substring(output.indexOf(message)+message.length(),output.length());

                fLOG.append("Mes di risposta:"+output+"\n");

                in.close();
                conn.close();
                fLOG.append("Connessione chiusa\n");

                show_alert ("Mess inviato : " + message + " - Risposta: " + output, 0,-2, form);
            } catch (IOException e){

                show_alert ("Invio al device BT fallito", 1, -2, FMain);
                fLOG.append("Invio al device BT fallito\n");
            }
        }
    }
    return output;
}

```

Listato sorgente della parte di programma che si occupa di inviare le stringhe dati in modalità AT necessarie per la comunicazione fra le unità Bluetooth. Tutti i comandi inviati alla DemoBoard rispettano lo standard AT, cosa che potete verificare visualizzando sul display del telefonino il file Log che contiene la storia delle sequenze di comando attivate.

ESTRATTO DAL LISTATO IN *JAVA*

```
//metodo chiamato appena viene trovato un dispositivo BT
public void deviceDiscovered(RemoteDevice remoteDevice, DeviceClass deviceClass){
    if ((deviceClass.getMajorDeviceClass() == 0x100) || (deviceClass.getMajorDeviceClass() == 0x1F00)) {
        try {
            ListaBT.append(remoteDevice.getFriendlyName(false), null);
        } catch (IOException e){
            ListaBT.append("Unknow", null);
        }
        BTDLst.addElement(remoteDevice.getBluetoothAddress());
    }
}
```

Routine richiamata ogni qualvolta viene identificato un dispositivo Bluetooth. Il programma Java necessario per poter fare dialogare il telefonino con la nostra DemoBoard Bluetooth è disponibile gratuitamente, insieme al listato sorgente, all'indirizzo www.elettronica.in.it.

monitor di controllo. Sulla schermata principale troviamo, come già detto, l'indirizzo MAC ed il nome del cellulare. In queste pagine riportiamo la sequenza visiva di quanto descriviamo, pertanto potrete confrontare le vostre schermate con quelle pubblicate e seguire il flusso delle operazioni facendo riferimento all'apposito diagramma. In questo momento telefono e DemoBoard ancora non dialogano. Premendo sul tasto "opzioni" della schermata iniziale accediamo all'elenco delle operazioni consentite. Fra l'insieme delle voci rese disponibili selezioniamo "Configura Bluetooth" e sul menu successivo confermiamo l'intenzione di cercare una periferica con cui stabilire una connessione. La risposta arriva rapidamente e sul display compare il nome del modulo Ezurio ed eventuali altri dispositivi nel campo di azione. Selezioniamolo e diamo OK. Sulla schermata di avvio ora,

oltre ai dati del cellulare, compaiono anche quelli del modulo Ezurio. Telefono e DemoBoard sono connessi e possono dialogare. Ripartendo dal menu delle scelte operative possiamo occuparci della procedura "configura linee I/O". Sul display compare la lista dei canali con accanto i simboli per inserire il segno di spunta; tramite questo menu potremo stabilire quali canali si dovranno comportare come uscita e quali come ingresso. Per confermare le scelte fatte selezioniamo "invia" dal menu opzioni e la stringa di configurazione viene trasmessa alla DemoBoard. A causa delle limitate dimensioni del display dei telefonini, abbiamo ridotto al minimo la quantità di informazioni di volta in volta visualizzate; pertanto, per confermare le impostazioni selezionate, spesso si deve ricorrere alla funzione "invia", disponibile tra le opzioni. Torniamo ora al menu delle opzioni di confi-

gurazione per occuparci della "gestione I/O". Essa ci consente di verificare lo stato logico dei canali, siano essi ingressi che uscite. Ricordiamo che sulla basetta della DemoBoard sono presenti dei jumper che consentono di assegnare un livello logico alle linee di ingresso/uscita. Ovviamente, nelle linee che imposteremo come uscite, non dovremo montare alcun jumper per evitare un corto circuito (in ogni caso ci sono le resistenze di protezione...); per quanto riguarda gli ingressi, invece, i jumper ci consentiranno di fornire a queste linee dei livelli prestabiliti senza dover ricorrere a segnali esterni. Oltre agli 8 canali I/O digitali sono disponibili due ingressi analogici monitorabili a distanza: selezionandoli, sul display del telefonino leggeremo la tensione presente agli ingressi relativi e osserviamo un grafico a barra orizzontale, una visualizzazione particolarmente efficace.

Per il

MATERIALE

La DemoBoard Bluetooth è disponibile in scatola di montaggio (cod. FT628K) al prezzo di 47,00 Euro. Il kit non comprende il modulo FT622M (costo 94,00 Euro) sul quale è montato il chip Ezurio BISM2. Quest'ultimo è anche disponibile separatamente al prezzo di 78,00 Euro. Il kit della DemoBoard comprende anche il software di gestione. Tutti i prezzi si intendono IVA compresa.

**Il materiale va richiesto a: Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 ~ Fax: 0331-778112 ~ <http://www.futuranet.it>**

circuiti stampati in 24 ore

**garantiamo il tempo di
consegna: 24 ore o i
circuiti sono gratis**

Potrete scegliere tra 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.

Prezzi a partire da* **€ 14,38** (doppia faccia foro metallizzato 7,50x7,50 cm) e da **€ 9,13** (singola faccia 7,50x7,50 cm) per FR4 1,6 mm con rame 35 µm, **tutti comprensivi di attrezzatura.**

Nessuna limitazione sul numero dei fori, sul numero degli utensili (diametri) e sul tipo di scontornatura (anche tondeggiante).

Distanza minima tra le piste e pista minima 8 mils (0,20 mm).

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



**+ QUALITÀ
- TEMPO**

millennium
md
dataware

visita il nostro sito per il dettaglio delle note tecniche

www.mdsrl.it

millennium dataware srl parco scientifico e tecnologico

15050 rivalta scrivia - tortona (al)

tel. 0131 860.254

fax 0131 860157

info@mdsrl.it

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

Piove e tira vento: metti la tenda sotto controllo

di Matteo Destro

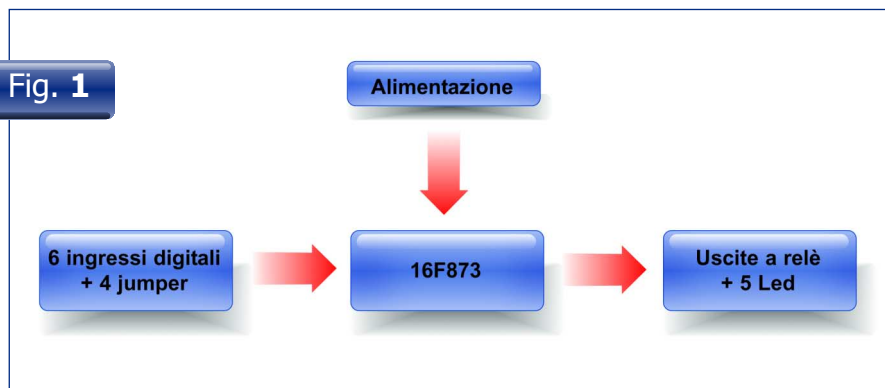
Utilizziamo la logica di programmazione a stati, tipica delle FSM (macchine a stati finiti) per realizzare un modulo a microcontrollore in grado di azionare in maniera automatica tende e tapparelle motorizzate. Possibilità di collegare più moduli in cascata.



Non avere più bisogno di azionare manualmente l'avvolgibile di una tapparella o aprire e chiudere un tenda motorizzata da esterno significa prestare la giusta considerazione agli eventi quotidiani che influiscono sulla qualità della nostra vita. Quando nella casa vivono anziani o disabili, capita spesso che essi debbano chiedere aiuto per posizionare la tapparella in modo adeguato alle necessità e, quando l'aiuto non è disponibile, siano costretti a rassegnarsi al tutto su o tutto giù. Che dire poi delle tende da sole ad azionamento manuale: aprirle e chiuderle diventa un esercizio fisico non

sempre adeguato alle forze dell'occasionale "atleta". Ecco dunque la nostra proposta: una scheda a microcontrollore progettata per gestire due uscite (in pratica due motori elettrici) da utilizzare per azionare tutte le tapparelle e le tende motorizzate di casa. Se gli elementi da controllare sono più di due non c'è alcun problema: potremo utilizzare più schede collegate in cascata fino ad un massimo di sedici. Un tocco di pulsante e tutti gli avvolgibili iniziano a lavorare in sequenza per fare entrare luce in casa o per chiudere le finestre. Questo ci può dare, ad esempio, la certezza ➤

Fig. 1



che tutti gli accessi siano chiusi prima di inserire il sistema antifurto. Ma quanto proposto è solo la base per un sistema domotico: possiamo anche, con i giusti abbinamenti, fare in modo che tende da sole o tapparelle vengano attivate mediante il segnale generato da un sensore di pioggia e vento; si veda a tale proposito il progetto (FT549K) proposto sul numero 89.

Il circuito è stato progettato in modo tale da poter funzionare senza problemi con tutti gli automatismi motorizzati disponibili in commercio. I motori vengono attivati tramite relè, due per canale. Questo perchè, ovviamente, ogni motore deve poter ruotare in un verso oppure in quello opposto e deve poter essere acceso e spento.

Struttura del modulo

Un singolo modulo di controllo è in grado di gestire, in modalità indipendente, due motori per la movi-

mentazione di tende/tapparelle; quindi l'azionamento della tenda/tapparella A è assolutamente separato da quello B e viceversa. L'interfaccia utente è costituita da quattro pulsanti e due interruttori. Ad ogni motore è assegnata una coppia di pulsanti che assolvono al compito "su/giù" mentre gli interruttori - che identifichiamo con le sigle UPC (tutto su) e DWC (tutto giù) - vengono usati per controllare in simultanea i due motori in modalità tutto aperto o tutto chiuso. Qualora vengano collegati più moduli fra di loro per azionare svariate coppie di motori, ogni modulo disporrà sempre di quattro pulsanti e due interruttori: i pulsanti agiscono sui motori collegati al singolo modulo, mentre gli interruttori sono sempre fra loro in parallelo.

Se desideriamo il "tutto su", basta agire sull'interruttore abbinato a questa funzionalità di una qualsiasi scheda che fa parte del sistema. Il

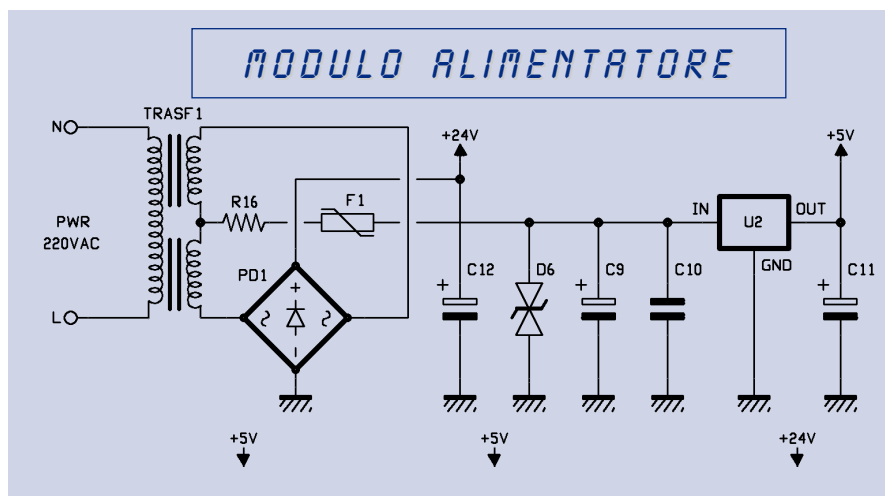
sistema di automazione tende/tapparelle consente di collegare fino a sedici moduli, pertanto l'impianto è in grado di pilotare 32 motori. Pur essendo collegate fra di loro in parallelo, le schede lavorano in cascata, una dopo l'altra. La sezione di potenza del progetto non avrebbe alcuna difficoltà ad azionare simultaneamente tutti i motori; il problema, caso mai, risiede nell'impianto elettrico di casa: 32 motori elettrici attivati simultaneamente determinano una corrente di spunto probabilmente eccessiva.

Per evitare questa eventualità ogni scheda dispone di quattro jumper che consentono di impostare un indirizzo (da 0 a 15) al fine di attivare in sequenza i motori con un intervallo predefinito.

Ogni scheda può essere tarata in funzione del tempo che richiede la singola tenda o tapparella; quanto serve per la calibrazione è presente sulla stessa basetta: un led indica quando la taratura è in corso ed altri quattro segnalano lo stato dei pulsanti di azionamento. Sul circuito stampato è anche presente un buzzer con lo scopo di segnalare acusticamente l'attivazione della procedura di taratura.

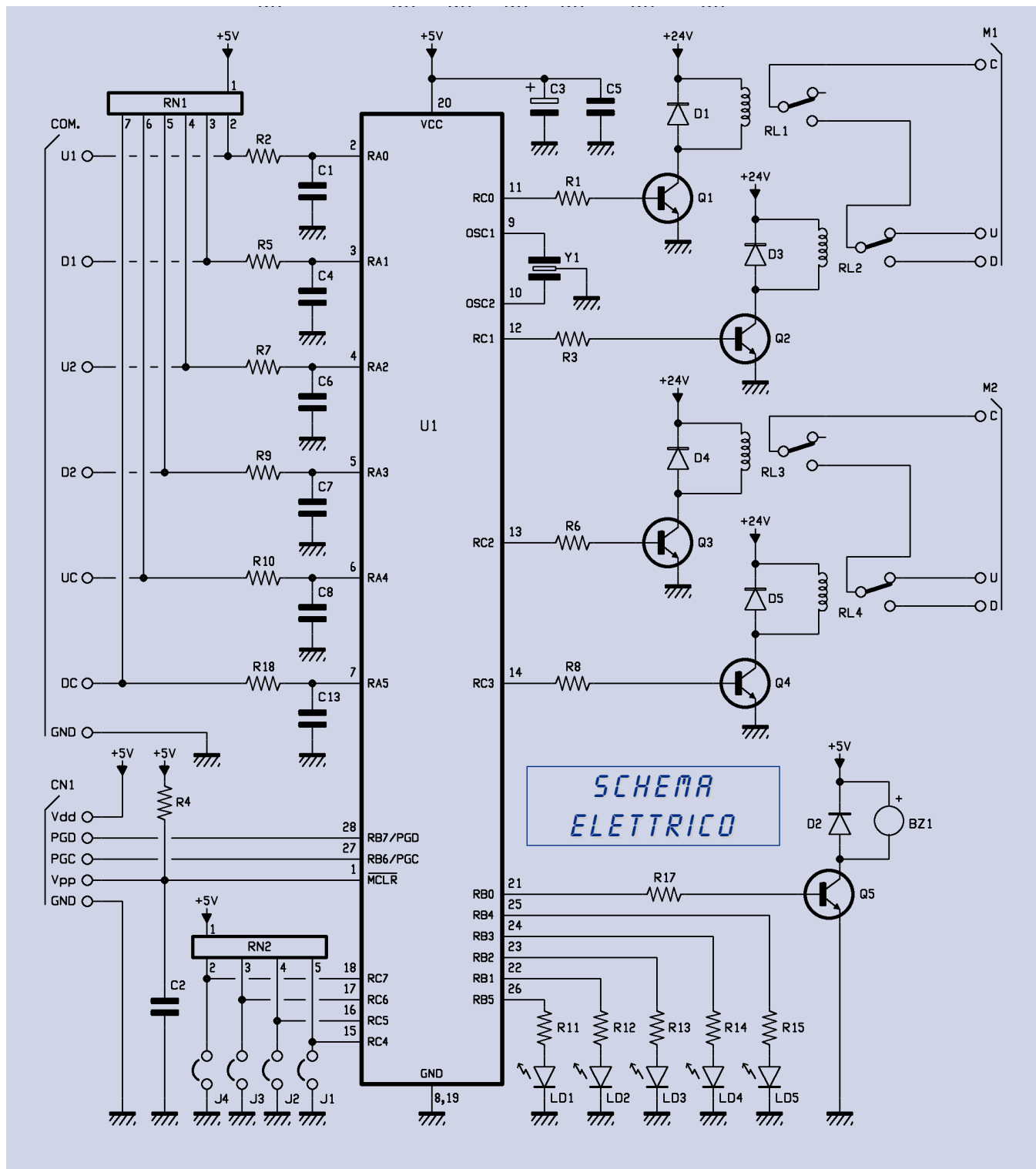
Analisi del circuito

Lo schema a blocchi di Fig. 1 evidenzia il principio di funzionamento del nostro modulo di controllo.



In alto, lo schema a blocchi del modulo progettato per il controllo di tende e tapparelle.

Il cuore del dispositivo è un PIC16F873 per il quale rendiamo disponibile gratuitamente (scaricabile dal sito della rivista) l'eseguibile ed il sorgente del firmware. A lato, circuito elettrico della sezione di alimentazione, e nella pagina accanto, schema elettrico generale del modulo di controllo con evidenziati i quattro relè di potenza (due per canale).



Le parti che lo compongono sono veramente poche: alimentazione, ingressi con jumper di configurazione, microcontrollore, uscita a relè con elementi di segnalazione. Occupiamoci del primo blocco: l'alimentazione. Il trasformatore T1 riduce la tensione di rete a 9Vca su entrambi i secondari, infatti il

modello di cui ci siamo avvalsi è realizzato con due secondari identici. Questa scelta è stata determinata dal fatto che il circuito richiede due livelli di alimentazione. Collegando in serie i due secondari ed utilizzando il ponte a diodi si ottengono i +24Vdc necessari per alimentare le bobine dei relè inseriti sulla scheda.

Utilizzando invece un solo secondario del trasformatore si ricavano i +5Vdc necessari ad alimentare la "logica" presente sulla scheda. A monte del regolatore di tensione 7805 troviamo un circuito per la soppressione delle sovratensioni che interviene quando la tensione supera i 33V. In questo caso D7 (un ➤

- Alimentazione scheda: 230 Vca;
- Assorbimento corrente a riposo: 27 mA;
- Assorbimento corrente a pieno carico: 29 mA;
- Corrente nominale di lavoro relè: 6 A;
- Corrente massima istantanea relè: 10 A;
- Tensione nominale di lavoro relè: 230 Vca;
- Tensione massima commutabile relè: 400 Vca;
- Tensione di alimentazione bobina relè: 24 Vdc.

