

Electronica In

Mensile di elettronica applicata, attualità scientifica, novità tecnologiche.

105

Telecontrollo GSM bidirezionale con antenna integrata

Alimentatore duale professionale

Client FTP con PIC e Sd-Card

Termostato con attivazione via radio



Letttore e scrittore SD-Card su USB



Timer per spegnimento automatico

Flash upgrade per Pic Start Plus

Tutorial C18

Sensore ultrasuoni universale



ESCLUSIVO -
**CORSO
CAN-BUS**

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, autosesting 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 LCD da pannello

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").

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

Oscilloscopio palmare

HPS10
EURO 185,00

2 MHz

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).



VPS10
EURO 190,00



HPS40
EURO 375,00

12 MHz

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

HPS10 Special Edition



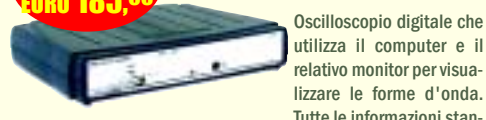
Stesse caratteristiche del modello HPS10 ma con display blu con retroilluminazione. L'oscilloscopio viene fornito con valigetta di plastica rigida. La fornitura comprende anche la sonda di misura isolata x1/x10.

HPS10SE
EURO 210,00

Oscilloscopio digitale per PC

PCS100A
EURO 185,00

1 canale 12 MHz



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.

Risposta in frequenza: 0Hz a 12MHz (± 3 dB); canali: 1; impedenza di ingresso: 1Mohm / 30pF; indicatori per tensione, tempo e frequenza; risoluzione verticale: 8 bit; funzione di autosesting; isolamento ottico tra lo strumento e il computer; registrazione e visualizzazione del segnale e della data; alimentazione: 9 - 10Vdc / 500mA (alimentatore compreso); dimensioni: 230 x 165 x 45mm; Peso: 400g. Sistema minimo richiesto: PC compatibile IBM; Windows 95, 98, ME, (Win2000 o NT possibile); scheda video SVGA (min. 800x600); mouse; porta parallela libera LPT1, LPT2 o LPT3; lettore CD Rom.

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.

Risposta in frequenza: 50 MHz ± 3 dB; ingressi: 2 canali più un ingresso di trigger esterno; campionamento max: 1 GHz; massima tensione in ingresso: 100 V; impedenza di ingresso: 1 MOhm / 30pF; alimentazione: 9 - 10 Vdc - 1 A; dimensioni: 230 x 165 45 mm; peso: 490 g.



PCS500A
EURO 495,00

2 canali 50 MHz

Generatore di funzioni per PC



PCG10A
EURO 180,00

Generatore di funzioni da abbinare 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. Possibilità di creare un'onda definendone i punti significativi. 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.

Frequenza generata: 0,01 Hz $\div$ 1 MHz; distorsione sinusoidale: <0,08%; linearità d'onda triangolare: 99%; tensione di uscita: 100m Vpp $\div$ 10 Vpp; impedenza di uscita: 50 Ohm; DDS: 32 Kbit; editor di forme d'onda con libreria; alimentazione: 9 $\div$ 10 Vdc - 1000 mA; dimensioni: 235 x 165 x 47 mm.

Generatore di funzioni 0,1 Hz - 2 MHz

DVM20
EURO 270,00



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.

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



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

Disponibili numerosi modelli di multimetri, palmari e da banco. Per caratteristiche e prezzi visita la sezione Strumenti del nostro sito www.futuranet.it

Tutti i prezzi sono da intendersi IVA inclusa.

con il patrocinio di
Comune di Faenza

Expo[®] Elettronica

mostra mercato

FAENZA 4-5 marzo 2006

 **FAENZA FIERE - Viale Risorgimento, 1** ore **9/18**

RIF. ELETTRONICA IN



**RADIO:
EXPO'**

**MERCATINO delle
RADIOCOMUNICAZIONI**

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

Solo Sabato 4 marzo

Sponsor Expo Elettronica 2006



organizzazione
BLU NAUTILUS srl
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



Pag. 48

Pag. 35

Pag. 71

15

TEMPORIZZATORE PER SPEGNIMENTO AUTOMATICO

Ideale per programmare lo spegnimento automatico di apparecchi per uso domestico funzionanti a 220 Vac. Va collegato in serie alla loro alimentazione; allo scadere dell'intervallo impostato con gli appositi comandi, sconnette automaticamente la linea.

21

ALIMENTATORE DUALE PROFESSIONALE

Gestito da microcontrollore, fornisce due tensioni continue ben stabilizzate e simmetriche comprese tra ± 1 e ± 36 volt rispetto a massa; ideale per il laboratorio, può erogare fino a 3A per ramo. I valori si impostano mediante una pulsantiera e con l'ausilio di un display LCD. In questa seconda puntata ci occupiamo dei moduli DAC e Microprocessore.

35

LETTORE E SCRITTORE SD-CARD SU USB

Consente di scrivere dati in memorie Secure Digital e di rileggerli da computer, mediante la porta USB. Vediamo come il dispositivo è stato sviluppato sui piani hardware e software, grazie all'adozione di un microcontrollore dotato di Universal Serial Bus.

48

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. Prima puntata.

S o m m a r i o

ELETTRONICA IN

www.elettronica.in
Rivista mensile, anno XII n. 105
FEBBRAIO 2006

Direttore responsabile:

Arsenio Spadoni
(Arsenio.Spadoni@elettronica.in)

Redazione:

Davide Scullino, Gabriele Daghetta, Paolo Gaspari, Boris Landoni, Alessandro Sottocornola, Francesco Doni.
(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
Telefax 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.DI.P. Angelo Patuzzi S.p.A.
via Bettola 18

20092 Cinisello B. (MI)

Telefono 02-660301 telefax 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 giorno 3-05-1995.

Una copia Euro 4,50, arretrati Euro 9,00

(effettuare versamento sul CCP n. 34208207 intestato a VISPA snc)

(C) 1995 - 2005 VISPA s.n.c.

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 fototipo sono realizzati in DeskTop Publishing

con programmi Quark XPress 6.1 e Adobe Photoshop 8.0 per

Windows. Tutti i diritti di riproduzione o di traduzione degli arti-

coli pubblicati sono riservati a termine di Legge per tutti i Paesi.

I circuiti descritti su questa rivista possono essere realizzati

solo per uso dilettantistico, ne è proibita la realizzazione a

carattere commerciale ed industriale. L'invio di articoli implica

da parte dell'autore l'accettazione, in caso di pubblicazione,

dei compensi stabiliti dall'Editore. Manoscritti, disegni, foto ed

altri materiali non verranno in nessun caso restituiti. L'utilizzo

degli schemi pubblicati non comporta alcuna responsabilità da

parte della Società editrice.

57

CLIENT FTP CON PIC E SD-CARD

Utilizziamo un microcontrollore Microchip per pubblicare via FTP i dati su un server Internet. Per la prima volta adottiamo un'interfaccia di rete che costruiamo interamente a partire da uno dei più popolari chip ethernet: l'RTL8019 di Realtek. Con uno specifico firmware, il circuito proposto può funzionare anche da Web-server. Seconda parte.

67

LA RIVISTA A PORTATA DI CLIC

Una nuova veste e tante interessanti proposte per il rinnovato sito di Elettronica In. Tra le iniziative più significative segnaliamo la disponibilità di tutti i nostri prodotti in formato digitale. E per gli abbonati appassionati di programmazione una gradita sorpresa!

71

SENSORE A ULTRASUONI UNIVERSALE

Unità di misura in grado di rilevare la presenza di persone e cose fino alla distanza di un paio di metri. Utilizzabile sia come radar di retromarcia per veicoli che per realizzare apparecchiature per l'automazione industriale, piccoli robot, ecc. Dispone di una barra a led per l'indicazione visiva della distanza e di un cicalino di allarme.

79

TUTORIAL COMPILATORE C18

Muoviamo i primi passi alla scoperta del C18 Microchip, ideale per la programmazione di sistemi complessi quali il CAN-Bus. In questa terza puntata analizziamo strutture, funzioni, parametri, puntatori ed altro ancora.

88

FLASH UPGRADE PER PICSTART PLUS

Per il più diffuso programmatore dei suoi microcontrollori, Microchip rende oggi disponibile il modulo flash per aggiornare il firmware senza dover tutte le volte programmare un nuovo micro e sostituirlo a quello esistente. Si installa facilmente nello zoccolo del PIC17C44 e...

91

TERMOSTATO CON ATTIVAZIONE VIA RADIO

Permette di azionare a distanza via radio una caldaia o un condizionatore d'aria usando un tradizionale termostato al quale è stato abbinato un sistema radio codificato. Il trasmettitore è collegato al termostato mentre il ricevitore è posto in prossimità della caldaia o del condizionatore.

101

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 terza puntata entriamo nel vivo dello sviluppo del modulo CAN e analizziamo l'hardware che useremo nei nostri esperimenti.



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

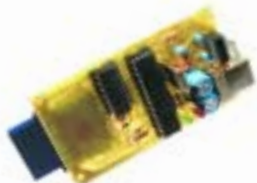
15



21



35



48



57



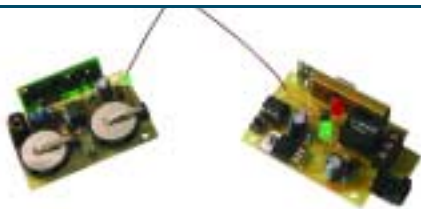
71



79



91



101



Editoriale

Influenza dei polli...o polli influenzati?

Fa riflettere e preoccupa l'avanzata del virus H5N1, vista finora - dai nostri occhi di spettatori annoiati di una TV "spazzatura" o assonnati lettori che sfogliano giornali tra un cambio di marcia e l'altro nel traffico mattutino - come una notizia sulla quale dibattere nella tranquillità di chi ha la consapevolezza che è una cosa seria ma riguarda solo gli altri. Un po' come chi, per cinica curiosità o bisogno di emozioni che risvegliano dal torpore di una vita piatta, rallenta per strada quando c'è un incidente nella direzione opposta di marcia, credendo di essere in un *reality* e pensando che ci si può godere lo spettacolo comodamente seduti in auto, tanto coinvolge gli altri. Ma puntualmente il problema degli altri bussa alla nostra porta; la notizia riesce a farsi udire malgrado il rumore di una campagna elettorale "gridata", come vuole la scuola di comunicazione del piccolo schermo, e ci mette in agitazione. Volendo fare dell'ironia diremmo: "poco male"... agitandoci si sente di meno il freddo causato dalle restrizioni all'uso del riscaldamento imposte per decreto da chi non può più nascondersi la nostra dipendenza dall'estero e l'incapacità di una classe dirigente che ancora non ha una seria politica energetica. "Il raziocinio è un lume che uno può accendere quando vuole obbligar gli altri a vedere, e può soffiarsi sopra, quando non vuol più veder lui"; così diceva il *Manzoni*, così si comporta chi legifera e impone agli altri norme tanto indispensabili che, quando non conviene, è il primo a ignorare. Stiamo cercando di recuperare anni di ritardo nella conversione a gas delle centrali termoelettriche, però poi dobbiamo fare "dietrofront" perché il metano scarseggia; ne avremmo di più svincolandoci da canali di approvvigionamento rigidi come i gasdotti: basterebbe comperarlo da altri paesi trasportandolo con le *gasiere*, navi simili alle petroliere, nelle cui stive si trova, però, metano liquido, che verrebbe scaricato in *rigassificatori* allestiti nei porti e lì riportato allo stato gassoso. Operazione che, tra l'altro, produce freddo a costo zero, utilizzabile in celle frigorifere per alimentari in arrivo via mare. Ma ancora si discute sul rischio di esplosione e l'obbligo di far navigare fuori da ampie zone di rispetto le altre navi. Mentre i colleghi della cronaca fanno il loro mestiere di divulgatori di notizie (spesso inquietanti, perché fanno più audience...) noi proviamo a consolarvi dicendovi che STM ha sviluppato un chip a basso costo per la diagnosi rapida dell'influenza aviaria, una lastrina comprendente un reattore integrato PCR (catena della reazione della polimerase) usato per separare e replicare alternativamente frammenti di DNA, moltiplicandoli fino a raggiungere una quantità sufficiente a sapere in breve tempo se un soggetto è contagiato dal temuto virus. Sì, sappiamo che vorreste già in queste pagine un'applicazione pratica del chip, o, magari, un generatore di elettricità da montare sul balcone di casa per alimentare una stufetta quando la Russia dovrà tenere tutto il metano per sè...