transzorb bidirezionale), cortocircuiterà verso massa facendo intervenire il termo fusibile F1 con la conseguente disattivazione del circuito di alimentazione. La protezione interviene anche in caso di eccessivo assorbimento di corrente, infatti il termo fusibile ha una soglia di intervento calibrata a 100mA. Prendiamo adesso in esame la sezione degli ingressi. Essa consiste di due parti: la prima gestisce i pulsanti di attivazione della scheda, la seconda si occupa dei quattro jumper per la configurazione dell'indirizzo del modulo in uso. I pulsanti sono collegati alla porta A del PIC, attraverso un circuito RC, in modo da proteggere gli ingressi dello stesso da eventuali disturbi e da eccessivi assorbimenti di corrente. La rete resistiva RN1 assicura una tensione di +5 Vdc stabile all'ingresso del PIC nel periodo in cui i pulsanti sono inattivi. Non appena viene premuto un pulsante, sull'ingresso del PIC la tensione va a 0 Vdc (massa). Quanto detto vale per tutti i sei attuatori. I jumper sono collegati con la medesima logica degli ingressi appena descritti, la differenza consiste nel fatto che non è necessaria la rete RC, in quanto i jumper si trovano sulla stessa scheda e quindi i relativi ingressi del PIC sulla porta B non sono soggetti a disturbi esterni. La sezione di uscita prevede quattro relè per gestire i due motori, questo significa, come abbiamo preceden-

temente accennato, che per ogni motore sono utilizzati due relè. Ad esempio, prendendo in considerazione RL1 e RL2 riscontriamo che RL2 imposta la direzione di rotazione del motore (quindi, nel nostro caso, se la tenda/tapparella sale oppure scende); mentre RL1 dà il consenso alla rotazione del motore. Sulla basetta sono presenti cinque led, connessi a catodo comune. Essi forniscono un'indicazione visiva dello stato raggiunto dalla scheda durante il normale funzionamento. Il led LD1, abbinato al buzzer, indica che la scheda è entrata in modalità di taratura. Gli altri led servono per indicare in quale direzione (su/giù) si stanno muovendo i motori 1 e 2. Mediante connettore CN1 è possibile effettuare la programmazione ed il debug in-circuit del micro

PIC16F873; tale operazione può essere eseguita utilizzando l'ICD2 della Microchip oppure un sistema di sviluppo equivalente.

Il microcontrollore può ovviamente essere programmato in maniera tradizionale, prima del montaggio sulla basetta. Il firmware necessario per questa applicazione può essere scaricato gratuitamente dal nostro sito sia in modalità sorgente che eseguibile.

Per quanto riguarda i collegamenti ai motori, rimandiamo ai disegni pubblicati che evidenziano con precisione i cablaggi necessari; raccomandiamo particolare attenzione in quanto i motori lavorano alla tensione di rete. L'indicazione U sta per up (su) e D per down (giù).

Firmware e taratura

Come possiamo notare, lo schema elettrico è molto semplice ed utilizza un numero esiguo di componenti, infatti le funzioni della scheda vengono gestite interamente dal PIC16F873. Per migliorare la leggibilità e rendere più agevole la stesura del codice abbiamo deciso di suddividere il firmware in più file. Il firmware basa il suo funzionamento su una "macchina a stati", che in seguito chiameremo "FSM", per la gestione delle varie fasi di movimentazione dei motori. La rea-

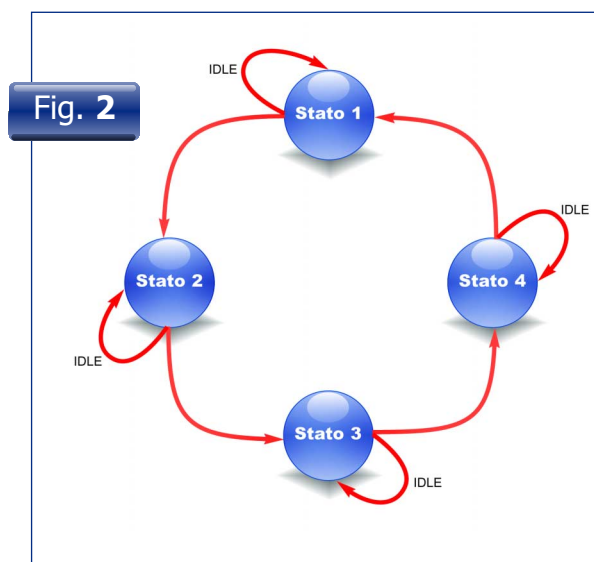


Fig. 2

A sinistra, schema di principio di una FSM (macchina a stati finiti) a quattro stati. Per ognuno di essi esiste una condizione di IDLE che pone il sistema in attesa dell'input per effettuare il cambio di stato. Nella pagina a destra in alto, la struttura della FSM implementata nel firmware.

lizzazione di una “FSM” prevede un insieme di stati finiti e le relative direzioni di evoluzione degli stessi. Le direzioni della “FSM” vengono definite in base alle “funzioni di stato futuro” le quali vengono determinate in base agli ingressi e allo stato corrente. Nella maggioranza dei casi si dà una rappresentazione grafica della “FSM” detta “grafo di transizione degli stati” (Fig. 2). Come si può osservare nel disegno, è stata

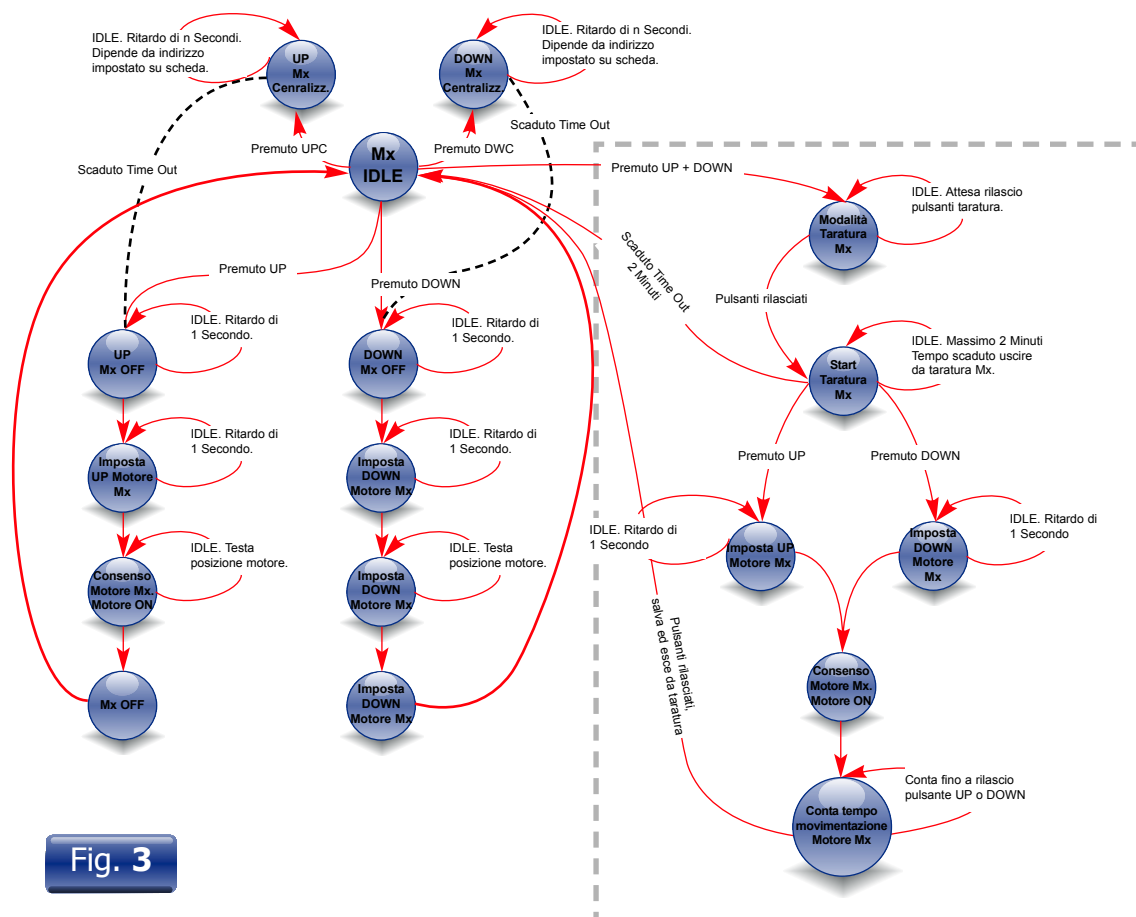


Fig. 3

creata una “FSM” con quattro soli stati. Il sistema può passare da uno stato all’altro in sequenza, quindi da stato 1 a stato 2, da stato 2 a stato 3 ecc. senza la possibilità di tornare indietro. Quando la “FSM” si posiziona su un nuovo stato resta in attesa di un evento che scateni il passaggio allo stato successivo (Condizione di “IDLE”).

La “FSM” che gestisce il sistema di movimentazione tende/tapparelle si basa su questa filosofia, cioè passa da uno stato al successivo in occasione di determinati eventi (variazione sugli ingressi, timer, ecc), in caso contrario rimane in attesa. Questo approccio permette di realizzare software complessi, soprattutto dove più task devono girare simultaneamente per soddisfare tutte le richieste del sistema, ad esempio la gestione di due seriali, la lettura da una RAM/FLASH, la gestione di un display eccetera: nel nostro caso gestire due tende/tappa-

relle contemporaneamente e indipendentemente l’una dall’altra.

Prendiamo ora in considerazione la Figura 3, quella in cui appare la macchina a stati utilizzata per realizzare il firmware del controllo tende/tapparelle. La rappresentazione grafica è costituita da due parti: una che descrive il funzionamento del controllo in situazione d’uso e l’altra (racchiusa nel rettangolo tratteggiato) durante la taratura. Le due strutture hanno un punto in comune: lo stato “MX IDLE” dove il sistema attende una variazione sugli ingressi per passare dallo stato presente a quello futuro. Come si può osservare il motore viene chiamato Mx, questo perché esistono due “FSM” all’interno del firmware, una che gestisce il motore M1 (tenda/tapparella “A”) e l’altra il motore M2 (tenda/tapparella “B”). Le due “FSM” sono indipendenti e possono lavorare contemporaneamente senza alcun problema.

Quindi per la descrizione della stessa si è generalizzato il nome del motore con Mx.

Descriviamo ora la taratura della scheda occupandoci, ad esempio, del motore M1 (tenda/tapparella “A”). Per entrare in modalità di taratura è necessario mantenere premuti per più di 2 secondi i pulsanti UP (tenda/tapparella “A”) e DOWN (tenda/tapparella “A”); quando la procedura di taratura è attiva si accende LD1; rilasciando i due pulsanti si passa allo stato successivo, dove il sistema rimane in attesa della pressione del pulsante UP (tenda/tapparella “A”) o DOWN (tenda/tapparella “A”). Se nessun pulsante viene premuto entro 2 minuti la procedura di taratura viene abortita e si ritorna nella condizione di “Mx IDLE”.

Quando si deve tarare la scheda è necessario che la tenda/tapparella sia in una posizione adeguata allo scopo, o “tutta su” o “tutta giù”. Se ➤

dovesse capitare di fare una taratura con tenda/tapparella a metà corsa si consiglia di entrare in taratura, decidere se muovere la tenda/tapparella verso l'alto o il basso e portarla in adeguata posizione, dopodiché ripetere da capo la procedura di taratura. Supponendo che la tenda/tapparella sia "tutta giù", premete il pulsante UP (tenda/tapparella "A") e mantenetelo premuto; così facendo si passa al nuovo stato in cui si impone la direzione del motore agendo sul relè RL2. Il led rosso LD2 si accenderà ad indicare la direzione impostata. Il passaggio

Tabella 1

Ritardo	J1	J2	J3	J4
0 Secondi	0	0	0	0
1 Secondi	1	0	0	0
2 Secondi	0	1	0	0
3 Secondi	1	1	0	0
4 Secondi	0	0	1	0
5 Secondi	1	0	1	0
6 Secondi	0	1	1	0
7 Secondi	1	1	1	0
8 Secondi	0	0	0	1
9 Secondi	1	0	0	1
10 Secondi	0	1	0	1
11 Secondi	1	1	0	1
12 Secondi	0	0	1	1
13 Secondi	1	0	1	1
14 Secondi	0	1	1	1
15 Secondi	1	1	1	1

allo stato successivo e quindi l'attivazione del motore con il consenso (relè RL1) viene eseguito solo dopo un ritardo di 1 secondo. Il ritardo è necessario per assicurarsi che i rimbalzi del relè dovuti alla commutazione siano completamente esauriti. Il motore non verrà fermato fintantoché risulterà premuto il pulsante UP (tenda/tapparella "A"). Durante questa fase il sistema conta il tempo necessario ad eseguire la corsa completa da tenda/tapparella "tutta giù" a tenda/tapparella "tutta su".

Rilasciato il pulsante verrà fermato il motore e quindi memorizzati i parametri di tempo per la movimentazione della tenda/tapparella "A" collegata al motore 1: i led LD1 e LD2 verranno spenti. Se, invece, la tenda/tapparella è "tutta su" bisogna premere il pulsante DOWN (tenda/tapparella "A"). Durante il normale funzionamento la "FSM" attende la pressione del pulsante UP o DOWN per muovere rispettivamente la tenda/tapparella verso l'alto o verso il basso. Supponendo di voler muovere la tenda/tapparella verso il basso pre-

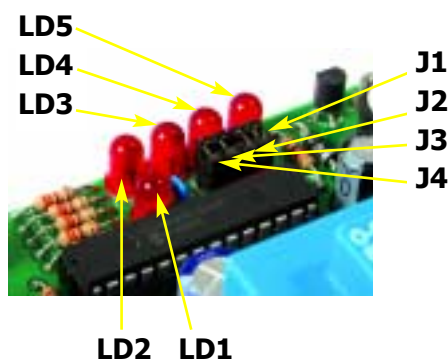


Tabella 2

	Ritardo	J1	J2	J3	J4
SCH 1	0 Secondi	ON	ON	ON	ON
SCH 2	1 Secondi	OFF	ON	ON	ON
SCH 3	2 Secondi	ON	OFF	ON	ON

mendo DOWN, la "FSM" si posiziona sullo stato "DOWN Mx OFF". Per prima cosa il motore viene spento, questo perché il comando DOWN potrebbe essere stato impartito mentre la tenda/tapparella stava salendo, quindi per non danneggiare il motore bisogna prima di tutto spegnerlo, invertire il senso di marcia e poi accenderlo nuovamente. Scaduto il ritardo di 1 secondo, la macchina passa allo stato successivo e imposta la direzione di movimento della

tenda/tapparella, in questo caso DOWN, agendo sul relè RL2. Per evidenziare la selezione viene acceso il led LD3. Anche in questo caso, dopo un'attesa di 1 secondo, la macchina passa allo stato successivo e dà consenso al movimento della tenda/tapparella agendo sul relè RL1. A questo punto la "FSM" rimane in attesa. Il firmware prevede che venga fermato alla pressione del tasto DOWN, oppure se la tenda/tapparella è arrivata a fine corsa. A questo punto la "FSM" ritorna nello stato "Mx IDLE". Naturalmente tutto quello che è stato detto vale anche per la movimentazione della tenda/tapparella verso l'alto. Nella descrizione del progetto abbiamo scritto che esistono due comandi di UP e DOWN centralizzati. Osservando la "FSM" si nota che premendo, ad esempio, DWC (DOWN Centralizzato) viene richiamato uno stato intermedio dove il sistema verifica i jumper in ingresso e decide di quanto ritardare la movimentazione delle due tende/tapparelle collegate alla scheda. I jumper permettono di ottenere sedici combinazioni differenti per un ritardo che va da 0 secondi a un massimo di 15 secondi (vedi tabella 1). Il ritardo introdotto dall'impostazione dei jumper serve per evitare che possano partire contemporaneamente più di due tende/tapparelle alla volta. L'esempio riportato in tabella 2 chiarisce meglio il concetto. Supponiamo di dover pilotare sei tende/tapparelle utilizzando tre schede. Ad ognuna di esse viene collegato il comando centralizzato UPC e DWC. Con queste impostazioni premendo, ad esempio UPC, avremo che la scheda 1 attiverà immediatamente le sue due tende/tapparelle, la scheda 2 le attiverà dopo un ritardo di 1 secondo mentre la scheda 3 dopo un ritardo di 2 secondi.

Naturalmente nessuno vieta di utilizzare due schede con il medesimo

File di configurazione

Define.inc	definizioni delle costanti utilizzate nel firmware
Variabili.inc	definizioni di variabili utilizzate nel firmware
Stati.inc	due macro per la gestione della "macchina a stati"
Macro.inc	macro di uso generale

File di codice programma

Mod_Tende_10.asm	file principale, contiene il codice del main menu e la gestione degli interrupt, inoltre collega tra loro tutti i restanti file .asm e .inc
EEPROM.asm	inizializzazione della EEPROM del PIC
Inizializzazioni.inc	codice di inizializzazione del firmware
InOut.asm	routine di gestione degli ingressi e delle uscite
Gestione_Tende.asm	file contenente la macchina a stati per gestire la movimentazione delle tende
Setup_Tende.asm	file contenente il codice per il setup della scheda

indirizzo (quindi impiegare anche più di 16 schede), unica accortezza, è assicurarsi che all'avvio i motori non determinino una corrente di spunto tale da far scattare il contatore. Si consiglia comunque di optare sempre per soluzioni che evitino il sovraccarico alla linea elettrica.

Realizzare una FSM

A questo punto vi starete sicuramente chiedendo come si possa fare per realizzare una "FSM". Il concetto è molto semplice: sono sufficienti due macro, una che imposta

LISTATO 1

```
SetState MACRO State,NewState
    movlw HIGH(NewState)
    movwf (State)
    movlw LOW(NewState)
    movwf (State+1)
```

lo stato futuro e una che lo esegue. Le due macro si chiamano rispettivamente: *SetState* e *Process*.

Osservando il Listato 1 si nota che in rosso è evidenziato il nome (*SetState*), in blu la parola riservata per creare la macro (MACRO) e in verde gli argomenti che vengono passati alla stessa. I due argomenti puntano, il primo a una variabile memorizzata nella RAM del PIC, l'altro all'indirizzo di memoria FLASH dell'etichetta che dà il

LISTATO 2

```
Process MACRO State
    LOCAL PCALL
    LOCAL PCLOAD
    goto PCALL
PCLOAD movfw(State)
    movwf PCLATH
    movfw (State+1)
    movwf PCL
    call PCLOAD
PCALL ENDM
```

nome al nuovo stato. Come si può osservare le quattro righe di codice non fanno altro che leggere l'indirizzo di memoria in FLASH puntato dall'etichetta, stato futuro (*NewState*), e passarlo alla variabile in RAM (*State*).

Anche per il Listato 2 il nome attribuito è evidenziato in rosso, le parole riservate in blu, le etichette in azzurro e in verde l'argomento che viene passato alla macro. Si noti che l'argomento è lo stesso che abbiamo precaricato nella precedente macro. La parola chiave LOCAL indica che le due etichette

LISTATO 3

```
CBLOCK 0x20
    Stato_M1_MSB
    Stato_M1_LSB
    Stato_M2_MSB
    Stato_M2_LSB
    Stato_BUZZER_MSB
    Stato_BUZZER_LSB
ENDC
```

utilizzate sono in ambito locale e nella sola macro. Leggendo il codice osserviamo che la prima istruzione è un "GOTO" forzato all'etichetta PCALL per eseguire una "call" all'etichetta PCLOAD. A questo punto viene caricato nel PC (Program Counter) il valore del puntatore in FLASH memorizzato nella variabile ricavata dall'argomento *State*. Prima viene caricata la parte più significativa e poi la parte meno significativa. Dopo aver immesso il valore del PC avremo un salto allo stato futuro dove verrà eseguito il relativo codice. L'istruzione "return" associata alla "call PCLOAD" si trova alla fine della routine dello stato futuro.

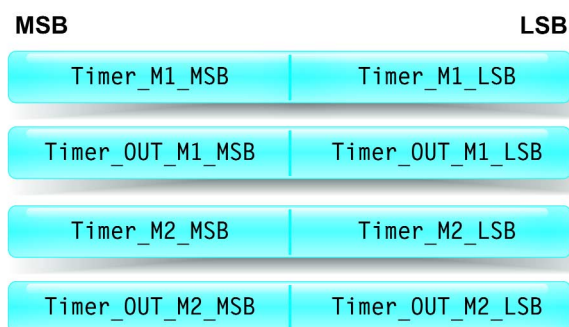
Come detto in precedenza l'argomento *State* punta a una variabile in RAM. Poniamo ora l'attenzione al Listato 3. Nel firmware sono presenti tre macchine a stati differenti, per ognuna di esse è necessario avere una variabile a 16 bit per puntare alla memoria FLASH del microcontrollore. In un microcontrollore a 8 bit una variabile a 16 bit viene rappresentata da due variabili

LISTATO 4

```
Main
.....
Process Stato_M1_MSB
FlashB0
Process Stato_M2_MSB
FlashB0
Process Stato_BUZZER_MSB
FlashB0
goto Main
```

consecutive a 8 bit. Pertanto quando si dichiara una variabile a 16 bit bisogna ricordarsi di dichiarare prima la parte alta (MSB) e poi la parte bassa (LSB) della variabile. Per evidenziare la differenza si è deciso di utilizzare i suffissi "MSB" e "LSB". Le variabili sono state dichiarate nel banco 0 di RAM come evidenziato dalla parola chiave CBLOCK 0x20. La parola chiave CBLOCK serve per allocare spazio in memoria RAM. A questo ➤

Fig. 4



La prima variabile contiene il valore istantaneo di tempo "t" mentre la seconda contiene il tempo massimo di attivazione del motore; le prime due righe riguardano M1, le seconde M2.

LISTATO 5

```
BUZZER_STATO_2
movf    Delay_Buzzer,W
skpz
return
movlw   K_Slow_Delay_Buzzer
movwf   Delay_Buzzer
bcf     BUZZER
SetState Stato_BUZZER_MSB,BUZZER_STATO_3
return
```

punto possiamo mostrare come vengono utilizzate all'interno del codice le due macro sopra descritte. Le chiamate alla macro *Process* vengono fatte esclusivamente nel *Main*, quindi solo nel *Main* verranno richiamati gli stati futuri. Come si può osservare alla macro viene passata la parte più significativa della variabile che contiene il puntatore alla memoria FLASH in cui si trova lo stato futuro, la parte meno significativa viene letta direttamente dalla macro in quanto è allocata alla posizione successiva alla parte più significativa. Nel Listato 4 sono mostrate le tre chiamate, una per ogni macchina a stati

LISTATO 6

```
BUZZER_STATO_3
movf    Delay_Buzzer,W
skpz
return
bcf     BUZZER
movlw   0x06
movwf   Count_Buzzer
clrf    Delay_Buzzer
SetState Stato_BUZZER_MSB,BUZZER_START
return
```

LISTATO 7