Nell'attesa che dal silicio prendano forma nuove idee, proseguiamo nel nostro cammino cercando, ogni mese, di darvi un po' di quella conoscenza, che non ci appartiene ma prendiamo a prestito e passiamo, come è dovere di ogni divulgatore, a chi vuole apprenderla. SD-Card, pubblicazioni in FTP, linguaggio C18 e CAN-Bus sono tra gli argomenti che tratteremo. Sono forse poca cosa, ma se non ci mette al riparo dall'*influenza dei polli*, un po' di cultura può almeno risparmiarci di diventare anche noi "polli influenzati".

Arsenio Spadoni

(Arsenio.Spadoni@elettronica.in.it)

[elencoInserzionisti]

Bias	Idea Elettronica
E.R.F.	Idriss
Expo Elettronica - Blu Nautilus	Ipersicurezza
Fiera di Empoli	Mostra reg. elettronica Scandiano
Fiera di Gonzaga	Promozioni Fieristiche
Fiera di Montichiari	RM Elettronica
Fiera di Pordenone	Telstar
Fiera di Rovigo	www.mdsrl.it
Futura Elettronica	

La tiratura di questo numero è stata di 22.000 copie.

Microtelecamere e telecamere su scheda

La videosorveglianza a portata di mano

**FUTURA
ELETTRONICA**

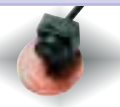
Via Adige, 11 - 21013 GALLARATE (VA)
Tel. 0331/799775 - Fax. 0331/778112 - www.futuranet.it

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

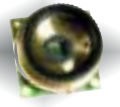
Modelli
CMOS
da circuito
stampato

FR302 - Euro 56,00	FR301 - Euro 27,00	FR300 - Euro 23,00
		
Tipo: sistema standard PAL (colori)	sistema standard CCIR (B/N)	sistema standard CCIR (B/N)
Elemento sensibile: 1/3" CMOS	1/3" CMOS	1/3" CMOS
Risoluzione: 380 Linee TV	240 Linee TV	240 Linee TV
Sensibilità: 3 Lux (F1.4)	2 Lux (F1.4)	2 Lux (F1.4)
Ottica: f=6 mm, F1.6	f=4,9 mm, F2.8	f=7,4 mm, F2.8
Alimentazione: 5Vdc - 10mA	5Vdc - 10mA	5Vdc - 10mA
Dimensioni: 20x22x26mm	16x16x15mm	21x21x15mm

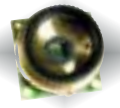
Modelli
CMOS

FR220 - Euro 96,00	FR220P - Euro 125,00	FR125 - Euro 44,00	FR126 - Euro 52,00	CAMZWCMM1 - Euro 26,00	CAMCOLMHA5 - Euro 44,00	CAMZWBLA3 - Euro 34,00
						
Tipo: sistema standard CCIR (B/N)	sistema standard CCIR (B/N)	sistema standard CCIR (B/N)	sistema standard PAL (colori)	sistema standard CCIR (B/N)	sistema standard PAL (colori)	sistema standard CCIR (B/N)
Elemento sensibile: 1/4" CMOS	1/4" CMOS	1/3" CMOS	1/3" CMOS	1/4" CMOS	1/3" CMOS	1/4" CMOS
Risoluzione: 240 linee TV	240 linee TV	380 Linee TV	380 Linee TV	380 Linee TV	380 Linee TV	240 Linee TV
Sensibilità: 0,5 Lux (F1.4)	0,5 Lux (F1.4)	0,5 Lux (F1.2)	3 Lux (F1.2)	0,5 Lux (F1.4)	1,5 Lux (F2.0)	0,1 Lux (1.2)
Ottica: f=3,5 mm, F2.6 PIN-HOLE	f=3,1 mm, F3.4 PIN-HOLE	f=5 mm, F4.5 PIN-HOLE	f=5 mm, F4.5 PIN-HOLE	f=2,2 mm	f=2,8 mm	f=3,6mm F2.0
Alimentazione: 7 -12Vdc - 50mA	7 -12Vdc - 20mA	12Vdc - 50mA	12Vdc - 50mA	8Vdc - 100mA	8Vdc - 100mA	9-12Vdc - 500mA
Dimensioni: 8,5x8,5x15 mm	8,5x8,5x10mm	27,5x17x18mm	20,5x28x17mm	18x18x17mm	26x21x18mm	54x38x28mm
		Stesso modello con ottica f=3,6mm FR125/3.6 - Euro 48,00	Stesso modello con ottica f=3,6mm FR126/3.6 - Euro 56,00	Confezione completa di alimentatore da rete.	Confezione completa di alimentatore da rete.	

Modelli
CCD
in B/N

FR72 - Euro 48,00	FR72/PH - Euro 46,00	FR72/C - Euro 46,00	FR72/LED - Euro 50,00
			
Tipo: sistema standard CCIR	sistema standard CCIR	sistema standard CCIR	sistema standard CCIR
Elemento sensibile: 1/3" CCD	1/3" CCD	1/3" CCD	1/3" CCD
Risoluzione: 400 Linee TV	400 Linee TV	400 Linee TV	400 Linee TV
Sensibilità: 0,3 Lux (F2.0)	0,5 Lux (F2.0)	in funzione dell'obiettivo	0,01 Lux
Ottica: f=3,6 mm, F2.0	f=3,7 mm, F3.5	-	f=3,6 mm, F2.0
Alimentazione: 12Vdc - 110mA	12Vdc - 110mA	12Vdc - 110mA	12Vdc - 150mA
Dimensioni: 32x32x27mm	32x32x20mm	32x32mm	55x38mm
Stesso modello con ottica: • f=2,5 mm FR72/2.5 € 48,00 • f=2,9 mm FR72/2.9 € 48,00 • f=6 mm FR72/6 € 48,00 • f=8 mm FR72/8 € 48,00 • f=12 mm FR72/12 € 48,00 • f=16 mm FR72/16 € 48,00		Il modulo dispone di attacco standard per obiettivi di tipo C/CS.	

Modelli
CCD
a COLORI

FR89 - Euro 95,00	FR89/PH - Euro 95,00	FR89/C - Euro 95,00	FR168 - Euro 110,00
			
Tipo: sistema standard PAL	sistema standard PAL	sistema standard PAL	sistema standard PAL
Elemento sensibile: 1/4" CCD	1/4" CCD	1/4" CCD	1/4" CCD
Risoluzione: 380 Linee TV	380 Linee TV	380 Linee TV	380 Linee TV
Sensibilità: 0,2 Lux (F1.2)	1 Lux (F1.2)	0,5 Lux (F1.2)	2 Lux (F2.0)
Ottica: f=3,7 mm, F2.0	f=5,5 mm, F3.5	-	f=3,7 mm, F2.0
Alimentazione: 12Vdc - 80mA	12Vdc - 80mA	12Vdc - 80mA	12Vdc - 65mA
Dimensioni: 32x32x32mm	32x32x16mm	32x34x25mm	26x22x30mm
Stesso modello con ottica: • f=2,9mm FR89/2.9 € 95,00		Il modulo dispone di attacco standard per obiettivi di tipo C/CS.	Stesso modello con ottica: • f=5,5mm FR168/PH € 110,00

Tutti i prezzi sono da intendersi IVA inclusa.

Firewire, USB e periferiche PC

Ormai i notebook che si trovano in commercio hanno, come porta cui connettere le periferiche esterne, solamente l'USB; qualcuno, però, oltre agli slot PCMCIA sembra disporre di un'altra connessione, che pare si chiami FireWire. Sapete dirmi di cosa si tratta e in cosa differisce dalla USB?

Matteo Riboldi - Asti

La consuetudine di non montare più le interfacce parallela (Centronics) e seriale (RS232) nei PC portatili deriva dal fatto che l'USB è un bus universale sul quale possono essere collegate praticamente tutte le tipologie di periferica, persino le interfacce wireless e Bluetooth (seppure tramite l'apertura di porte COM virtuali).

Quanto alla FireWire, è un'interfaccia più limitata, ma non per le prestazioni che può garantire, bensì per il fatto che è nata ad opera di Apple Computer (è stata classificata come IEEE1394) allo scopo di realizzare uno standard di connessione al computer per le videocamere e le macchine fotografiche digitali (DV). A parte questa specificità

della FireWire, ciò che differenzia le due interfacce va cercato nelle loro peculiarità: la USB è sostanzialmente un bus a due fili più l'alimentazione; i due fili portano l'uno il clock e l'altro (bidirezionale) i dati. Il bus può indirizzare 127 dispositivi, con i quali dialoga (in base al loro identificativo) ad una velocità limitata a 60 Mbit/s. nella prima versione introdotta (la 1.1) e portata a 480 Mbps nella più recente implementazione (USB 2.0).

L'USB 2.0 usa un'architettura master-slave, in cui il computer fa da gestore e decide le velocità di tutte le connessioni, da e verso il PC e tra le periferiche.

I due fili di alimentazione forniscono 5 volt e 500 mA di corrente. Il FireWire è ancora un bus, ma a sei fili, lungo i quali transitano i dati e il clock; il transfer rate, originariamente di 400 Mbit/s. nell'ultima implementazione (FireWire 800 o IEEE1394b) raggiunge ben 800 Mbps. Questo spiega la ragion d'essere dell'interfaccia, nata, quando ancora c'era l'USB 1.1, per trasferire i filmati dalle videocamere DV al computer (per l'editing video) con maggior rapidità di quanto permettesse l'USB stesso. Anche questa connessione può dare

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.

alimentazione ai dispositivi collegati: precisamente 5 volt, però con maggiore generosità per quel che riguarda la corrente, che può spingersi a 9 A (45 W).

La Firewire supporta fino a 63 periferiche organizzate in una rete non ciclica e prevede una comunicazione "peer-to-peer" tra i dispositivi che, quindi, possono comunicare tra loro senza dover utilizzare il computer come arbitro.