```
Timer0_Interrupt
movlw   0x05
movwf   TMR0
.....
movf    PRE_Delay_100mSec,W
skpz
decf    PRE_Delay_100mSec
.....
movf    PRE_Delay_100mSec,W
skpz
goto    NoDelay

movlw   K_PRE_Delay_100mSec
movwf   PRE_Delay_100mSec
.....
NoDelay
.....
```

presente nel firmware. Il restante codice presente nel *Main* è stato omissso perché non significativo per l'esempio.

Le chiamate alla macro in cui si imposta lo stato futuro sono invece distribuite nel codice in opportuni punti, soprattutto dopo il verificarsi di determinati eventi. Come si può notare la macro per impostare lo stato futuro (Listato 5) viene eseguita solo e soltanto se la variabile *Delay_Buzzer* è

uguale a 0 altrimenti torna al *Main*. L'etichetta *BUZZER_STATO_3* identifica il nome dello stato futuro. All'etichetta *BUZZER_STATO_3* è associato un indirizzo di memoria FLASH, tale valore viene posto in memoria nella variabile dello *Stato_Buzzer_MSB* (Listato 6). Ritornati nel *Main* verrà quindi eseguita la macro *Process*, descritta precedentemente, la quale salterà al nuovo stato. Anche in questo caso per eseguire la macro *SetState* è necessario che la variabile *Delay_Buzzer* vada a zero altrimenti ritorna al *Main*. Come al punto precedente la macro imposterà lo stato futuro, in questo caso il nome dello stato futuro è *BUZZER_START*.

Compresa la dinamica di gestione della "FSM" passiamo alla descrizione di come viene gestita la movimentazione della tenda/tapparella. Il fattore che determina la conclusione dell'apertura o chiusura di una tenda/tapparella è il "tempo", infatti la tenda/tapparella o meglio il motore impiegherà un tempo *t* per portare la tenda/tapparella dalla posizione "tutta su" alla posizione "tutta giù" e viceversa. Quindi la scheda deve conoscere quanto tempo occorre per effettuare la manovra, questo vale sia per la

LISTATO 8

```
btfss   Flag_CYCLE_M1_DOWN
goto    No_M1_INC_Time
cpl16   Timer_M1_MSB, Timer_M1_LSB, 0xFFFF
je      No_M1_INC_Time
movlw   0x01
addwf   Timer_M1_LSB
btfsc   _C
incf    Timer_M1_MSB
No_M1_INC_Time

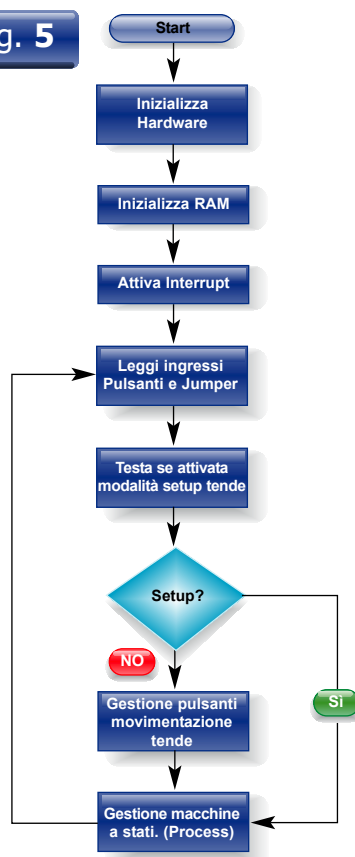
btfss   Flag_CYCLE_M1_UP
goto    No_M1_DEC_Time
cpl16   Timer_M1_MSB, Timer_M1_LSB, 0x0000
je      No_M1_DEC_Time
movlw   0x01
subwf   Timer_M1_LSB
btfss   _C
decf    Timer_M1_MSB
No_M1_DEC_Time
```

tenda/tapparella “A” che per quella “B”. La condizione viene soddisfatta dalla procedura di setup della scheda descritta precedentemente. Una volta acquisito il tempo necessario per la corsa delle tende/tapparelle il sistema è sempre in grado di determinare dove si trova e quindi sa anche quanto tempo è rimasto per concludere la corsa verso l’alto o il basso. La risoluzione per le costanti di tempo è dell’ordine dei 100 millisecondi, inoltre le variabili a 16 bit permettono di gestire un tempo massimo di 109 minuti e 13 secondi. Nella memoria RAM ci sono due variabili a 16 bit che contengono il massimo valore di tempo permesso per la movimentazione delle tende/tapparelle “A” e “B” (dato che viene acquisito al setup) e due variabili a 16 bit per la movimentazione istantanea delle tende/tapparelle “A” e “B” (vedi Figura 4). L’incremento o il decremento della variabile di tempo (`Timer_M1_MSB` `Timer_M1_LSB`) per la movimentazione delle tende/tapparelle viene gestita dalla routine di interrupt del `TIMER0` del PIC. Il `TIMER0` è impostato per generare un interrupt ogni 1 msec.

Per ottenere una base dei tempi di 100 msec è necessario “prescalare” l’interrupt di 1 msec. Per farlo si realizza un prescaler software che è una variabile decrementata fino a zero: quando la variabile è uguale a zero significa che sono trascorsi 100 msec. A questo punto si esegue il codice associato al prescaler software ricordandosi di ricaricare la variabile (valore da assegnare 100 in decimale). Per capire meglio vediamo una parte del codice della gestione dell’interrupt `TIMER0` (Listato 7).

Come si può osservare abbiamo una variabile impostata a 8 bit, `PRE_Delay_100mSec`, che realizza un prescaler software di 100msec. La variabile viene decrementata

Fig. 5



Oltre alla macchina a stati, il firmware segue il flusso dello schema a blocchi riportato a lato, tra l'altro molto semplice e intuitivo. Il processo parte con l'inizializzazione della scheda ovvero con la configurazione delle porte A, B e C del PIC (Input/Output) ed eventualmente si impostano le uscite a valori noti. Successivamente si inizializza la memoria RAM che consiste nell'azzeramento della memoria utente per caricare in seguito dei valori noti nelle variabili di interesse, compresa la lettura in EEPROM dei valori di massimo e minimo per la movimentazione della tenda/tapparella e la posizione attuale della stessa. Infine si procede con l'attivazione degli interrupt. Conclusa l'inizializzazione si passa al "main" dove, ciclicamente, vengono eseguite una serie di istruzioni, comprese le macro per richiamare lo stato futuro della "FSM". L'attività principale all'interno del main è la gestione degli ingressi: a seconda dei valori letti vengono eseguite o richiamate parti di codice differente. Ad esempio, se si sta eseguendo il setup della scheda, non è possibile eseguire la parte di codice che si occupa della gestione pulsanti per la movimentazione delle tende/tapparelle.

fintanto che non va a zero (la variabile viene precaricata con il valore 100, in decimale, equivalente a 100 msec). Il `TIMER0` genera un interrupt ogni 1 msec e quindi moltiplicato per 100 dà come risultato 100 msec; a questo punto verrà eseguito il codice evidenziato in rosso. Ora viene ricaricata la variabile dei 100msec, poi vengono eseguite tutte le istruzioni sottostanti. Al prossimo interrupt del `TIMER0` questa routine non verrà eseguita perché la variabile `PRE_Delay_100mSec` non è più a zero. All’interno della zona evidenziata in rosso, dove la maggior parte delle istruzioni è omessa, si trova il codice per gestire l’incremento o il decremento delle variabili temporali per la movimentazione delle tende. Osserviamo il codice evidenziato in rosso del riquadro successivo, Listato 8: esso gestisce l’incremento o il decremento della variabile temporale del motore 1

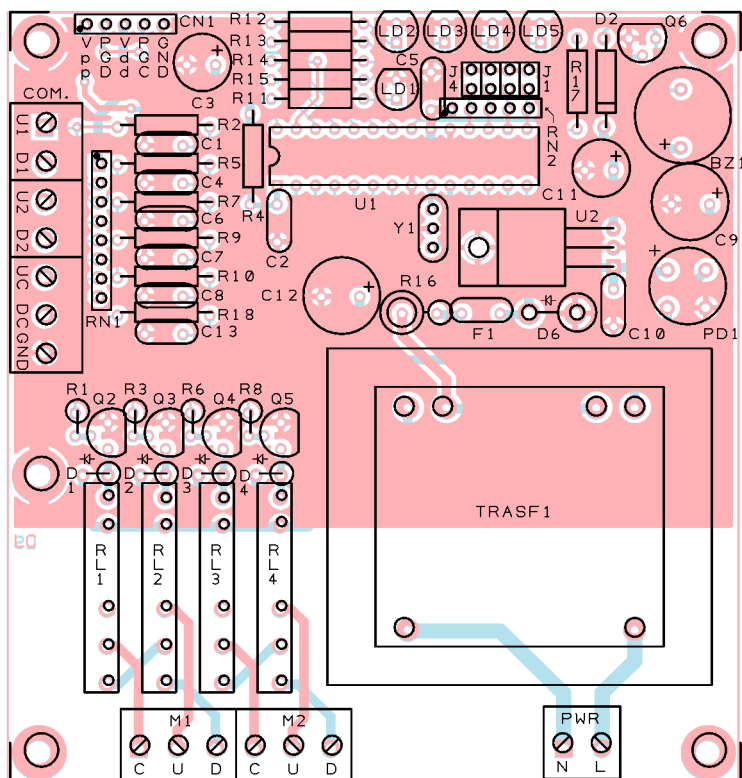
tenda/tapparella “A”. Se il Flag `Flag_CYCLE_M1_DOWN` è pari a “1” allora stiamo eseguendo un’operazione di DOWN tenda/tapparella “A” quindi la variabile temporale va incrementata, se invece è il flag `Flag_CYCLE_M1_UP` a valere “1” stiamo eseguendo un’operazione di UP tenda/tapparella “A” e quindi la variabile temporale va decrementata. Il codice è provvisto di test per evitare che ci siano degli overflow o underflow durante l’incremento e il decremento della variabile temporale. Lo stesso discorso si può applicare alla variabile temporale per la gestione della movimentazione della tenda/tapparella “B”. Ricapitolando, usando un trucco software, si possono ottenere delle costanti di tempo superiori al millisecondo di base del `TIMER0` in maniera semplice ed efficace. L’errore commesso nella temporizzazione della tenda/tapparella sarà al massimo di 100 msec. Infatti se ➤

ELENCO COMPONENTI:

R1: 4,7 kohm
 R2: 100 kohm
 R3: 4,7 kohm
 R4: 22 kohm
 R5, R7: 100 kohm
 R6, R8: 4,7 kohm
 R9, R10: 100 kohm
 R11+R15: 330 ohm
 R16: 10 ohm 2W
 R17: 1 kohm
 R18: 100 kohm
 RN1: rete resistiva 6x4K7 + comune
 RN2: rete resistiva 4x4K + comune
 C1: 220 pF ceramico
 C2: 220 pF ceramico
 C3: 10 μ F 25 VL elettrolitico
 C4: 220 pF ceramico
 C5: 100 nF multistrato
 C6+C8: 220 pF ceramico
 C9: 470 μ F 50 VL elettrolitico
 C10: 100 nF multistrato
 C11: 100 μ F 50 VL elettrolitico
 C12: 470 μ F 50 VL elettrolitico
 C13: 220 pF ceramico
 U1: PIC16F873 (MF662)
 U2: 7805
 D1+D5: 1N4002
 D6: 1.5KE33CA
 Q1+Q5: BC547
 LD1+LD5: led 5 mm rosso
 Y1: Risonatore ceramico 8 MHz
 BZ1: Buzzer 5V con elettronica
 PD1: Ponte diodi W08M
 F1: Termofusibile 100mA (60V_RXE010)
 TRASF1: Trasformatore 8VA 2x9V
 RL1+RL4: Relè Finder 6A

Varie:

- Morsettieria componibile 2 poli 90° (3 pz.)
- Morsettieria componibile 3 poli 90° (3 pz.)
- Zoccolo 14+14
- Strip femmina 5 poli
- Strip maschio 2 poli (4 pz.)
- Jumper (4 pz.)
- Vite 10 mm 3 MA
- Dado 3 MA
- Circuito stampato codice S662

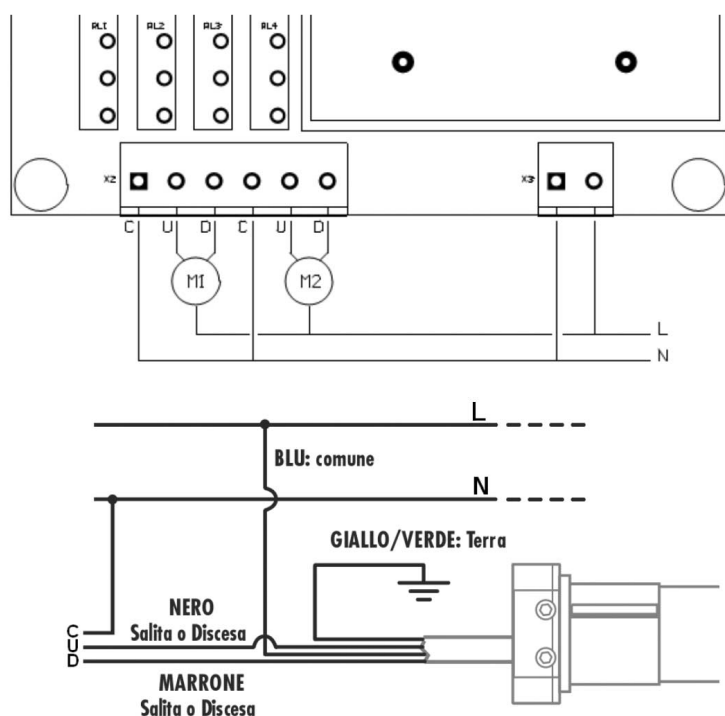


si imposta il comando di UP o DOWN della tenda/tapparella quando la variabile PRE_Delay_100 mSec è appena stata caricata si commette un errore di +100 msec e pertanto il motore sarà spento 100 mSec più tardi; l'errore non disturba in alcun modo il funzionamento della scheda.

Un'ultima osservazione: per convenzione abbiamo deciso che la condizione tenda/tapparella "tutta su" equivale alla condizione di tempo $t=0$, mentre la tenda/tapparella "tutta giù" equivale alla condizione $t=T_{max}$. Questo spiega perché, quando la tenda/tapparella si muove verso il basso, viene incre-

mentato il valore della variabile temporale e, nel caso contrario, la variabile si decrementa. Il sistema è anche in grado di memorizzare la posizione in cui si trovava dopo una mancanza di alimentazione in modo da poter riprendere a funzionare correttamente al ripristino della tensione di rete. Se invece l'a-

Fig. 6



Attenendosi allo schema di cablaggio riportato a sinistra si possono collegare facilmente i motori alla basetta. Sul mercato esistono diversi modelli di motori tubolari per l'azionamento degli avvolgibili, consigliamo, prima di acquistare il prodotto, di controllarne le caratteristiche tecniche e di effettuare il calcolo del peso che può gestire per verificarne la compatibilità con il sistema di automazione che si intende realizzare. Raccomandiamo di effettuare i collegamenti solo dopo aver tolto tensione al circuito. Al modulo di controllo può essere abbinato il kit FT549K (ElettronicaIn n. 89) che attiva il relè di uscita in presenza di pioggia o vento in funzione del sensore usato, indipendentemente dal numero dei moduli di controllo abbinati.

limentazione viene a mancare mentre la tapparella è in movimento non è possibile salvare la posizione. Di conseguenza, al riavvio il sistema legge una falsa posizione: per ovviare al problema si deve semplicemente premere il pulsante DOWN per entrambe le tende/tapparelle e lasciare che eseguano l'intera corsa. A questo punto il sistema è nuovamente riallineato e pronto al suo normale funzionamento. Oltre alla macchina a stati, il firmware segue il flusso di uno schema a blocchi (Figura 5), tra l'altro molto semplice e intuitivo. Come si può osservare si comincia con una inizializzazione della scheda, dove si configurano le porte A, B e C del PIC (Input/Output) ed eventual-

mente si impostano le uscite a valori noti. Successivamente si inizializza la memoria RAM che consiste nell'azzeramento della memoria utente per caricare in seguito dei valori noti nelle variabili di interesse, compresa la lettura in EEPROM dei valori di massimo e minimo per la movimentazione della tenda/tapparella e la posizione della stessa. Infine si procede con l'attivazione degli interrupt. Conclusa l'inizializzazione si passa al *Main* dove, ciclicamente, vengono eseguite una serie di istruzioni, comprese le macro per richiamare lo stato futuro della "FSM". L'attività principale all'interno del *Main* è la gestione degli ingressi: a seconda dei valori letti verranno eseguite o richiamate

parti di codice differente. Ad esempio, se si sta eseguendo il setup della scheda, non è possibile processare la parte di codice che si occupa della gestione pulsanti per la movimentazione dei motori.

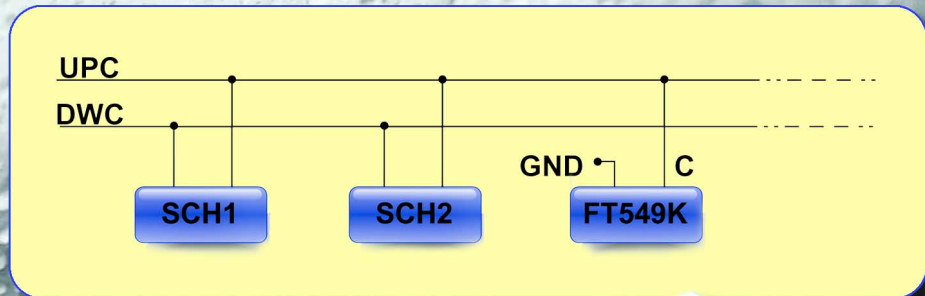
Il montaggio

Suggeriamo di iniziare il montaggio inserendo e saldando i componenti a più basso profilo per proseguire col regolatore di tensione con lo zoccolo per il micro e con i condensatori elettrolitici. Per ultimi montate i relè, i connettori ed il trasformatore di alimentazione. Naturalmente durante il montaggio prestate la massima attenzione ai componenti polarizzati; con il tronchesino tagliate tutti di terminali ➤

Per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono facilmente reperibili in commercio. Il microcontrollore già programmato (cod. MF662) costa 21,00 Euro IVA compresa. Il master del CS ed il firmware del micro possono essere scaricati gratuitamente dal sito della rivista (www.elettronica.in.it).

**Il materiale va richiesto a: Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 ~ Fax: 0331-778112 ~ <http://www.futuranet.it>**



che fuoriescono dal lato rame della basetta e, in quell'occasione controllate anche che non vi siano corto circuiti tra piazzuole o piste adiacenti. Prima di inserire il PIC nel suo zoccolo è consigliabile verificare le tensioni di alimentazione controllando che le tensioni presenti sul positivo di C2 e di C1 siano rispettivamente di +24 V e +5 V; anche sul positivo di C3 la tensione deve essere di +5 V. E' ora la volta del PIC che deve essere programmato utilizzando il firmware scaricabile gratuitamente dal nostro sito (www.elettronica.in.it). Come accennato all'inizio, se possedete il

programmatore ICD2 della Microchip (o dispositivo equivalente) potrete caricare il programma utilizzando la tecnica in-circuit; in caso contrario dovreste programmare il micro (utilizzando un programmatore adatto) impiegando la procedura tradizionale, ovvero inserendo il chip nell'apposito zoccolo del programmatore.

Se piove o tira vento

In precedenza abbiamo accennato alla possibilità di abbinare questo progetto ad un sensore pioggia o vento per ottenere una gestione automatica di tende e tapparelle.