Per esempio, una videocamera digitale potrebbe riversare il filmato video su un hard disk esterno senza l'intervento del computer.

Supporta il collegamento a caldo e la presenza di più Host tramite una gestione degli IP software. Quindi una connessione FireWire può essere utilizzata per creare una rete locale tra due computer quattro volte più veloce di una Ethernet a 100 Mbit.

La versione Sony (conosciuta come iLink) ha solo quattro pin e non fornisce alcuna alimentazione; i dispositivi vanno alimentati a parte. L'interfaccia Firewire è tecnicamente superiore all'interfaccia USB, ma questa è molto più diffusa per via dei brevetti; infatti Apple richiede il pagamento dei brevetti per ogni implementazione della FireWire (costo: 0,25 dollari per utente finale).

Sebbene siano cifre ridotte, molti produttori realizzano prodotti a bassissimo margine di guadagno e preferiscono utilizzare la tecnologia USB, che, essendo esente dal pagamento di brevetti, consente loro di ottenere prodotti più economici.



In alto a sinistra il connettore maschio Firewire 400 presente anche nelle videocamere DV, al centro gli spinotti Firewire 400 e 800 e a destra la presa IEEE1394b. In mezzo, a sinistra la presa USB B e a destra lo spinotto USB B; sotto, a sinistra lo spinotto USB di tipo A e, a destra, la presa USB A.

Con il patrocinio:
COMUNE DI EMPOLI
A.R.I. - Sezione di Vinci

EMPOLI

Palazzo delle Esposizioni
8 e 9 Aprile 2006

Mostra Empolese della radiantistica e dell'elettronica

**mostramercato attrezzature
elettroniche, radiantistiche,
ricezione via satellite,
informatica &
editoria specializzata**

per informazioni :

ENTE ORGANIZZATORE E SEGRETERIA:
STUDIO FULCRO s.a.s.

Tel. 010.56.11.11 - Fax 010.59.08.89

www.studio-fulcro.it - info@studio-fulcro.it

Campagna abbonamenti 2005 / 2006

Perché 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 semplicemente appassionati, non potete perdere neppure un fascicolo della nostra rivista! Citiamo, ad esempio, alcuni degli argomenti di cui ci siamo occupati nel corso del 2005:



Unità di memoria con SD-Card

Come utilizzare una SD-Card per realizzare una economica unità di memoria gestibile mediante protocollo seriale.

Questo progetto può essere collegato sia ad un PC che utilizzato con apparecchiature stand-alone che necessitano di una elevata capacità di memoria. Completo di programma di test per PC.

Controllo accessi long-distance con RFID

Controllo accessi a "mani libere" basato su tecnologia RFID a TAG attivi in grado di garantire una portata di alcuni metri. Possibilità di funzionamento sia in modalità stand-alone che in abbinamento ad un Personal Computer.

Localizzatore portatile GPS/GSM con cartografia Internet

Piccolissimo localizzatore remoto con modulo combinato GSM/GPS Wavecom la cui posizione può essere verificata sfruttando una connessione Internet ed utilizzando le cartine presenti in rete.

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.
- ✓ Anche 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:

- ➔ 1) La **Discount Card** che ti permette di usufruire di uno sconto del **10%** su tutti i prodotti acquistati direttamente presso la ditta **FUTURA ELETTRONICA**.
- ➔ 2) un **volume a scelta** della collana "**L'ELETTRONICA PER TUTTI**" (15,00 cad.).



Programmiamo con i PIC



100+1 circuiti elettronici



Alla scoperta della CCTV

- ➔ 3) Gli abbonati (e solo loro!) potranno scaricare gratuitamente dal sito www.elettronicaain.it i file sorgente dei programmi e dei firmware utilizzati in molti dei progetti pubblicati.

new

Abbonamento annuale

solo **e 36,00** con omaggio!!!

Come fare ad abbonarsi?

→ On-line tramite Internet

compilando il modulo riportato nella pagina "Abbonamenti" disponibile nel sito Internet "www.elettronica.in.it".
Se possedete una carta di credito potrete effettuare il pagamento contestualmente alla richiesta.
E' anche possibile attivare l'abbonamento richiedendo il pagamento tramite C/C postale.

oppure

→ 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 ne possedete una 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*;

☐ Programmiamo con i PIC; ☐ 100+1 circuiti elettronici; ☐ Alla scoperta della CCTV.

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

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. E' 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.

Network-enable

Prezzi speciali per quantità

Una serie di prodotti che consentono di collegare qualsiasi periferica dotata di linea seriale ad una LAN di tipo Ethernet. Firmware aggiornabile da Internet, software disponibile gratuitamente sia per Windows che per Linux.

EM100 Ethernet Module



Realizzato appositamente per collegare qualsiasi periferica munita di porta seriale ad una LAN tramite una connessione Ethernet. Dispone di un indirizzo IP proprio facilmente impostabile tramite la LAN o la porta seriale. Questo dispositivo consente di realizzare apparecchiature "stand-alone" per numerose applicazioni in rete. Software e firmware disponibili gratuitamente.

[EM100 - Euro 52,00]

EM120 Ethernet Module



Simile al modulo EM100 ma con dimensioni più contenute. L'hardware comprende una porta Ethernet 10BaseT, una porta seriale, alcune linee di I/O supplementari per impieghi generici ed un processore il cui firmware svolge le funzioni di "ponte" tra la porta Ethernet e la porta seriale. Il terminale Ethernet può essere connesso direttamente ad una presa RJ45 con filtri mentre dal lato "seriale" è possibile una connessione diretta con microcontrollori, microprocessori, UART, ecc.

[EM120 - Euro 54,00]

EM200 Ethernet Module



Si differenzia dagli altri moduli Tibbo per la disponibilità di una porta Ethernet compatibile 100/10BaseT e per le ridotte dimensioni (32,1 x 18,5 x 7,3 mm). Il modulo è pin-to pin compatibile con il modello EM120 ed utilizza lo stesso software messo a punto per tutti gli altri moduli di conversione Ethernet/seriale. L'hardware non comprende i filtri magnetici per la porta Ethernet. Dispone di due buffer da 4096 byte e supporta i protocolli UDP, TCP, ARP, ICMP (PING) e DHCP.

[EM200 - Euro 58,00]

EM202 Ethernet Module



Modulo di conversione Seriale/Ethernet integrato all'interno di un connettore RJ45. Particolarmente compatto, dispone di quattro led di segnalazione posti sul connettore. Uscita seriale TTL full-duplex e half-duplex con velocità di trasmissione sino a 115 Kbps. Compatibile con tutti gli altri moduli Tibbo e con i relativi software applicativi. Porta Ethernet compatibile 100/10BaseT.

[EM202 - Euro 69,00]

DS100 Serial Device Server

- ✓ Convertitore completo 10BaseT/Seriale;
- ✓ Compatibile con il modulo EM100.

[DS100 - Euro 115,00]



Server di Periferiche Seriali in grado di collegare un dispositivo munito di porta seriale RS232 standard ad una LAN Ethernet, permettendo quindi l'accesso a tutti i PC della rete locale o da Internet senza dover modificare il software esistente. Dispone di un indirizzo IP ed implementa i protocolli UDP, TCP, ARP e ICMP. Alimentazione a 12 volt con assorbimento massimo di 150 mA. Led per la segnalazione di stato e la connessione alla rete Ethernet.

[Disponibile anche nella versione con porta multistandard RS232 / RS422 / RS485, codice prodotto **DS100B** - Euro 134,00].

DS202R Tibbo



Ultimo dispositivo Serial Device Server nato in casa Tibbo, è perfettamente compatibile con il modello DS100 ed è caratterizzato da dimensioni estremamente compatte. Dispone di porta Ethernet 10/100BaseT, di buffer 12K*2 e di un più ampio range di alimentazione che va da 10 a 25VDC. Inoltre viene fornito con i driver per il corretto funzionamento in ambiente Windows e alcuni software di gestione e di programmazione.

[DS202R - Euro 134,00]

E' anche disponibile il kit completo comprendente oltre al Serial Device Server DS202R, l'adattatore da rete (12VDC/500mA) e 4 cavi che permettono di collegare il DS202R alla rete o ai dispositivi con interfaccia seriale o Ethernet [DS202R-KIT - Euro 144,00].

EM202EV Ethernet Demoboard



Scheda di valutazione per i moduli EM202 Tibbo.

Questo circuito consente un rapido apprendimento delle funzionalità del modulo di conversione Ethernet/seriale EM202 (la scheda viene fornita con un modulo). Il dispositivo può essere utilizzato come un Server Device stand-alone. L'Evaluation board implementa un pulsante di setup, una seriale RS232 con connettore DB9M, i led di stato e uno stadio switching al quale può essere applicata la tensione di alimentazione (9-24VDC).

[EM202EV - Euro 102,00]

Tabella di comparazione delle caratteristiche dei moduli Ethernet Tibbo

	EM100	EM120	EM200	EM202
Codice Prodotto				
Collegamenti	Pin			RJ45
Porta Ethernet	10BaseT		100/10BaseT	
Filtro	Interno	Esterno		Interno
Connettore Ethernet (RJ45)	Esterno			Interno
Porta seriale	TTL; full-duplex (adatto per RS232/RS422) e half-duplex (adatto per RS485); linee disponibili (full-duplex mode): RX, TX, RTS, CTS, DTR, DSR; Baudrates: 150-115200bps; parity: none, even, odd, mark, space; 7 or 8 bits.			
Porte supplementari I/O per impieghi generali	2	5		0
Dimensioni Routing buffer	510 x 2 bytes	4096 x 2 bytes		
Corrente media assorbita (mA)	40	50	220	230
Temperatura di esercizio (°C)	Ambiente		55° C	40° C
Dimensioni (mm)	46,2 x 28 x 13	35 x 27,5 x 9,1	32,1 x 18,5 x 7,3	32,5 x 19 x 15,5

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

Tutti i prezzi si intendono IVA inclusa.

SIEMENS SORVEGLIA GLI AUTOBUS DI VERONA

La società tedesca lavora con il comune veneto nell'intento di fronteggiare microcriminalità e atti vandalici e garantire sicurezza ai passeggeri e al personale. La decisione dell'AMT (Azienda Mobilità e Trasporti) di Verona di installare sistemi di videosorveglianza sugli autobus della città ha coinvolto, ad oggi, 60 mezzi; sarà estesa entro breve all'intera flotta. Il progetto è stato realizzato dalla divisione Building Technologies di Siemens. Il sistema, denominato T-Movemat 4, prevede l'installazione di quattro videocamere a bordo dell'autobus: tre sono collocate nella zona passeggeri e una nel posto di guida. Ogni autobus risulta quindi costantemente sorvegliato, grazie a riprese che vengono memorizzate su hard-disk e messe a disposizione della Questura e del Comando dei Vigili. In questo modo la segnalazione di

eventuali reati alle autorità competenti avviene in tempo reale facilitando, se necessario, un intervento. La videocamera posta nell'area di guida del conducente è volta a tutelare la sua incolumità, grazie alla ripresa di immagini che poi vengono salvate in una parte non riscrivibile dell'hard-disk. In caso di necessità l'autista può attivare un pulsante di allarme, inviando, tramite collegamento GSM/GPRS, le registrazioni a un computer situato presso il Comando dei Vigili, permettendo di identificare in breve tempo l'eventuale aggressore. I passeggeri sono avvisati della presenza delle videocamere da un cartello che dovrebbe essere un deterrente per i reati. Le videocamere sono collegate con un'unità di registrazione che acquisisce le immagini, opportunamente crittografate. Maggiori informazioni al sito Internet www.siemens.it.

BELKIN VA IN RETE CON AGERE

Agere Systems annuncia che Belkin Corporation, leader in connectivity solutions, userà il chip Ethernet Gigabit ET1011 nelle sue schede di rete esterne con interfaccia USB 2.0 pensate per potenziare le prestazioni in ambito networking di PC fissi e notebook. Il chip Agere ET1011 PHY permetterà di incrementare la velocità di comunicazione su LAN dei computer provvisti di USB 2.0, consentendo data-rate fino a 300 megabit per secondo, ossia tre volte più veloci delle tradizionali interfacce fast-ethernet 10/100 megabit per secondo. Basato sulla tecnologia TruePHY™ Gigabit Ethernet e sul software di diagnostica dei cavi, l'ET1011 rende possibile realizzare collegamenti privi di errori anche a distanze di 140 metri, consumando appena 750 milliwatt, potenza che è inferiore di circa il 15 % di quella richiesta dai chip concorrenti. Agere

rende disponibile il chip in due package PLCC, a 84-e 68-pin. Quest'ultima versione supporta la Reduced Gigabit Media Independent Interface (RGMII) mentre la 84-pin, oltre alla RGMII, supporta GMII, Ten Bit Interface (TBI) e Reduced Ten Bit Interface. Le dimensioni del chip sono di appena 10x10x1 mm (LxHxP).

Belkin (www.belkin.com) produce periferiche per networking, hardware per VoIP, soluzioni audio e video, dispositivi di interfaccia USB e FireWire, Bluetooth, telefoni cellulari e PDA, oltre a molte altre soluzioni di mobile computing.



SOLUZIONI CAN PER L'ETHERNET

Ixxat propone CAN@net II, un gateway CAN/Ethernet destinato a una gran varietà di impieghi: ad esempio la versione CANnet II/VCI permette la connessione di reti CAN con PC via Ethernet. Le applicazioni vengono così interconnesse mediante la VCI (Virtual CAN Interface) driver library, fornita a corredo, grazie alla quale è possibile usare anche altri tipi di interfaccia, quali PCMCIA, USB e Bluetooth. Dato che VCI affaccia il CAN sul protocollo TCP/IP, CAN@net II può essere usato come qualsiasi componente di rete e indirizzato sia da rete locale che da Internet. Ulteriori informazioni al sito www.ixxat.de.



SONY STABILISCE UN NUOVO STANDARD QUALITATIVO

In arrivo dal colosso giapponese due nuove Handycam DV da 3 Megapixel: a marzo il modello DCR-HC90E con CCD da 3 Megapixel e ad aprile il nuovo camcorder DCR-PC1000E. Tra le caratteristiche che contraddistinguono i nuovi prodotti, obiettivi Zeiss Vario-Sonnar (con zoom ottico 10x) audio Surround 5.1, Wide LCD, ripresa in formato 16:9 nativo e possibilità di registrare foto su Memory Stick DUO. Il nuovo camcorder DCR-PC1000E, dotato di 3 sensori CMOS (ciascuno da 790.000 pixel, da 1/6 di pollice) e di un Processore Enhanced Imaging, è in grado sia di acquisire immagini dotate di grande dinamica per una massima precisione di dettaglio e di offrire un'elevata velocità di elaborazione. Indicato per gli utenti più esigenti, desiderosi di raggiungere la più alta qualità oggi possibile nel campo del video, mette inoltre a disposizione i controlli manuali tipici dell'impiego professionale, a cui si accede tramite un comodo dispositivo per la gestione delle funzioni. Se l'innovativa tecnologia 3 CMOS assicura una precisa gradazione di dettagli e sfumature, in modo da offrire immagini sempre naturali e realistiche, salvaguardando tutti i particolari anche nel caso di forte illuminazione o zone d'ombra, il processore Enhanced Imaging permette una veloce elaborazione delle informazioni relative a un'immagine estesa su un più ampio angolo di visione. La tecnologia 3 CMOS consiste nel riprendere l'immagine con tre sensori distinti, ciascuno dei quali riceve la sola porzione di luce, scomposta da un prisma dicroico, entrante dall'obiettivo, corrispondente al colore assegnatole. La DCR-HC90E monta un unico sensore CCD da 3 Megapixel con filtro a colori primari e un obiettivo Carl Zeiss Vario-Sonnar T, a garanzia di una grande luminosità e sensibilità ai colori e di una perfetta acquisizione delle immagini in movimento. La nuova videocamera è anche in grado di scattare fotografie e di registrare immagini in formato 16:9 nativo, sfruttando tutta la larghezza del sensore CCD e permettendo la visione di immagini con un angolo di visione più ampio rispetto al tradizionale formato di ripresa in 4:3. Questa caratteristica la rende ideale per riprodurre le immagini sia sugli ormai diffusi televisori "widescreen" sia sullo stesso monitor LCD del camcorder, un 2,7 pollici con tecnologia Wide Hybrid. Per maggiori informazioni: www.sony.it.



DTV1000: IL SISTEMA DI ST PER LA TV DIGITALE

Il 6 gennaio STMicroelectronics ha presentato il sistema DTV100, una soluzione per la TV digitale ad alte prestazioni dual-channel (capace di ricevere contemporaneamente due canali); pronto per entrare in produzione e compatibile con tutti gli standard mondiali, è basato sul più recente processore single-chip ST-D2000: fabbricato con la tecnologia CMOS da 90 nm, garantisce attualmente il miglior rapporto costo-prestazioni nel mercato della TV digitale. Integra le funzioni di decoder digitale (set-top-box, STB) e di elaboratore delle immagini. Può decodificare e visualizzare trasmissioni digitali e analogiche, eliminando l'interferenza multi-path e il rumore che proviene dai canali adiacenti. Il dispositivo può decodificare contemporaneamente due segnali con definizione standard ed è in grado di eseguire la decompressione

ad alta definizione (HD), l'elaborazione video e la visualizzazione su schermi CRT o LCD/plasma. Il processore video ad alta definizione garantisce un'eccellente qualità dell'immagine. Il sistema Contour-Sensitive De-interlacing™ (CSDi™), brevettato da ST, elimina gli effetti negativi della codifica interlacciata utilizzando algoritmi sofisticati come l'analisi video pixel per pixel, l'interpolazione adattativa dei pixel e la sovrapposizione dei campi. Un altro fattore importante che migliora la qualità dell'immagine è la riduzione del rumore di cromaticità e luminanza digitale 3D. Le funzioni Perfect PAP™



(Immagine nell'Immagine ad alta definizione) e PIP permettono di utilizzare l' ST-D2000 per realizzare un televisore capace di visualizzare due quadri separati ad alta definizione su un unico schermo, affiancati o uno sopra l'altro. Alle immagini di qualità cinematografica si può associare l'audio ad alta fedeltà ottenuto dai processori STV82x7/8. I DSP (Digital Sound Processor, processori digitali di suono) STV82x7/8 implementano funzioni standard, come enhanced stereo o surround sound 5.1-channel e alcune funzioni speciali per il miglioramento dell'audio come i più avanzati effetti di audio virtuale. STD2000 è compatibile con i principali standard mondiali di TV digitale, come l'europeo DVB (Digital Video Broadcasting) o l'americano ATSC (Advanced Television Systems Committee). Ulteriori info su www.st.com.

È ORA DI SCARICARE I DATI!



Le Pen-Drive USB sono ormai diventate il mezzo principe di trasporto dei dati; bisogna però ricordarsi di portarle con sé quando si esce, altrimenti... Per chi viaggia di frequente esiste una soluzione a prova di sbadato: DiskGO! USB Watch Drive, uno stupendo orologio a quarzo impermeabile al cui interno si trova una memoria flash da 128 o 256 MB, tipo quelle delle Pen-Drive. Un'idea originale che consente di andare in giro senza doversi ricordare di prendere la penna USB, risparmiandosi il fastidio di portarla al collo o tenerla insieme alle chiavi ad accrescere un già ingombrante mazzo. Tramite un mini-connettore e un cavetto in dotazione può essere collegato a Personal Computer e MAC per caricare e scaricare file di testo, MP3 ecc. I cavi in dotazione sono due, così da poterli lasciare sempre collegati al computer e connettersi all'orologio in breve tempo. La memoria può funzionare anche da disco di boot; i requisiti software del computer cui collegarlo sono: CPU Pentium III o superiore, almeno 20 MB di spazio libero, 32 MB di RAM, porta USB 2.0, USB 1.0/1.1, sistema operativo Microsoft Windows 98, Me, 2000, XP, oppure Linux 2.4.0 o successivo, Mac OS 9.x, Mac OS X 10.1.5 o successivo. Visto che le pile degli orologi danno poca tensione, l'alimentazione della memoria viene prelevata dall'USB (assorbimento minore di 50 mA). Le velocità di lettura e scrittura sono 1 MB/s e 750 KB/s. Info su www.meritline.com.

PIÙ AUTONOMIA PER I THINKPAD

Lenovo introduce due nuovi modelli della serie ThinkPad (ex IBM) che stabiliscono un nuovo livello di performance e autonomia: l'adozione di una speciale batteria supplementare garantisce un perfetto funzionamento per 11 ore. Progettati intorno al processore Intel Centrino Duo mobile technology e al Chipset Mobile 945 Express (Front-side bus di 667 MHz) i notebook delle nuove serie ThinkPad X60 e ThinkPad T60 consentono di operare all'interno di locali in cui è presente una wireless wide area network (WWAN). La grande autonomia è stata ottenuta sia con l'aggiunta di una batteria, sia riducendo i consumi (di ben il 37 %) rispetto alle precedenti serie ThinkPad. La tecnologia Intel dual-core assicura alta produttività e migliori prestazioni nelle applicazioni multitasking rispetto ai prodotti basati su singolo processore. Informazioni su www.lenovo.com.

I-STICK: OLTRE LA PEN DRIVE

Più piccola e sottile delle convenzionali Pen-Drive USB, I-Stick (Intelligent Stick) Flash è una conveniente soluzione di Personal Storage ad alte prestazioni. I-Stick è un dispositivo di memorizzazione di massa basato su Smart-Card collegabile, mediante un apposito adattatore, alla presa USB di qualsiasi computer che ne sia provvisto. Sfruttando le interfacce USB 2.0, consente di utilizzare al meglio la velocità di accesso ai dati della sua memoria flash e si candida a fare da concorrente alle Pen-Drive USB. Esteriormente le I-Stick appaiono come delle stecche di plastica terminanti con una scheda riportante i contatti di accesso, somigliante a una Smart-Card; quest'ultima si inserisce in un piccolo lettore da introdurre nella presa USB del computer. I dispositivi della Pro Series garantiscono una velocità di comunicazione pari a 85x o superiore, che permette di leggere e scrivere molto rapidamente: in base al modello, da 7,2 a 25,5 MB al secondo, quindi ben oltre il limite delle tradizionali Pen-Drive USB. Le I-Stick sono disponibili in tre serie: la Standard Series (transfer-rate di 48x, 7,2 MB/s.) ha capacità che vanno da 32 MB a 1 GB; la Pro 85 Series (transfer-rate di 120x, 18 MB/s.) viene fornita con capacità di 256 e 512 MB. Infine, la Pro 170 Series (transfer-rate di 170x, ossia 25,2 MB/s) è disponibile con capacità di 1 GB. Le memorie I-Stick della Pro series vengono fornite con software per Microsoft Windows che consente varie funzioni, quali, ad esempio, la protezione mediante password dei dati introdotti, il partizionamento e la creazione di un settore di boot (per utilizzare la memoria come disco di avvio). Più informazioni su www.synchrotech.com.