Potremo utilizzare, ad esempio, il circuito con uscita a relè presentato sul fascicolo 89. Questo sensore può essere utilizzato come attuatore collegando l'uscita del relè alla linea comune UPC per chiudere una tenda in presenza di vento forte oppure alla DWC per abbassare tutte le tapparelle di casa quando arriva il classico temporale estivo. Il relè del modulo sensore potrà rimanere eccitato anche per un tempo molto lungo, in quanto l'intervallo necessario per aprire o chiudere gli elementi posti sotto controllo è determinato dal firmware del modulo tende/tapparelle.

RM ELETTRONICA SNC

vendita componenti elettronici
rivenditore autorizzato:



**FUTURA
ELETTRONICA**

NUOVA

ELETTRONICA

G.P.E.



Via Val Sillaro, 38 - 00141 ROMA - tel. 06/8104753



Corso di programmazione: **CAN BUS**

a cura di Carlo Tauraso

Nato come protocollo di comunicazione seriale per fare colloquiare tra loro tutti i sistemi elettronici presenti a bordo delle autovetture, si sta affermando anche nell'automazione industriale e nella domotica. In questa e nella puntata successiva presentiamo un sistema in grado di registrare in real-time i messaggi scambiati tra due nodi.



A conclusione di questo corso (e prima di presentare alcuni progetti pratici) abbiamo ritenuto opportuno dedicare altre due puntate alla pubblicazione di un interessante sistema in grado di registrare in real-time i messaggi scambiati tra due nodi. La cosa interessante è che non sarà necessario effettuare alcuna modifica hardware alla demoboard utilizzata nel corso, semplicemente dovremo sostituire il firmware del PIC con il *monitorCAN.hex* scaricabile gratuitamente dal sito della rivista. Ne è venuto fuori un progetto molto interessante che rappresenta uno strumento di diagnostica completo per tutti quei sistemi che utilizzano il CAN Bus, ad iniziare da quelli di cui ci siamo occupati nel corso. Dal punto di vista software ci siamo appoggiati al CANKing, un CAN Monitor Freeware distribuito dall'azienda svedese Kvaser (www.kvaser.com). Dal lato software abbiamo

creato dei template che permettono di configurare tale programma in modo che possa funzionare con i nodi sviluppati durante il corso dando la possibilità non solo di monitorare i messaggi ma anche di entrare in Configuration mode a runtime, modificando i registri del modulo CAN in modo da vederli in funzione in diretta. Sarà così possibile, quindi, sperimentare in maniera approfondita tutti gli argomenti trattati durante le varie puntate del corso senza lasciare nulla al caso. Per quanto riguarda il firmware è stato necessario creare un codice che risultasse compatibile sia con i comandi inviati via RS-232 dal software che con la configurazione hardware della demoboard. Il sistema rappresenta un "banco di prova" indispensabile per tutti coloro che vogliono effettuare dei test approfonditi sui propri prototipi. Molto spesso, infatti, quando si ha a che fare con diversi firmware che dialogano attraverso ➤

so una struttura di rete è fondamentale capire che cosa avviene lungo il canale di comunicazione. Ne sanno qualcosa coloro che per mestiere o semplice passione hanno avuto a che fare con lo sviluppo software/firmware in ambiente Ethernet: quante volte un buon monitor ha permesso di risolvere un problema di comunicazione consentendo di identificare il terminale o il processo che lo ha determinato. Vedere i messaggi che transitano sul bus di una rete CAN può essere estremamente utile ed interessante, sia per lo sviluppo del firmware da inserire sui vari nodi che per comprendere a fondo come avviene la comunicazione. Se poi volessimo tentare delle sperimentazioni particolari quali, ad esempio, l'elaborazione parallela, il sistema diventerebbe fondamentale per capire se la suddivisione dei compiti tra i vari nodi avviene correttamente. Attraverso il template denominato "Registri" da noi predisposto si può indagare sul funzionamento dei registri del modulo CAN vedendo come l'invio di un messaggio con ID predefinito venga intercettato attraverso un filtro. In questa puntata proveremo a monitorare i messaggi inviati dal nodo RX utilizzando lo stesso firmware del secondo esperimento del corso. Vedremo chiaramente l'aumento di temperatura nei messaggi trasmessi fino al cambiamento dell'ID quando si supera la soglia stabilita. Iniziamo, dunque, con la procedura di installazione del CANKing.

CANKing: la procedura d'installazione

Il pacchetto autoestraente CANKingPIC.exe è

scaricabile gratuitamente dal sito della rivista. Facendo doppio clic su di esso si avvia una procedura di installazione molto semplice. Il file Readme proposto contiene alcune informazioni sull'aggiornamento del browser Internet Explorer dovuto all'utilizzo della COMCTL4.7 e di un help basato su HTML. Diciamo subito che potete anche saltare questo passo visto che basta aver installato l'IE 4.0 per avere già tutto ciò che serve. Dopo aver accettato la licenza di utilizzo viene visualizzata la directory di installazione. Consigliamo di mantenere quella proposta dal software; nel caso sia necessario modificarla, ricordatevi di farlo anche quando effettuate l'installazione dei Template. Dopo aver fatto nuovamente clic sul pulsante "Avanti", viene visualizzato un form che permette di scegliere quali parti dell'archivio installare. Eliminate la spunta sia sull'opzione COMCTL che su quella relativa all'HTML Help file system in maniera da effettuare l'installazione del programma senza ulteriori aggiornamenti del sistema (Figura 1). A questo punto è possibile scegliere il gruppo di programmi nel quale verrà inserito il link per avviare il software. Anche in questo caso consigliamo di lasciare l'opzione proposta. Viene avviata la copia dei file al termine della quale fate clic sul pulsante "Finish". Viene quindi creata una voce di menu chiamata CANKing sotto Start->Programmi->Microchip. La struttura software è adesso pronta per essere configurata. Ci sono diverse parti da modificare per far funzionare correttamente il sistema con i nodi utilizzati durante le varie fasi

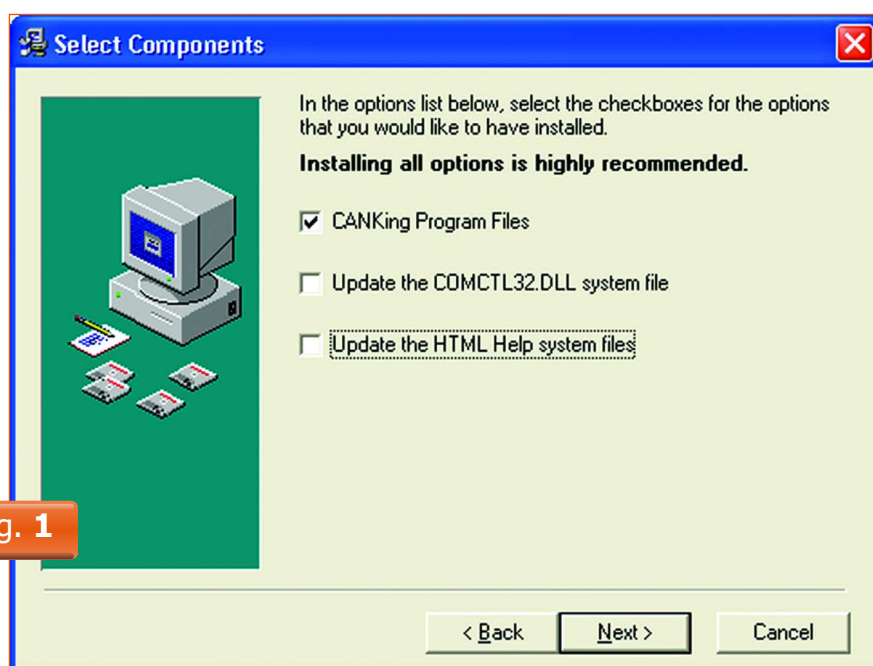


Fig. 1

del corso. Ci limiteremo a descrivere quelle essenziali per prendere confidenza con l'ambiente operativo in modo da poterlo personalizzare facilmente. Per rendere le cose più semplici anche a quanti vogliono essere immediatamente operativi, abbiamo predisposto un ulteriore file autoestraente che effettua automaticamente tutte le configurazioni creando due modelli di funzionamento che permettono sia di usare il software come Monitor che di analizzare il funzionamento dei registri CAN del PIC18F452.

Template Corso CAN: la procedura d'installazione

Dopo aver effettuato l'installazione del CANKing, passiamo alla sua integrazione con i Template predisposti. Anche in questo caso l'operazione è molto semplice. Dopo aver fatto doppio clic sul file CANTemp.exe, viene proposto un primo menu nel quale è possibile selezionare il linguaggio utilizzato nella procedura. Avviata l'installazione viene visualizzato un avviso relativo al fatto che il programma rappresenta un'integrazione di un software pre-esistente che deve essere già presente sul PC. Viene, quindi, proposta la directory d'installazione (Figura 2). Lasciate il percorso proposto o modificatelo attraverso il pulsante "Seleziona" posizionandovi sulla cartella Templates della directory utilizzata per l'installazione del CANKing. Facendo clic su



Fig. 2

"Continua" viene avviata la copia dei file. Al termine viene presentata una finestra di avviso: fate clic su OK per uscire (Figura 3). Prendiamo ora il

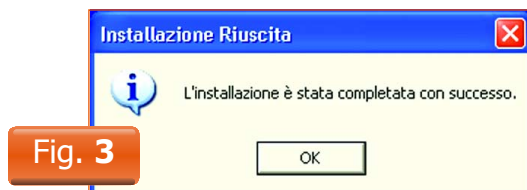


Fig. 3

nodo RX utilizzato durante il corso ed aggiorniamo il firmware contenuto nella sua memoria flash attraverso il CANmon.hex, anch'esso scaricabile gratuitamente dal sito della rivista. Il file binario della memoria EEPROM non è da modificare.

CANKing: uso

Avviate il software. La prima volta viene visualizzata una finestra attraverso la quale è possibile accedere all'help online ed alle informazioni di installazione. Selezionate la voce



Fig. 4

"Start using CANKing" e "Don't ask me again" per evitare che il form venga riproposto ogni volta che avviate il programma. Fate clic su OK. Viene visualizzato un messaggio per avvisare l'utente che l'utilizzo del software di monitoraggio su un sistema CAN reale può portare a dei malfunzionamenti che in alcuni casi possono avere conseguenze piuttosto gravi (Figura 5). Se la cosa può sembrare eccessiva, non bisogna dimenticare che molto spesso il CAN Bus viene utilizzato per sistemi piuttosto critici, sia in ambito industriale che automobilistico. Pur essendo la sicurezza degli utenti molto importante, è evidente che nel nostro caso potete tranquillamente selezionare la voce "Don't show this warning in the future" e fare clic su "OK, I know what I'm doing". Noi, infatti, utilizzeremo il sistema solo in un ambiente sperimentale. Il warning viene definitivamente disattivato premendo il tasto F7. Viene quindi visualizzata una nuova ➤

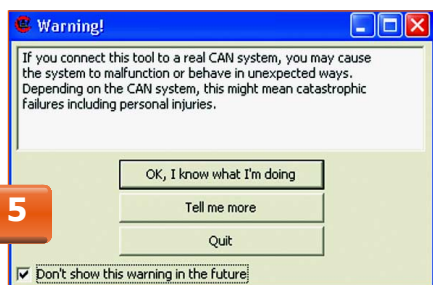


Fig. 5

finestra che permette di aprire un Template o un progetto salvato: scegliete la seconda opzione come indicato in Figura 6. Verrà visualizzato un elenco di modelli disponibili il primo dei quali, denominato TX/RX, permette di monitorare i messaggi trasferiti sul bus. Il secondo, invece, permette di effettuare un debug dei registri del PIC18F458 dedicati al modulo CAN, permettendo di comprenderne a fondo il funzionamento (Figura 7). Prima di aprire il primo

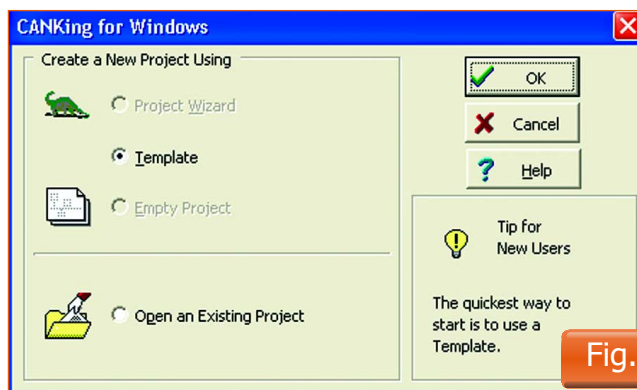


Fig. 6

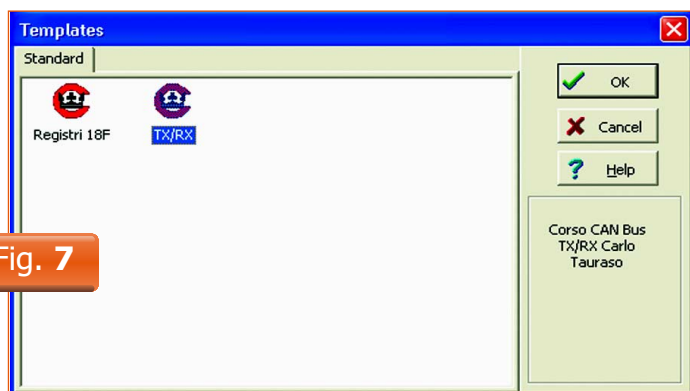


Fig. 7

modello predisponete i due nodi utilizzati durante il corso. Inserite nel nodo TX il firmware distribuito col capitolo 6 (quello relativo al secondo

ricevere i comandi da PC. Selezionando il modello TX/RX si apriranno una serie di finestre ognuna delle quali ha una funzione particolare. Selezioniamo innanzitutto la porta seriale attraverso la quale il nodo comunica con il PC; la selezione avviene attraverso la voce di menu Options->PIC18+CAN... come si vede in Figura 8.

La finestra relativa presenta alcu-

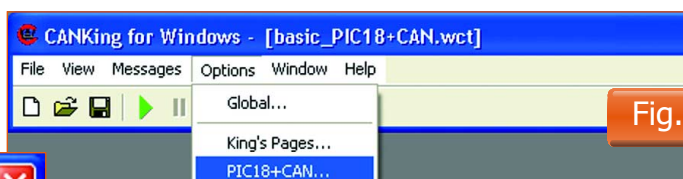


Fig. 8

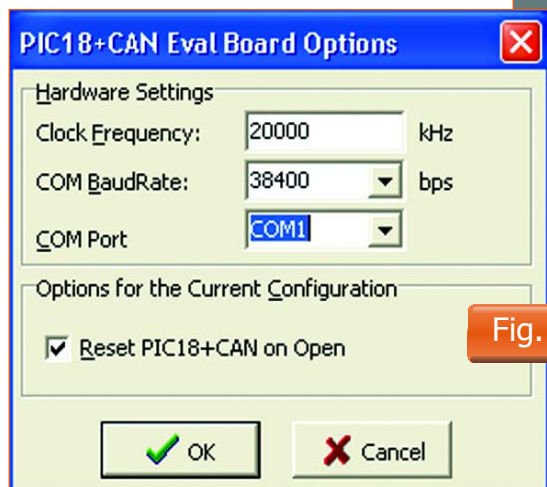


Fig. 9

ni comandi attraverso i quali è possibile modificare la frequenza dell'oscillatore che controlla il clock del PIC, variare la velocità di comunicazione della seriale, cambiare la porta in uso ed effettuare il reset della card all'avvio. Selezionate una porta libera impostando la frequenza a **20MHz** e la velocità a **38.400bps** (Figura 9): dopo aver fatto clic su "OK" ed aver applicato la modifica vedrete un cambiamento nella finestra "Evaluation Board" che aggiorna lo stato del nodo. Quest'ultimo risulta correttamente collegato al PC (Connected) ed attende

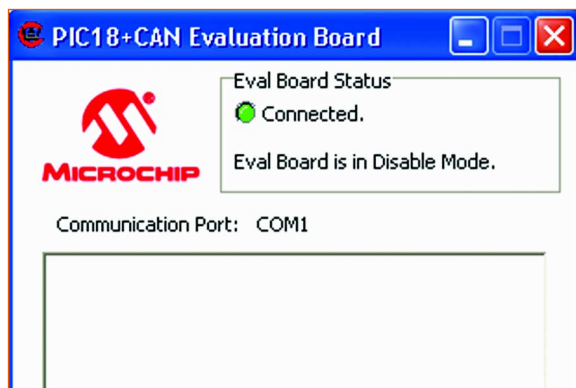


Fig. 10

il collegamento con il bus (Disable Mode) come mostrato in Figura 10. Una volta selezionata la porta seriale ed applicata la modifica sarà possibile avviare il software anche con il nodo Monitor non alimentato. La finestra "Evaluation Board" presenterà uno stato "Not Present". Alimentate il nodo Monitor: vedrete comparire una stringa di avviso nel pannello sottostante ("Avvio nodo MONITOR Corso CAN...") come si vede in Figura 11. Fate clic sul pulsante "Reset Board" e lo stato passerà a Connected/Normal Mode.

Nodo TX: sniffing dei messaggi