Fiera dell'Elettronica e del Radioamatore

**Gonzaga (Mn)
25/26 marzo 2006**

Parco Fiera Millenaria
Orario continuato 8,30 - 18,00



Fiera Millenaria di Gonzaga Srl
Via Fiera Millenaria, 13 | 46023 Gonzaga (MN)
Tel. 0376.58098 - 0376.58388 | Fax 0376/528153
E-mail: info@fieramillenaria.it

<http://www.fieramillenaria.it>

Servizio on-line di vendita moduli Aurel con spedizione in 24/48 ore.

Modello

Ricevitore supereterodina FM 433 MHz

Economico ricevitore supereterodina FM di dati digitali modulati in FSK operante alla frequenza di 433,92 MHz. Elevata selettività e sensibilità garantiscono ottime prestazioni di immunità ai disturbi. Bassa tensione di uscita in assenza di portante. In accordo con le Normative Europee.

RX-4MF1

Euro 15,00

Alimentazione: 5V; consumo: 6mA; frequenza: 433.92MHz; sensibilità: -111dBm; banda passante RF a -3dB: 600kHz; banda passante IF a -3dB: 70 kHz; dimensioni: 40 x 17,4 x 5,5mm.



NEW

Modello

Ricevitore per HCSxxx -106 dBm

Ricevitore a radiofrequenza ad alta sensibilità e basso costo ottimizzato per essere utilizzato con la famiglia HCSxxx Microchip. Condensatore variabile, basso assorbimento, alta immunità ai disturbi di alimentazione e bassa radiazione in antenna. In accordo con le Normative Europee.

AC-RX2

Euro 5,00

Alimentazione: +5V; consumo: 2.5mA; frequenza: 433.92MHz; sensibilità: -106 dBm; dimensioni: 38,1 x 12,3 x 3mm.



NEW

Modello

Trasmettitore SAW 868 MHz con antenna

Modulo trasmettitore SAW con antenna integrata, ideale per applicazioni ove sia richiesta la massima potenza irradiabile e il minimo ingombro in termini di spazio occupato.

TX-8L25IA

Euro 13,00

Alimentazione: 3V; consumo: 2.5mA (con duty cycle 50%); frequenza: 868,3MHz; potenza di uscita (ERP): 25mW; emissione RF spurie: -50dB; frequenza di modulazione: 5kHz; dimensioni: 56 x 18,5 x 5mm.



NEW

Modello

Ricetrasmittitore lungo raggio 2,4 GHz

Il transceiver a lungo raggio XTR-CYP-24 implementa il modulo Cypress CYWM6935 LRTM 2.4GHz DSSS Radio SoC e ne aumenta la potenza RF (ERP) fino a 15 dBm (rispetto a 0 dBm del modulo originale) consentendo di raggiungere una portata di circa 150 metri. Opera nella banda libera ISM (Industrial, Scientific and Medical) a 2.4GHz e offre un sistema radio completo per l'integrazione in dispositivi nuovi o esistenti. Soluzione ideale per automazione domestica e industriale.

XTR-CYP-24

Euro 22,00

Alimentazione: 3.3V; consumo: 0,25 µA (stand-by) - 60mA (RX mode) - 100mA (TX mode); modulazione: GFSK; sensibilità in ricezione: -95dB; potenza RF (ERP) in trasmissione: 10mW; numero di canali: 78; larghezza canale: 1MHz; dimensioni: 35 x 25mm.



NEW

Modello

Ricetrasmittitore multicanale

Il transceiver multicanale XTR-7020A-4 rappresenta una ulteriore soluzione semplice ed economica al problema della ricezione dati in radiofrequenza. Il microprocessore integrato incapsula i dati entranti in logica TTL RS-232 in pacchetti evitando all'utente la necessità di scrivere routine software per la gestione della ricezione dati. L'XTR-7020A-4 permette, tramite la programmazione di registri interni, la gestione della canalizzazione (10 canali sulla banda a 434MHz), della velocità dei dati seriali (9600-19200-38400-57600-115200 bps, impostabili tramite pin di input) e della potenza RF irradiata (da -8 a +10 dBm). Soluzione ideale per automazione industriale, radio modem, controllo accessi.

XTR-7020A-4

Euro 38,00



NEW

Modello

Caratteristiche

	Vdc	Sensibilità RF	Frequenza	Velocità di trasmissione
XTR-434	+5V	-100 dBm	433.92 MHz	100 Kbps
XTR-434L	+5V	103 dBm	433.92 MHz	50 Kbps
XTR-869	+5V	-100 dBm	869.95 MHz	100 Kbps

Euro 38,00

Euro 38,00

Euro 44,00

Ricetrasmittitori radio FM ad alta velocità

Moduli ricetrasmittitori operanti sulle bande 434/869 MHz. Elevata immunità ai campi elettromagnetici interferenti ed elevata potenza di trasmissione. Due limiti di baud-rate per ottimizzare le singole esigenze di ricezione dati. Scambio RX/TX ultraveloce. Conforme alle Normative Europee EN 300 220, EN 301 489 e EN 60950.



Modello

Caratteristiche

	Vdc	Frequenza	Potenza d'uscita	Portata
WIZ-434-SML-IA/5V	+5V - 30 mA	433,92 MHz	3mW	~100 m
WIZ-434-SML-IA/12V	+9÷15V - 30 mA	433,92 MHz	3mW	~100 m
WIZ-869-TRS	+9÷15V - 30 mA	869,85 MHz	3,3mW	~100 m
WIZ-903-A4	+5V - 40 mA	433-434 MHz	0.1÷3mW	~100 m
WIZ-903-A8	+5V - 40 mA	868-870 MHz	0.1÷3mW	~100 m
XTR-903-A4	0÷3V - 40 mA	433-434 MHz	0.15÷10mW	~100 m
XTR-903-A8	0÷3V - 40 mA	868-870 MHz	0.15÷10mW	~100 m

Euro 66,00

Euro 66,00

Euro 70,00

Euro 44,00

Euro 38,00

Euro 38,00

Euro 44,00

Link seriali di ricezione, radiomodem

Moduli ricetrasmittitori ideali per sostituire un collegamento seriale via cavo mediante una connessione wireless RF half-duplex con velocità di trasmissione seriale selezionabile tra 9600, 19200, 57600 e 115200 bps. Disponibili per le bande 434/869 MHz; l'antenna risulta integrata sul circuito stampato.



Informazioni, datasheet e ordini on-line: www.futuranet.it

Temporizzatore per spegnimento automatico

di *Alessandro Sottocornola*

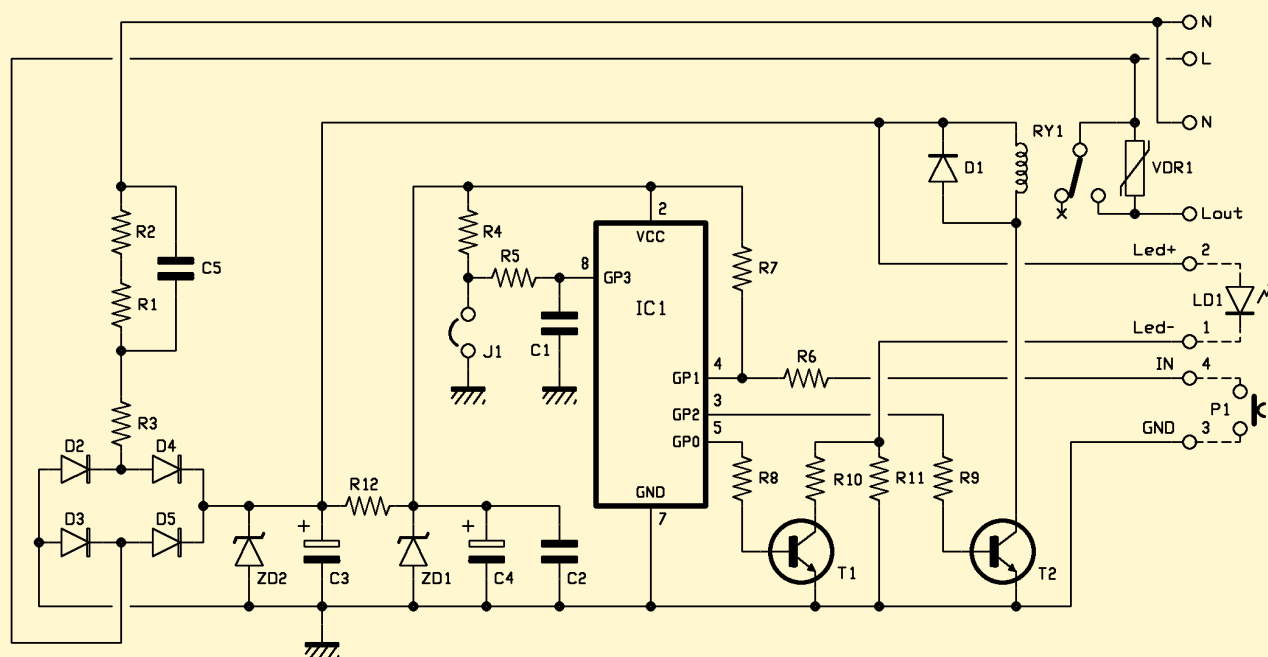


Ideale per programmare lo spegnimento automatico di apparecchi per uso domestico funzionanti a 220 Vac. Va collegato in serie alla loro alimentazione; allo scadere dell'intervallo impostato con gli appositi comandi, sconnette automaticamente la linea.