Siamo ora pronti a vedere come si effettua realmente un monitoraggio dei messaggi. Il modello selezionato si occupa di predisporre tutte le configurazioni necessarie sia per quanto riguarda i parametri di comunicazione del bus che per la formattazione dei risultati. Alimentiamo il nodo TX.

Inizialmente si accende il led verde e il nodo attende la pressione dello switch che controlla RB0.

Posizioniamoci sulla finestra "CAN

Controller" e facciamo clic sul pulsante "Go On Bus". Il led verde posto sul pannello si accende a significare che il sistema è pronto. Premete il pulsante collegato a RB0 del nodo TX: vedrete lampeggiare il led rosso ed immediatamente si animeranno le finestre "CAN Controller" e "Output Window". I messaggi ricevuti vengono registrati sequenzialmente nella finestra "Output Window" mentre nell'altra vedrete comparire le statistiche relative come riportato in Figura 12. Le informazioni vengono

vengono registrate sequenzialmente riportando i valori in esadecimale.

Provate a riscaldare (o raffreddare a seconda della temperatura ambiente) la sonda termica del nodo TX fino a portarla al limite predefinito. Nell'esperimento

avevamo utilizzato un valore in EEPROM pari a 0190h che corrisponde a circa 23°C. Vedrete ad un certo punto la variazione da parte del nodo TX dell'ID dei messaggi inviati col valore che passa da 121h a 123h segnalando così lo stato di allarme. La sequenza relativa è visibile in Figura 13. Il formato di registrazione è molto semplice: la prima colonna riporta l'ID del messaggio, la seconda la lunghezza dei dati trasmessi, le successive i valori dei byte. Le ultime due colonne riportano un riferimento temporale e la direzione del messaggio (Ricezione o Trasmissione).

Potete provare a monitorare i messaggi per tutti gli esperimenti che trovate nel corso, verificando il reale funzionamento dei nodi utilizzati.

CANKing: personalizzazioni

Il software in questione si presta a diverse personalizzazioni che possono ➤

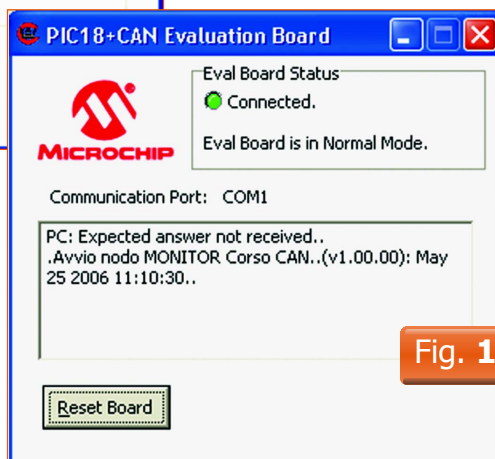


Fig. 11

avevamo utilizzato un valore in EEPROM pari a 0190h che corrisponde a circa 23°C. Vedrete ad un certo punto la variazione da parte del nodo TX dell'ID dei messaggi inviati col valore che passa da 121h a 123h segnalando così lo stato di allarme. La sequenza relativa è visibile in Figura 13. Il formato di registrazione è molto semplice: la prima colonna riporta l'ID del messaggio, la seconda la lunghezza dei dati trasmessi, le successive i valori dei byte. Le ultime due colonne riportano un riferimento temporale e la direzione del messaggio

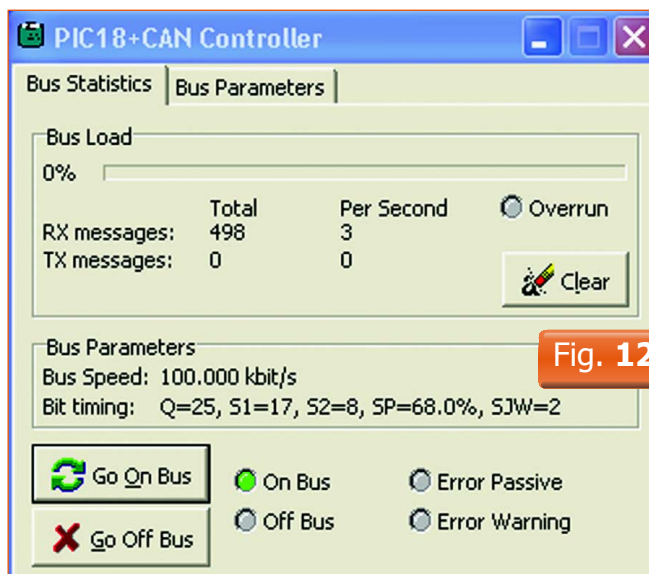


Fig. 12

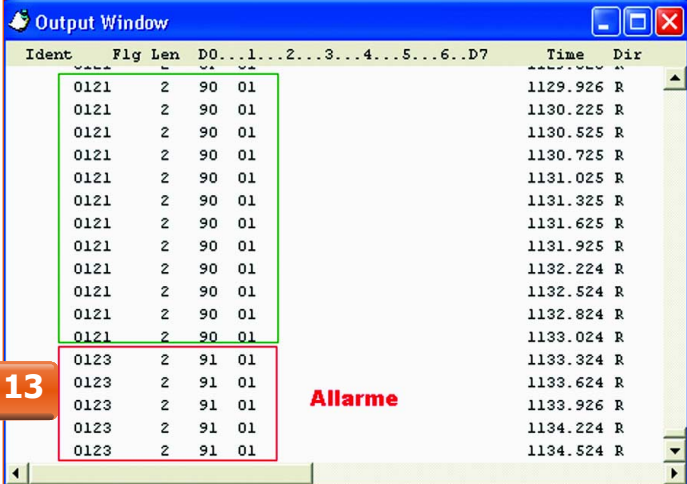


Fig. 13

Ident	Flg	Len	D0...	1...	2...	3...	4...	5...	6...	D7	Time	Dir
0121	2	90	01								1129.926	R
0121	2	90	01								1130.225	R
0121	2	90	01								1130.525	R
0121	2	90	01								1130.725	R
0121	2	90	01								1131.025	R
0121	2	90	01								1131.325	R
0121	2	90	01								1131.625	R
0121	2	90	01								1131.925	R
0121	2	90	01								1132.224	R
0121	2	90	01								1132.524	R
0121	2	90	01								1132.824	R
0121	2	90	01								1133.024	R
0123	2	91	01								1133.324	R
0123	2	91	01								1133.624	R
0123	2	91	01								1133.926	R
0123	2	91	01								1134.224	R
0123	2	91	01								1134.524	R

Allarme

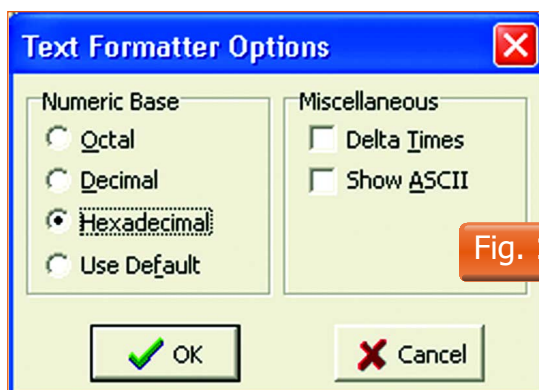
poi essere salvate in appositi file progetto. Ad esempio è possibile modificare la formattazione dei risultati ottenuti attraverso la finestra “Select Formatters”. Basta selezionare uno dei formati disponibili e fare clic sul pulsante “Use”. Selezionando la voce nel pannello “Active Formatters” è possibile modificare anche alcune opzioni di formattazione (se disponibili) come, ad esempio, il salvataggio in esadecimale o decimale dei valori rilevati (vedi Figura 14). Analogamente risulta fattibile la modifica dei parametri di funzionamento del bus attraverso la pad “Bus Parameters” presente sul form “CAN Controller” (Figura 15). Naturalmente bisogna prestare molta attenzione a quello che si sta facendo perché se non si modificano anche i parametri di funzionamento (a livello firmware) dei diversi nodi utilizzati, il monitoraggio non può avvenire correttamente. A quanti vogliono intraprendere personalizzazioni particolari segnaliamo che è possibile effettuare delle modifiche direttamente nella struttura XML dei file .wct che contengono tutti i dati utilizzati nei Template. Lo si può fare facilmente utilizzando un piccolo applicativo chiamato treedit.exe che viene installato nella stessa directory di CANKing.

CANKing: iniettare messaggi

Oltre a permettere il monitoraggio, il sistema è anche in grado di immettere sul bus un messaggio o sequenze di messaggi. Ad esempio l’invio singolo può avvenire attraverso la combinazione CTRL+U che prevede una struttura universale. Provate a cambiare il firmware del nodo TX con

quello del nodo RX presentato nel capitolo 6. Utilizzando CTRL+U o la voce Messages-> Universal-> Universal apparirà una finestra all’interno della quale potrete inserire il campo ID, la lunghezza dei dati (DLC) ed i valori da inviare. In Figura 16 si vede un tipico messaggio relativo ad una acquisizione con temperatura nella norma (ricordatevi che in questo caso i campi sono decimali quindi = 121h = 289d). Ogni volta che fate clic sul pulsante “Send” viene immesso sul bus il messaggio relativo come si vede

chiaramente nella “Output Window” di Figura 17. Ma la cosa più interessante è senz’altro la possibilità di raccogliere una sequenza di messaggi in uno script da



inviare tutto assieme, scegliendo anche la temporizzazione con cui tali messaggi vengono trasmessi. Questo fatto permette di simulare un ambiente CAN molto più realistico. Ogni volta che inviamo un messaggio esso viene salvato nella cosiddetta History List. Per visualizzarla è sufficiente usare la voce di menu View->History List; in Figura 18 vediamo come si presenta tale lista dopo aver inviato per dieci volte consecutive il messaggio singolo visto nel paragrafo precedente. La sequenza può essere salvata in uno

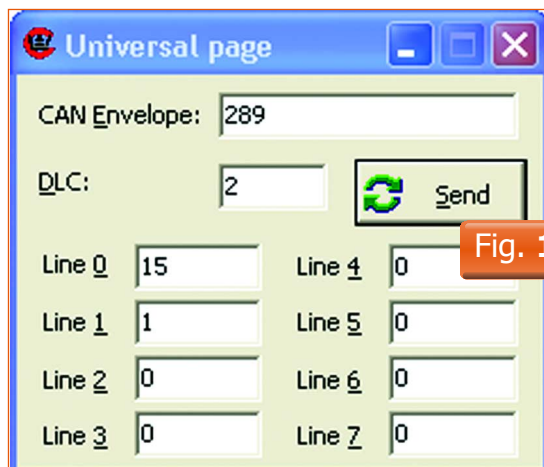


Fig. 16

Ident	Flg	Len	D0...D7	Time	Dir
0121	2	0F	01	47.924	T
0121	2	0F	01	54.976	T
0121	2	0F	01	56.129	T
0121	2	0F	01	57.230	T
0121	2	0F	01	58.178	T
0121	2	0F	01	59.171	T
0121	2	0F	01	60.119	T
0121	2	0F	01	61.087	T
0121	2	0F	01	62.027	T
0121	2	0F	01	68.808	T
0121	2	0F	01	69.846	T

Fig. 17

script con estensione .lst che potrà essere riutilizzato in altre occasioni; potremo inviare l'intera sequenza o singoli messaggi da selezionare facendo clic sulla riga relativa. È possibile scegliere la temporizzazione da utilizzare visualizzando la finestra

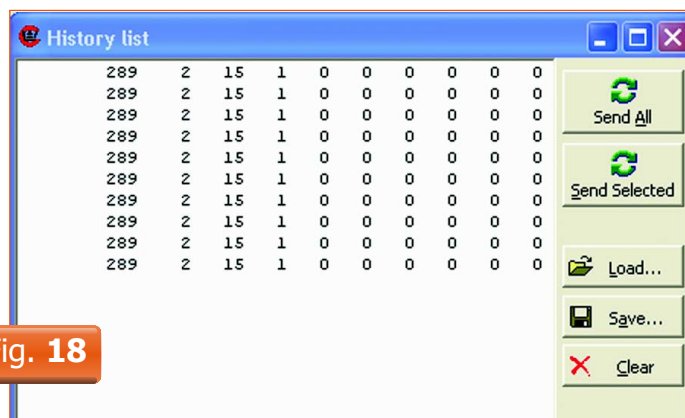


Fig. 18

“Timed Transmission” attraverso la voce di menù View -> Timed Transmission.

Come si vede in Figura 19, attraverso il form relativo è possibile definire l'intervallo di tempo tra un messaggio e l'altro definendo se la trasmissione deve avvenire una sola volta (One shot) oppure ripetuta ciclicamente. Una volta stabilito il modo con cui vogliamo effettuare la trasmissione, possiamo fare clic sul pulsante “Send All”. Il processo può essere controllato attraverso la “Output Window” e fermato e riavviato attraverso i pulsanti “Send”, “Pause”, “Stop” presenti sul form “Timed Transmission”. Nel momento in cui si avvia la trasmissione vedremo lampeggiare il led rosso del nodo di ricezione così come avevamo impostato nel firmware del secondo esperimento. È del tutto evidente che la demoboard assieme al CANKing rappresenta un sistema didattico e di testing veramente completo, permettendo sia la diagnostica

che la sperimentazione. Il nostro spazio per questo numero è terminato ma il CANKing con i nostri Template ha ancora diverse funzionalità interessanti. Il mese prossimo vedremo come utilizzare il secondo modello “Registri 18F” per studiare il funzionamento dei registri del modulo

CAN integrato nel 18F458 sperimentando la riconfigurazione a runtime vista nell'ultimo capitolo del corso.

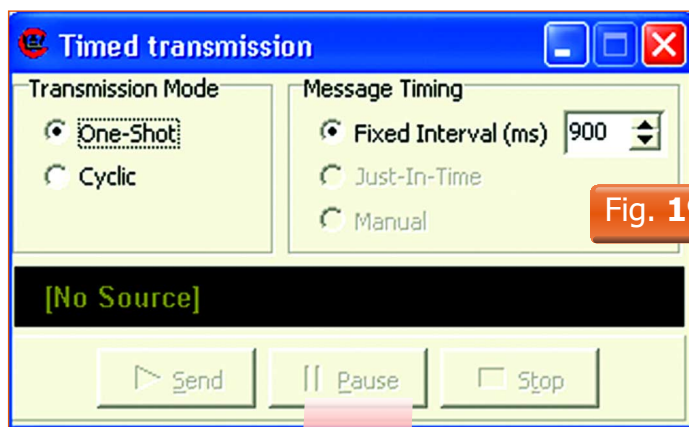


Fig. 19

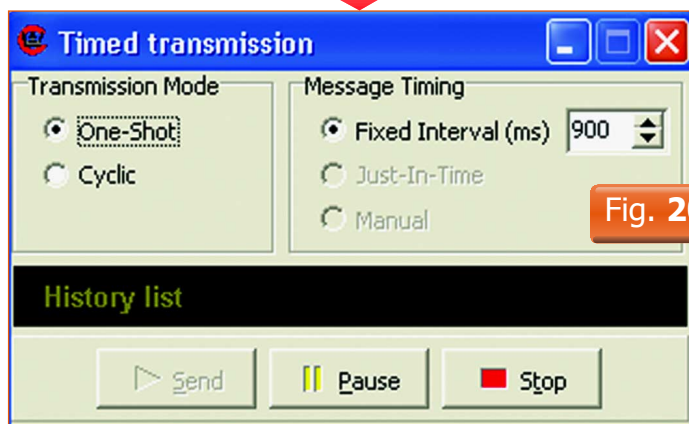


Fig. 20

Grande FIERA dell'ELETTRONICA 2006

FORLÌ 2-3 dicembre ore 9/18

col patrocinio del
Comune di Forlì



FIERA di FORLÌ • Via Punta di Ferro



Rif. ELETTRONICA IN

Photo Cine Video
2 Sale pose con modelle

dell'**Fiera Astronomia**

**CONCORSO
INVENTORE
ELETTRICO e
ELETTRONICO**

**DISCHI e CD
USATI e DA
COLLEZIONE**

**RADIO EXPO'
MERCATINO delle
RADIOCOMUNICAZIONI**

**Expo.
Elettronica**

Il grande Circuito di Fiere
dell'elettronica & Co.

organizzazione
BLU NAUTILUS
tel. 0541 439573
www.blunautilus.it

Per ottenere un **INGRESSO RIDOTTO** scarica il biglietto dal sito www.blunautilus.it
o presenta questa inserzione alla cassa



*a cura della
Redazione*

<http://www.alvarion.it>



Nella comunicazione la richiesta supera l'offerta: Wi - Max (Worldwide Interoperability for Microwave Access) la tecnologia broadband che consente la

connessione veloce ad internet senza fili a distanze chilometriche è la proposta su cui grandi aziende stanno scommettendo. Alvarion vanta già esperienze significative nel settore e nel sito sono illustrate le sperimentazioni compiute ultimamente in Italia.

<http://www.nokia.com>



Wi-Max è una connessione a larga banda adatta per la gestione di una MAN (metropolitan area network), vale a dire una copertura dai 30 ai 50 Km. Ma Wi-Max non serve solo ai PC, può anche essere utilizzata per telefonia e trasmissioni video: si tratta quindi di un sistema che non richiede cablaggi e che offre servizi anche dove, fin ora, era economicamente sconsigliato renderli disponibili, basti dire che Kabul sembra essere la città più Wi-Max della terra. Nella telefonia mobile Nokia e Motorola sono in prima linea e contano di mettere in commercio nel 2008 telefonini in grado di avvalersi delle reti Wi-Max capaci di offrire servizi evoluti.

<http://www.intel.com>



Nel settore dei personal computer, dove c'è sempre tanta voglia di banda larga, Intel si è schierata a favore di Wi-Max e sta già lavorando a chip in grado di offrire la possibilità di utilizzare il nuovo sistema a tutti gli utenti di PC portatili e fissi. In Italia, fra i grandi che si stanno impegnando, c'è Siemens: in accordo con la Pubblica Amministrazione, sta compiendo passi significativi per creare le infrastrutture che rendano la tecnologia disponibile a tutti. Secondo molti analisti, in alcuni paesi emergenti che stanno adottando questa tecnologia, Wi-Max dovrebbe raggiungere nel 2009 una quota di mercato broadband pari al 40%.

Vendo:

-Basetta sperimentale 10x12cm con contatti a molla a euro 9,00;
 -Simulatore di resistenze da 5 ohm a 1Mohm 1/3W con sonda a euro 4,00;
 -Simulatore di condensatori da 100p.F. a 4,7 µF 110V con sonda a euro 9,00;
 -Doppio wattmetro da 0,5 a 200W, impedenza da 4 a 8ohm con sonde a euro 10,00;
 -Microaspirapolvere a euro 9,00;
 -Microscopi tascabile con luce per microelettronica, micro-meccanica, ecc. a euro 12,00;
 -Accumulatore gel multiuscite contemporanee da 3 a 12V, 10A con vari spinotti e ricaricatore a euro 20,00;
 -Strumenti per la taratura dei giradischi stereo a euro 12,00;
 -Mini trasmettitore V.H.F. canale 12 audio/video a euro 5,00;
 -Pistola con pompa aria compressa con munizioni a euro 5,00;
 -11 card di suoni e ritmi per vecchie tastiere e sintetizzatori Roland a euro 14,00;
 -Vecchie giacenze di articoli introvabili elettronici, ottici, giocattoli, ecc. Spedisco.
 Contattare Pietro al numero di telefono 0371-30418.