Quante volte vi è capitato di uscire di casa o lasciare il vostro posto di lavoro e, rientrando, accorgervi che avete lasciato acceso qualcosa? Sicuramente almeno una volta, perché anche ai più attenti e meticolosi succede di scordarsi acceso il televisore, l'impianto hi-fi, uno strumento di misura, piuttosto che una piantana alogena o altro ancora. Tra le poche o tante volte, ci auguriamo che gli effetti della propria disattenzione si siano limitati a qualche soldo in più sulla bolletta della "luce" e non a qualcosa di peggio; già, perché talvolta lasciare alimentati determinati utilizza-

tori senza la dovuta sorveglianza può portare ad incidenti dalle conseguenze anche gravi. Ad esempio, macchine utensili, stufe elettriche, ferri da stiro per citarne alcuni, possono procurare non pochi grattacapi; problemi che possono essere evitati disponendo lo spegnimento automatico mediante un apposito timer da collegare in serie alla linea di alimentazione. Nelle pagine successive vogliamo proporvi proprio una soluzione del genere: parliamo di un temporizzatore di power-off che alimenta, mediante lo scambio di un relé, l'utilizzatore del quale si desidera lo spegnimento automatico; ➤

SCHEMA ELETTRICO



allo scadere del tempo impostato, lo scambio torna ad aprirsi e sconnette la linea di alimentazione. Il circuito è equipaggiato con un relé capace di commutare tensioni dell'ordine dei 230 V e correnti di 10 ampere; è quindi adatto ad alimentare dispositivi funzionanti in alta tensione, che assorbano al massimo 10 A: ad esempio, una stufa elettrica da 2.000 watt. Laddove servisse gestire carichi più esigenti, lo scambio potrà essere usato per alimentare la bobina di un servo-relé a 230 V con portata adeguata. Ma vediamo più da vicino di cosa si tratta, osservando lo schema elettrico del circuito, che ci mostra come

la funzione venga ottenuta con pochissimi componenti elettronici.

Schema elettrico

La temporizzazione viene implementata da un piccolo microcontrollore PIC10F200 di casa Microchip: si tratta di un micro con architettura ad 8 bit, contenente una piccola memoria flash nella quale si trova il programma, che è molto semplice e consta essenzialmente di un timer programmabile mediante un pulsante e un jumper; con lo stesso pulsante si avvia e si arresta la temporizzazione.

Quando si fornisce alimentazione al circuito, il programma residente nel

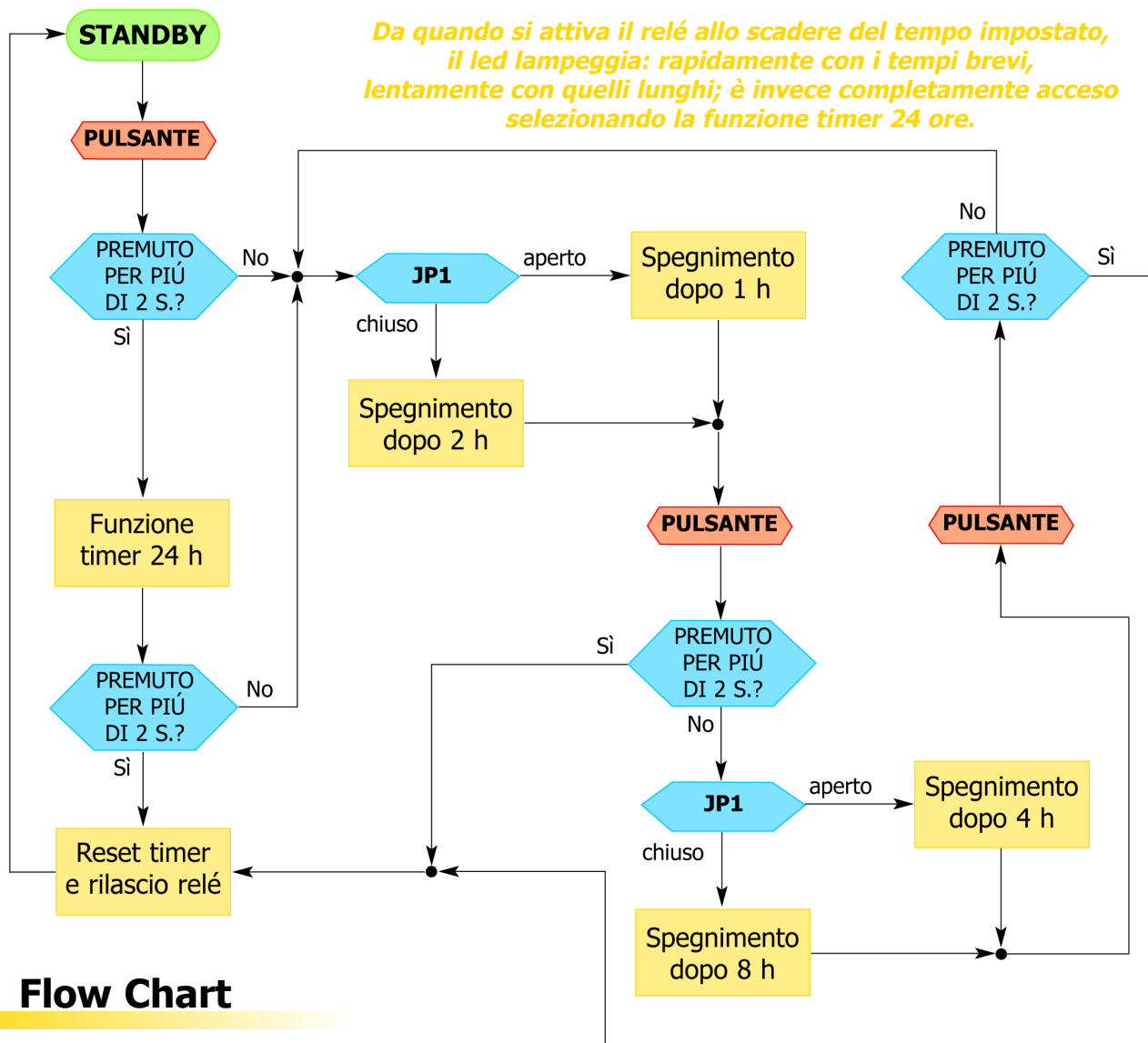
PIC per prima cosa inizializza le linee di I/O, impostando GP1 e GP3 come ingressi, e GP0 e GP2 come uscite; mediante la prima legge lo stato del jumper di selezione JP1, mentre con GP3 rileva la condizione del pulsante, condizione che, lo vedremo tra breve, determina le funzioni del temporizzatore.

GP0 pilota, mediante il transistor T1, il led di segnalazione LD1; GP2 si occupa dell'attivazione del relé. T1 e T2 sono entrambi NPN che usiamo da interruttori statici; per mandarli in conduzione, le corrispondenti linee di I/O del microcontrollore assumono il livello logico alto, in modo da polarizzarne le basi fino alla saturazione.

Particolare attenzione la merita il led, che, in base alla condizione di funzionamento del circuito, viene acceso a piena luce o a intensità ridotta: quando il software del PIC deve farlo illuminare al massimo, pone ad 1 logico GP0, cosicché T1 si vede polarizzata la base per effetto della R8 e conduce tra collettore ed emettitore, determinando nel diodo lo scorrimento di una corren-

- Spegnimento programmato del carico;
- Intervallo selezionabile 1, 2, 4, 8, 24 h;
- Selezione intervallo tramite pulsante;
- Segnalazione a led;
- Stadio di uscita a relé;
- Corrente commutabile: 10 A (@ 230 Vac);
- Alimentazione diretta a 230 V/50 Hz.

Flow Chart



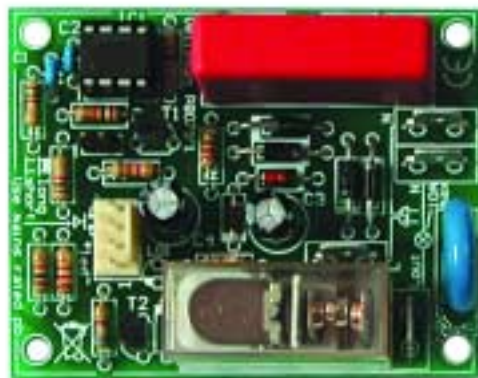
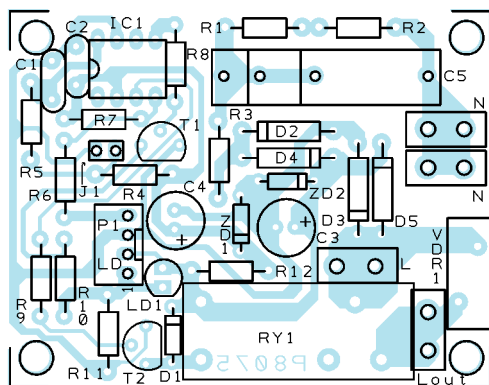
te limitata dal valore resistivo della R10; per ottenere la luce ridotta, il micro pone GP0 a livello basso, cosicché T1 è interdetto e nel led scorre una corrente che attraversa, stavolta, R11. In quest'ultimo caso l'intensità è minore perché dovuta alla R11, il cui valore è decisamente più alto di quello della R10.

Continuando con l'analisi dello schema, notate il diodo D1: serve a cortocircuitare le extratensioni inverse che si generano ai capi della bobina all'interdizione del T2 e che, se non bloccate, danneggerebbero la giunzione base-collettore di tale transistor. Sempre in tema di protezioni, siamo stati previdenti

anche per quel che riguarda lo scambio del relé: tra i contatti usati per alimentare l'utilizzatore abbiamo posto il varistore VDR1; si tratta di un componente la cui resistenza è praticamente infinita fin quando la tensione ai suoi capi si mantiene sotto quella di soglia (300 V, nel nostro caso...) e va praticamente in cortocircuito se invece lo si sottopone a una differenza di potenziale superiore. Scopo del varistore è proteggere lo scambio del relé da archi elettrici che possono formarsi se, al momento dell'apertura del contatto C/NA, l'utilizzatore presenta una tensione eccedente quella tollerata dallo scambio stesso

(240÷260 V); tale situazione può verificarsi quando il relé deve gestire carichi fortemente induttivi, i quali, all'apertura del circuito, reagiscono generando extratensioni inverse causate dalla natura inerziale degli induttori nei riguardi della corrente, caratteristica che, all'apertura della linea di alimentazione, li porta a cercare di mantenere le condizioni precedenti e quindi lo stesso flusso di corrente.

Detto ciò, passiamo al funzionamento del programma di gestione: dopo l'inizializzazione, il micro legge una sola volta GP3, allo scopo di impostare le temporizzazioni corrispondenti ai comandi che ➤



ELENCO COMPONENTI:

R1: 330 kohm 1%
R2: 330 kohm 1%
R3: 220 ohm 1%
R4: 47 kohm
R5, R6: 1 kohm
R7÷R9: 10 kohm
R10: 2,2 kohm
R11: 6,8 kohm
R12: 3,9 kohm
C1: 100 nF multistrato

C2: 100 nF multistrato
C3: 100 µF 35 VL elettrolitico
C4: 100 µF 35 VL elettrolitico
C5: 470 nF 400 VL poliestere
IC1: PIC10F200 (VK8075)
ZD1: zener 5V1 500mW
ZD2: zener 24V 1,3W
D1: 1N4148
D2÷D5: 1N4007
LD1: led rosso (vedi testo)

T1, T2: BC547
VDR1: 14D471K
RY1: Relé 24Vdc 10A

Varie:
- Zoccolo 4+4 pin
- Pulsante NA
- Jumper
- Strip maschio 4 pin
- Circuito stampato

L'utente darà con il pulsante; quest'ultimo viene poi letto continuamente (tramite GP1). Fin quando P1 resta a riposo, il micro segnala la condizione di stand-by (relé a riposo) con l'accensione del led a mezza luminosità. Quando rileva la chiusura del tasto (mediante il partitore formato da R6 ed R7) conta il tempo per il quale la linea GP1 resta a livello basso; se lo zero logico dura meno di due secondi, il timer avvia la temporizzazione più corta tra quelle definite dal jumper. Riguardo al ponticello, va precisato che la sua condizione va stabilita a circuito spento; data l'alimentazione, il led lampeggerà due volte lentamente se il jumper verrà trovato chiuso, condizione che imporrà l'intervallo lungo (2 o 8 ore) oppure rapidamente, se, invece, JP1 sarà stato lasciato aperto, condizione che attiva nel microcontrollore l'in-

tervallo breve (1 o 4 h). Preciso ciò, torniamo all'attivazione del timer: abbiamo detto che il relé scatta quando si preme il pulsante; la prima pressione breve (per meno di 2 s.) seleziona la più corta delle durate, ossia 1 o 2 ore a seconda che JP1 sia, rispettivamente, aperto o chiuso; la condizione è evidenziata dal rapido lampeggiare del led presente nel circuito, che si spegnerà a tempo esaurito (contestualmente al rilascio del relé). Una successiva pressione per meno di due secondi imposta la durata più lunga (4 h con JP1 aperto, 8 h con JP1 chiuso); la temporizzazione lunga è segnalata dal lampeggio lento del led fino allo scadere del tempo o al cambio di programma mediante P1. Un ulteriore intervento per meno di 2 secondi provoca il reinserimento dei tempi corti e così via: insomma, la pressione del pulsante per meno

di 2 s. fa passare dalle durate corte (1 o 2 h) alle lunghe (4 o 8 ore) e viceversa.