Vendo:

-Interfaccia per modi digitali. Isola galvanicamente il computer ed il ricetrasmittitore. Contenuta in robusto contenitore professionale di alluminio anodizzato (115 x 60 x 30 mm). Completa di schema ed istruzioni, nuova.
 Il tutto a euro 35,00 + spedizione.
 -Antenna loop magnetica per i 2 metri - sintonizza da 120 a 170 MHz circa, nuova. Il tutto a euro 13,00 + spedizione.
 Contattare Luigi al numero 0125-615327.

Vendo i seguenti libri:

-“Televisori a transistor-ricerca sistematica dei guasti”;
 -“Televisori a colori-ricerca sistematica dei guasti”;
 -“Informazioni tecniche con dizionario di elettronica inglese/italiano”;
 -“Video riparatore-misure, allineamenti e ricerca guasti dei televisori” (6° edizione);
 -“La televisione a colori...? è quasi semplice”;
 -“Ripariamo i videoregistratori” (1° parte);
 -“Schermari di apparecchi radio a transistor” (5° e 9° volume) dell'Antonelliana;
 -“Servizio videotecnico-riparazione dei televisori, verifica, messa a punto” (6° edizione rivenduta);
 -“Databook, linear integrated circuits” (1° edizione SGS-ATES);
 -“Schermari di apparecchi TV B/N e TVC” (Antonelliana-Il Rostro-CELI);
 -“L'Italia delle TV locali, atlante guida per il tecnico e l'installatore JCE”;
 -“La costruzione e il calcolo dei piccoli trasformatori monofasi Delfino con assistenza.”
 Contattare Arnaldo al numero 0376-397279.

Offro:

collaborazione anche a progetto a ditte del settore. Provincia di Salerno.

Realizzo:

schede finite, prevalentemente digitali a microcontrollore. Schema, disegno del PCB, sviluppo di firmware, assemblaggio (non smd).

Eseguo:

riparazioni ed installazione elettroniche.

Vendo:

clonatori di eeprom per serie 24xxx, 93xxx, MDA2061/2 e NVM3060, 27xxx, 28xxx, 29xxx, 49xxx, PIC12C508, PIC16F84, ecc...
 Contattare Vittorio al numero 089-813042.

Vendo:

Nuovo bromografo semiprofessionale Delta Elettronica mai usato, quattro lampade, timer, area utile espositiva 40x27, corredato da basette presensibilizzate varie dimensioni.
 Il tutto a euro 80,00.
 Contattare Gennaro al numero 335-398994.

Sviluppo:

Firmware per microcontrollori Microchip PICXX e STXX, e realizzo prototipi. Sviluppo inoltre programmi in Visual Basic e C++. Contattare Gianni al numero 0376-396743.

Sviluppo

Progetti e prototipi analogici e digitali. Progetti software windows, Visual Basic.
 Contattare Giorgio al numero 392-5280800.

Vendo:

-4 Kg di componenti elettronici misti;
 -Muktimetro digitale (7 campi di misura e 32 portate);
 -Trapanino da 12V con alimentatore per forare le basette;
 -10 riviste di elettronica. Il tutto a euro 50,00.
 Contattare Francesco al numero 347-4133862.

Vendo:

-Regolatore di potenza “Fiber” alim. 220V a euro 15,00;
 -Temporizzatore multitutto con display HCA-A alim. 12/240V a euro 30,00;
 -Omron Level Meter E4M-3AK ultrasuoni con uscita analogica 4-20mA e NO/NC alim. 220V a euro 100,00;
 -Videocitofono B/N digitale a euro 100,00;
 -Termostato elettronico “Ascon” con display uscita analogica 4-20mA + NC/NO alim. 220/110V a euro 100,00;
 -Alimentatore per Commodore C128 a euro 20,00;
 -Contatore UP/DOWN con display 24VAC a euro 35,00;
 -Conduttivimetro “Castagnetti” a euro 20,00.
 Contattare il numero 348-7243384.



Tantissime idee per i tuoi regali tecnologici di Natale!

Sorprendi tutti con prodotti introvabili, unici, sicuramente originali: entra nel sito www.futuranet.it e scoprirai migliaia di prodotti per il tuo hobby e il tuo lavoro.

BABBO NATALE
in MOVIMENTO
su CORDA

€ 12⁰⁰



cod. XMSC4

TELECAMERA CMOS
a COLORI WIRELESS
con RICEVITORE

€ 69⁰⁰

cod. CP293

FALSA TELECAMERA
con LED
cod. FR223

€ 24⁰⁰

DVR a 4 CANALI
(senza HDD)

€ 362⁰⁰
cod. FR327

FREQUENZIMETRO 2,4GHz
ALTA RISOLUZIONE

€ 96⁰⁰

cod. DVM13MFC2

MIXER
a 6 CANALI
con ECO e
RIVERBERO
DIGITALI

€ 108⁰⁰

cod. PROMIX66

SFERA CAMBIACOLORE
MINIATURA

€ 2¹⁰

cod. CLB1M

CASSAFORTE DIGITALE

€ 42⁰⁰

cod. SSE017DG

cod. SSE017BL

ROBOT "TITAN TANK"
in KIT

€ 58⁰⁰ cod. KSR8

ROBOT "BOTBALL"
in KIT

€ 58⁰⁰

cod. KSR9

SET TX/RX a 2 CANALI

cod. VM109

LOCALIZZATORE
PORTATILE
GPS/GSM

€ 395⁰⁰

cod. FT596K

TORCIA a LED con DINAMO e
CARICATORE per CELLULARI

cod. ZL388

€ 12⁵⁰

STAZIONE METEO
WIRELESS 868MHz

€ 59⁰⁰

LOGGER GPS
8MB con
USCITA USB

€ 240⁰⁰

cod. GL50S

SENSORE di RETROMARCIA
con DISPLAY

cod. SPBS4

€ 72⁰⁰

cod. WS9620IT

€ 45⁰⁰

CAVO LUMINOSO COLORATO 45m

cod. RL145B

€ 56⁰⁰

cod. RL145G

cod. RL145C

cod. RL145R

cod. RL145Y

PUNTATORE LASER
VERDE TASCABILE

€ 58⁰⁰

cod. MP1900G

PROGRAMMATORE
per PIC in KIT

€ 35⁰⁰

cod. K8076

FUTURA
ELETRONICA

VIA ADIGE, 11 - 21013 GALLARATE (VA)
TEL. 0331/799775 - FAX. 0331/778112

Tutti i prezzi sono da intendersi IVA inclusa.

Puoi acquistare questi e tutti gli altri prodotti distribuiti presso i nostri Rivenditori, il nostro punto vendita di Gallarate (VA) oppure on-line collegandoti al sito www.futurashop.it. Consegne in tutta Italia in 24-48 ore!

Oscilloscopi e generatori di funzioni

velleman 

Tutta l'attrezzatura che vuoi per il tuo laboratorio elettronico

Tutti i prezzi s'intendono IVA inclusa.

**Oscilloscopio digitale
2 canali 30 MHz**



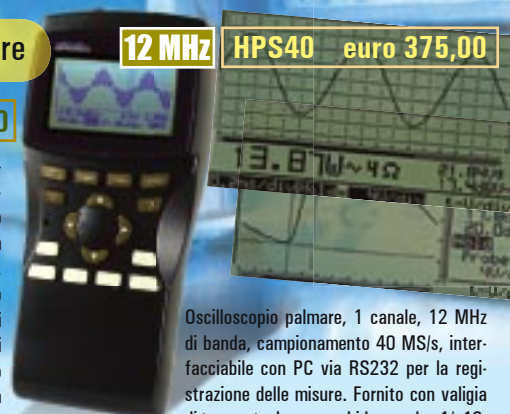
APS230 euro 690,00

Compatto oscilloscopio digitale da laboratorio a due canali con banda passante di 30 MHz e frequenza di campionamento di 240 Ms/s per canale. Schermo LCD ad elevato contrasto con retroilluminazione, autotest della base dei tempi e della scala verticale, risoluzione verticale 8 bit, sensibilità 30 μ V, peso (830 grammi) e dimensioni (230 x 150 x 50 mm) ridotte, possibilità di collegamento al PC mediante porta seriale RS232, firmware aggiornabile via Internet. La confezione comprende l'oscilloscopio, il cavo RS232, 2 sonde da 60 MHz x1/x10, il pacco batterie e l'alimentatore da rete.

Oscilloscopio palmare

2 MHz HPS10 euro 185,00

Finalmente chiunque può possedere un oscilloscopio! Il PersonalScope HPS10 non è un multimetro grafico ma un completo oscilloscopio portatile con il prezzo e le dimensioni di un buon multimetro. Elevata sensibilità - fino a 5 mV/div. - ed estese funzioni lo rendono ideale per uso hobbistico, assistenza tecnica, sviluppo prodotti e più in generale in tutte quelle situazioni in cui è necessario disporre di uno strumento leggero e facilmente trasportabile. Completo di sonda 1x/10x, alimentazione a batteria (possibilità di impiego di batteria ricaricabile).



12 MHz HPS40 euro 375,00

Oscilloscopio palmare, 1 canale, 12 MHz di banda, campionamento 40 MS/s, interfacciabile con PC via RS232 per la registrazione delle misure. Fornito con valigetta di trasporto, borsa morbida, sonda x1/x10. La funzione di autotest ne facilita l'impiego rendendo questo strumento adatto sia ai principianti che ai professionisti.

Oscilloscopio LCD da pannello

VPS10 euro 190,00

Oscilloscopio LCD da pannello con schermo retroilluminato ad elevato contrasto. Banda passante massima 2 MHz, velocità di campionamento 10 MS/s. Può essere utilizzato anche per la visualizzazione diretta di un segnale audio nonché come multimetro con indicazione della misura in rms, dB(rel), dBV e dBm. Sei differenti modalità di visualizzazione, memoria, autorange. Alimentazione: 9VDC o 6VAC / 300mA, dimensioni: 165 x 90mm (6.5" x 3.5"), profondità 35mm (1.4").



**HPS10SE
euro 210,00**

HPS10 Special Edition

Stesse caratteristiche del modello HPS10 ma con display blu con retroilluminazione. L'oscilloscopio viene fornito con valigetta di plastica rigida. La fornitura comprende anche la sonda di misura isolata x1/x10.

ACCESSORI PER OSCILLOSCOPI:

- **PROBE60S** - Sonda X1/X10 isolata/60MHz - Euro 19,00
- **PROBE100** - Sonda X1/X10 isolata/100MHz - Euro 34,00

- **BAGHPS** - Custodia per oscilloscopi HPS10/HPS40 - Euro 18,00
- **PS005** - Alimentatore non regolato 9Vdc - Euro 7,50
- **PS905AC** - Alimentatore non regolato 9Vac - Euro 6,00

Oscilloscopio digitale per PC

1 canale 12 MHz

2 canali 50 MHz

PCS500A euro 495,00

PCS100A euro 185,00

Oscilloscopio digitale che utilizza il computer e il relativo monitor per visualizzare le forme d'onda. Tutte le informazioni standard di un oscilloscopio digitale sono disponibili utilizzando il programma di controllo allegato. L'interfaccia tra l'unità oscilloscopio ed il PC avviene tramite porta parallela: tutti i segnali vengono optoisolati per evitare che il PC possa essere danneggiato da disturbi o tensioni troppo elevate. Completo di sonda a coccodrillo e alimentatore da rete.

Collegato ad un PC consente di visualizzare e memorizzare qualsiasi forma d'onda. Utilizzabile anche come analizzatore di spettro e visualizzatore di stati logici. Tutte le impostazioni e le regolazioni sono accessibili mediante un pannello di controllo virtuale. Il collegamento al PC (completamente optoisolato) è effettuato tramite la porta parallela. Completo di software di gestione, cavo di collegamento al PC, sonda a coccodrillo e alimentatore da rete.

Oscilloscopio palmare 2 MHz

Il più pratico oscilloscopio al mondo! Tutte le funzioni possono essere gestite semplicemente con il proprio pollice agendo sull'apposito joystick. Completo di interfaccia RS232 per scarico dati. Banda passante di 2 MHz con sensibilità migliore di 0,1 mV; frequenza di campionamento: 10 Ms/s. Viene fornito completo di adattatore di rete 9 V / 500 mA.



PPS10 euro 220,00

NEW

Oscilloscopio USB per PC 2 x 60 MHz



**PCSU1000
euro 520,00**

L'oscilloscopio digitale PCSU1000, dall'innovativo design studiato per ottimizzare gli spazi, utilizza per il suo funzionamento l'alimentazione prelevata dalla porta USB del PC al quale è connesso permettendo un rapido e semplice utilizzo. L'elevata risoluzione, la sensibilità d'ingresso inferiore a 0,15 mV combinati con una larga banda passante ed una frequenza di campionamento fino ad 1 GHz, fanno di questo dispositivo un valido strumento in grado di soddisfare anche i tecnici più esigenti. Particolarmente indicato per coloro che debbono effettuare misurazioni on site con il supporto di un notebook. Lo strumento viene fornito completo di software e con librerie DLL per la realizzazione di applicazioni personalizzate.

Generatore di funzioni 0,1 Hz-2MHz

Semplice e versatile generatore di funzioni in grado di fornire sette differenti forme d'onda: sinusoidale, triangolare, quadra, impulsiva (positiva), impulsiva (negativa), rampa (positiva), rampa (negativa). VCF (Voltage Controlled Frequency) interno o esterno, uscita di sincronismo TTL / CMOS, simmetria dell'onda regolabile con possibilità di inversione, livello DC regolabile con continuità. L'apparecchio dispone di un frequenzimetro digitale che può essere utilizzato per visualizzare la frequenza generata o una frequenza esterna.

DVM20 euro 245,00



Generatore di funzioni per PC PCG10A euro 180,00

Strumento abbinabile ad un PC; il software in dotazione consente di produrre forme d'onda sinusoidali, quadre e triangolari oltre ad una serie di segnali campione presenti in un'apposita libreria. Il collegamento al PC può essere effettuato tramite la porta parallela che risulta optoisolata dal PCG10A. Può essere impiegato unitamente all'oscilloscopio PCS500A nel qual caso è possibile utilizzare un solo personal computer. Completo di software di gestione, cavo di collegamento al PC, alimentatore da rete e sonda a coccodrillo.



**FUTURA
ELETTRONICA**

Via Adige, 11 - 21013 Gallarate (VA)
Tel. 0331/799775 - Fax. 0331/778112

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