Ma cosa accade se il tasto viene premuto a lungo, ossia oltre i 2÷3 secondi? Ebbene, facendolo quando il timer è a riposo (led acceso a luce fioca e relé diseccitato) si imposta la funzione timer 24 h: il relé scatta e viene diseccitato dopo un giorno esatto; l'inserimento di tale opzione viene segnalato dall'accensione del led a luce fissa e alla massima intensità. Durante la funzione timer 24 ore un'ulteriore pressione prolungata del tasto provoca la disattivazione del relé e l'annullamento della temporizzazione impostata. Invece, se il tasto viene premuto per almeno 2÷3 secondi durante le temporizzazioni 1, 2, 4, 8 ore, si provoca il reset del temporizzatore e la disattivazione immediata dell'utilizzatore, ovvero

l'istantaneo rilascio del relé. In tutti i casi, l'annullamento del conteggio del timer e il rilascio del relé sono evidenziati dall'accensione del led a luce ridotta, che coincide con la condizione di stand-by.

Questo è quanto riguarda le funzioni e il comando del temporizzatore. Vediamo adesso il discorso inerente all'alimentazione: guardando lo schema elettrico notate che l'intero circuito funziona direttamente con la tensione di 220 Vac e che lo scambio del relé non è libero ma, piuttosto, è previsto che commuti la fase sull'utilizzatore; ne deriva che il dispositivo di cui controllare lo spegnimento deve funzionare con la rete di distribuzione elettrica. Abbiamo risparmiato il trasformatore optando per una soluzione più leggera ed economica, e meno ingombrante: praticamente raddrizziamo direttamente l'alternata con il ponte di Graetz formato dai diodi D2, D3, D4, D5, in serie al quale è stata posta un'impedenza costituita da R1, R2, R3 e C5; quest'ultima serve a far cadere la tensione in eccesso e a funzionare da resistenza zavorra del diodo Zener ZD2. In sostanza, sfruttiamo la reattanza capacitiva del C5 che, alla frequenza della rete elettrica, vale $X_c = 10^6 / (6,28 \times 50 \times 0,47) = 6.775$ ohm, per far cadere la tensione in eccesso; a determinare la caduta concorre anche la R3, scopo della quale è, però, evitare eccessi di corrente nello zener ZD2 nel caso in cui il circuito venga collegato alla

rete nel momento in cui la tensione sinusoidale ha il valore massimo. Infatti, in tal caso il condensatore, essendo normalmente scarico, si lascia attraversare dal picco di tensione. Le resistenze R1 ed R2 servono a scaricare C5 quando il circuito viene privato dell'alimentazione: in tal modo si evita di prendere pericolose scosse toccando le piste dello stampato quando, vedendo il dispositivo sconnesso dalla rete, si è portati a pensare di poterci mettere su le mani. Dal ponte raddrizzatore, ossia tra i catodi di D4 e D5 e gli anodi di D2 e D3, escono impulsi sinusoidali unidirezionali, ad una frequenza doppia di quella della rete elettrica; tramite ZD2 gli impulsi vengono cimitati a 24 volt e vanno a caricare l'elettrolitico C3, tra le cui armature si trova la tensione che alimenta la bobina del relé. Un secondo zener, siglato ZD1, con l'ausilio della sua resistenza zavorra limita a 5 V la tensione ai propri capi e ricava così ciò che serve a far funzionare il microcontrollore e il led.

Realizzazione pratica

Vediamo adesso come costruire il timer di spegnimento, iniziando con il circuito stampato, facilmente ottenibile per fotoincisione; la pellicola occorrente si ricava da una stampa su carta da lucido o acetato (ma anche su un foglio bianco, avendo l'accortezza di prolungare di un paio di minuti il tempo di esposizione agli UV nel bromogra-

fo) della traccia lato rame disponibile nel sito Internet www.elettronica.it. Incisa e forata la basetta, disponetevi i componenti iniziando con le resistenze e proseguendo con i diodi, dei quali vi rammentiamo che il catodo è il terminale vicino alla fascetta colorata sul corpo; seguendo il piano di montaggio, inserite e saldate lo zoccolo per il microcontrollore, i due transistor (T1 deve stare con la parte piatta rivolta alla resistenza R4, mentre il lato delle scritte del T2 deve affacciarsi al diodo D1) e i condensatori, che suggeriamo di inserire in ordine di altezza. Per il ponticello montate sul c.s. una strip di due punte a passo 2,54 mm, che, all'occorrenza, chiuderete con un jumper adatto. Infine, sistemate il varistore (non ha polarità quindi potete metterlo nel verso che preferite...) il relé (entra solo nel verso giusto) e il pulsante, che prevediamo sia del tipo normalmente aperto con led spia interno: il contatto del tasto va connesso con due fili alle piazzole (3 e 4) corrispondenti, ossia alle due che si trovano vicine alla R4; il led va invece collegato alle due restanti, rammentando che il catodo corrisponde al contatto 1. Se usate un pulsante tradizionale (normalmente aperto) il led può prendere posto sul circuito stampato, nel quale sono già state previste apposite piazzole, ovvero sul pannello dell'eventuale contenitore nel quale racchiuderete il timer, collegato con due spezzoni di filo; la piazzola del catodo è quella ➤

Per il

MATERIALE

Il progetto descritto in queste pagine è disponibile in scatola di montaggio al prezzo di 16,00 Euro. Il kit comprende tutti i componenti, il circuito stampato forato e serigrafato, le minuterie ed il microcontrollore già programmato. 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>**

IL FIRMWARE

```

DEFINE OSC 4
DEFINE OSCCAL_1K 1

@          DEVICE MCLR_OFF
@          DEVICE BOD_OFF

SYMBOL J1=GPIO.3      'JUMPER
SYMBOL LED=GPIO.0     'LED
SYMBOL RELE=GPIO.2    'RELE
SYMBOL PULS=GPIO.1    'PULSANTE

TMP        VAR        BYTE
TMP1       VAR        BYTE
PRESSIONE  VAR        BYTE
TEMPO     VAR        BYTE
TIME      VAR        WORD

OPTION_REG=%10000000

clear

TEMPO=0

MAIN:

    PRESSIONE=0
    WHILE PULS=0 AND PRESSIONE<21 'FINO A QUANDO NON
RAGGIUNGO I 2 SEC 0 IL PULSANTE NON VIENE RILASCIATO
RIMANGO NEL CICLO
        PRESSIONE=PRESSIONE+1
        PAUSE 100
    WEND

    IF PRESSIONE <>0 THEN
        HIGH LED
        PAUSE 1000
        IF PRESSIONE>=20 THEN 'SE IL PULS è STATO
PREMUTO PER 2 SEC VADO IN 24H
            LOW LED
            IF TEMPO=0 THEN 'SE ERO GIÀ IN 24H
VADO A RIPOSO
                TEMPO=24
            ELSE
                TEMPO=0
                LOW RELE
            ENDIF
        ELSE
            IF J1=0 THEN 'SE IL JUMPER è CHIUSO 4H
                        0 8H
                IF TEMPO=4 THEN
                    TEMPO=8
                ELSE
                    TEMPO=4
            ENDIF
        ENDIF
    ENDIF

    IF TEMPO=2 THEN 'SE IL JUMPER è APER
                        TO 1H 0 2H
        TEMPO=1
    ELSE
        TEMPO=2
    ENDIF
ENDIF
ENDIF
ENDIF

FOR TMP1=1 TO TEMPO
    HIGH RELE
    GOSUB ORA
    IF PULS=0 THEN
        GOTO MAIN
    ENDIF
NEXT TMP1
LOW RELE

GOTO MAIN

ORA:
    TIME=0
    WHILE PULS=1 AND TIME<360 'SE IL PULSANTE NON è
PREMUTO E NON è PASSATA 1 H RIMANI NEL CICLO DI WHILE
        PAUSE 100
        TIME=TIME+1
        IF TEMPO=24 THEN
            HIGH LED
        ELSE
            IF TEMPO=1 OR TEMPO=4 THEN
                TMP=TIME//10 'SE È PASSATO UN
                            SECONDO
                IF TMP=0 THEN
                    TOGGLE LED 'FACCIO LAMPEGGIARE
                                IL LED
                ENDIF
            ELSE
                TMP=TIME//20 'SE SONO PASSATI DUE
                            SECONDI
                IF TMP=0 THEN
                    TOGGLE LED 'FACCIO LAMPEGGIARE
                                IL LED
                ENDIF
            ENDIF
        ENDIF
    ENDIF
    WEND
    LOW LED
RETURN
    
```

più vicina al relé. Per completare il montaggio dei componenti inserite il microcontrollore nel suo zoccolo, badando che la tacca di riferimento sia rivolta al C2. Ora pensate al cablaggio: i due capi del cordone di alimentazione da collegare alla rete vanno inseriti e stagnati nei contatti L ed N (rispettivamente neutro e fase) scegliendone due a caso; nei restanti L ed N connettete i cavi che portano all'utilizzatore del quale

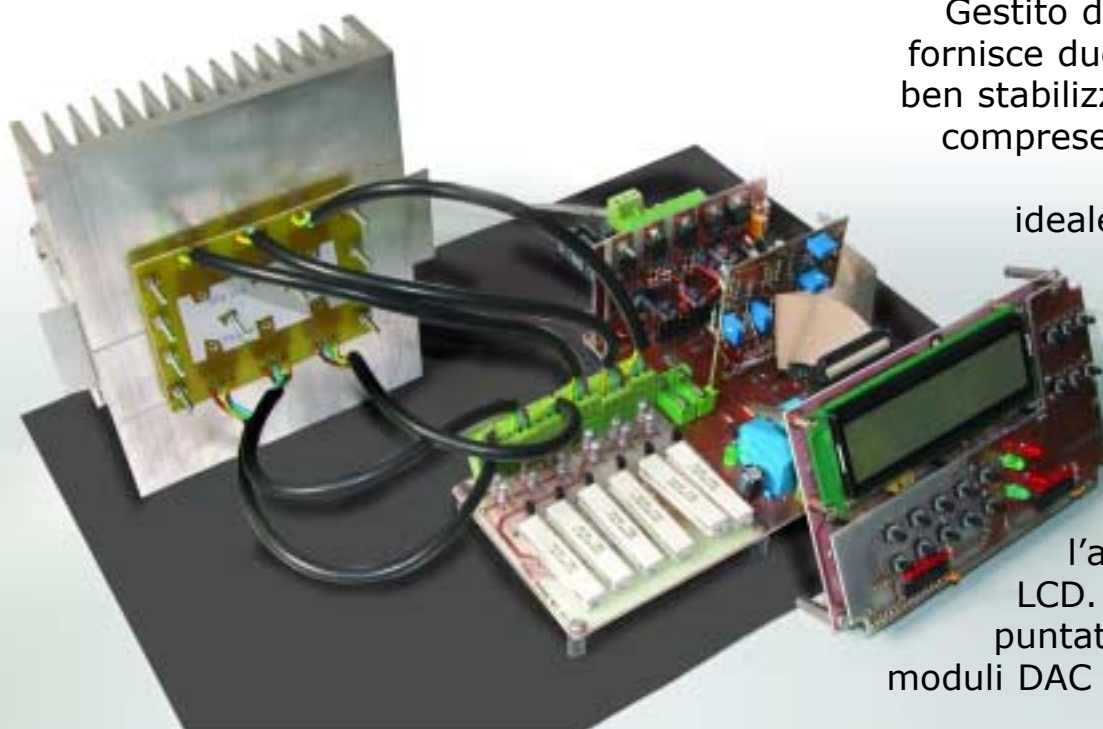
controllare lo spegnimento. Se intendete usare il timer con apparati che assorbono fino a 5 A utilizzate fili aventi sezione di 1,5 mmq; altrimenti optate per una sezione di 2,5 mmq. Oltre i 5 A è anche consigliabile stagnare bene tutta l'area delle piazzole dei piedini dello scambio del relé e quella dei contatti N. Per agevolare le connessioni potete saldare nelle piazzole L ed N delle lamelle fast-on da stampato,

quindi intestare i cavi da collegare (rete e utilizzatore) con dei fast-on femmina volanti.

Assemblato il tutto, potete decidere se montarlo in un apposito contenitore, oppure inserirlo direttamente nell'apparecchio da controllare. In tutti i casi ricordate che il circuito è alimentato direttamente con i 220 volt, quindi occorre maneggiarlo con cura e mai quando è collegato alla rete, pena una forte scossa!

Alimentatore duale professionale

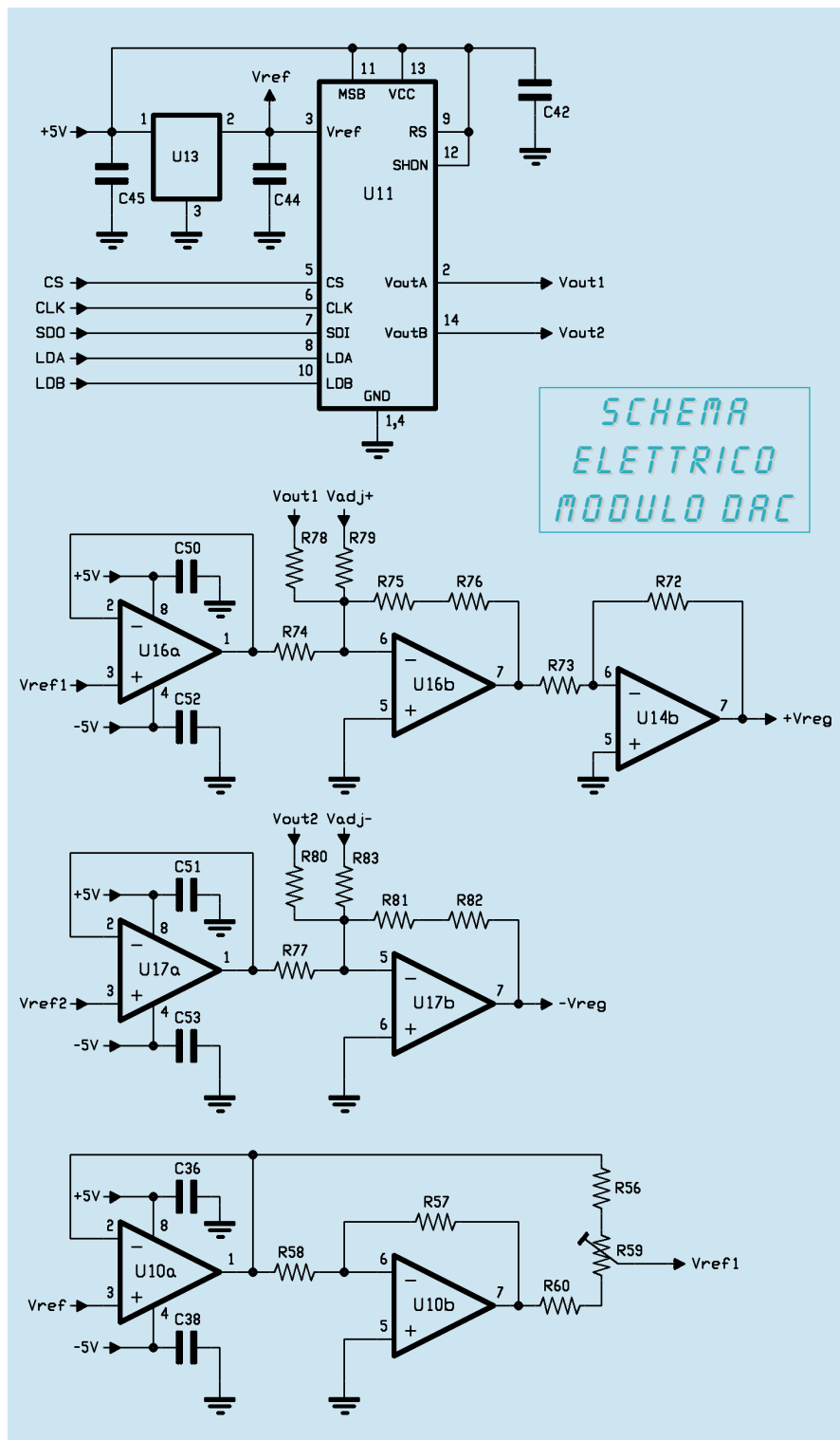
di Matteo Destro



Gestito da microcontrollore, fornisce due tensioni continue ben stabilizzate e simmetriche comprese tra ± 1 e ± 36 volt rispetto a massa; ideale per il laboratorio, può erogare fino a 3A per ramo. I valori si impostano mediante una pulsantiera e con l'ausilio di un display LCD. In questa seconda puntata ci occupiamo dei moduli DAC e Microprocessore.

Eccoci puntuali all'appuntamento con la seconda parte riguardante l'alimentatore stabilizzato gestito da microcontrollore; dopo aver analizzato lo schema a blocchi dell'insieme e della sezione DAC, visti i dettagli teorici e costruttivi dell'alimentatore principale, del modulo di filtro e della sezione di potenza, proseguiamo analizzando i moduli mancanti, ossia quello del microcontrollore, il D/A converter e la pulsantiera/interfaccia con l'utente. Iniziamo con il DAC, che occupa un posto di rilievo nell'architettura dell'insieme. Si tratta del circuito che provvede alla

generazione delle tensioni di riferimento per gli operazionali della sezione di potenza e quindi di quelle che determinano le differenze di potenziale erogate dalle uscite dell'intero alimentatore; il DAC ricava detti potenziali basandosi sui dati numerici che il modulo a microcontrollore gli invia attraverso l'interfaccia seriale SPI di cui dispone. I potenziali sono frutto dell'impostazione operata manualmente dall'utente tramite il pannello di comando e vengono "aggiustati" in base alle risultanze dei circuiti di retroazione che, dalla sezione di potenza, portano al microcontrollore le ➤



informazioni sulle tensioni effettivamente erogate dall'alimentatore. Al micro giungono anche le informazioni sulla corrente assorbita dai carichi, come si desume dallo schema della sezione di potenza pubblicato nella prima puntata. Si noti che i circuiti di "sense" usati per rilevare le correnti erogate dalle uscite

determinano cadute di tensione sulle resistenze di shunt, cadute che vengono compensate dal micro, il cui software opera per formare i potenziali di riferimento da mandare alla sezione di potenza.

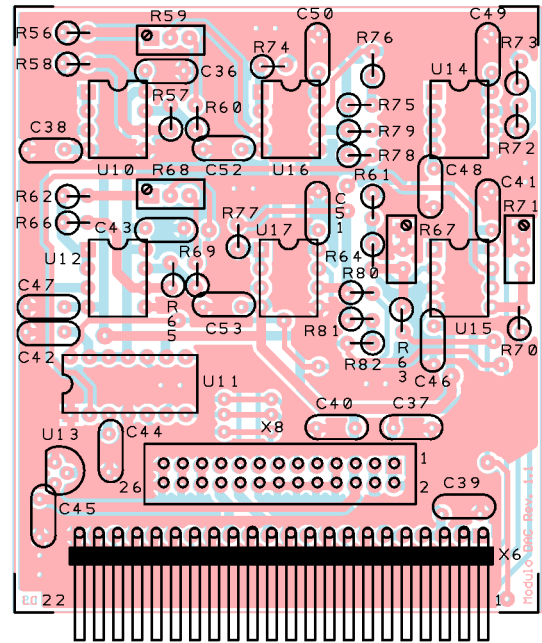
Il modulo DAC

Insieme all'unità a microcontrollo-

re, è il cuore del sistema: in esso è contenuta l'elettronica di precisione necessaria ad ottenere le tensioni di controllo per il ramo positivo e per il ramo negativo del blocco di potenza del nostro alimentatore, tensioni che vengono ulteriormente amplificate dagli operazionali posti sulla scheda di potenza. Il componente centrale del modulo è il convertitore digitale analogico (DAC) il quale, dato un segnale di tipo digitale al suo ingresso, restituisce all'uscita una componente analogica. Il DAC utilizzato è un AD7395 (integrato U11) dell'Analog Devices e presenta le seguenti caratteristiche: 10 bit di risoluzione, 2 canali di conversione indipendenti, interfaccia seriale SPI a 3 fili (*Enable*, *Clock* e *Data*) pin di shut-down e reset. Il suo schema a blocchi riassume l'elettronica presente al suo interno (la parte digitale si trova tutta a sinistra mentre quella di tipo analogico è sulla destra).

L'interfaccia seriale è composta da un semplice shift-register per gestire le informazioni inviate dal microprocessore, più due latch che servono da buffer di memoria. Quando il dato da convertire è stato completamente trasferito nello shift-register, è possibile passarlo a uno dei due latch, scelto in base al canale di uscita sul quale si vuole trasferire l'informazione. L'abilitazione di uno dei due latch avviene attraverso i pin di controllo LDA o LDB, entrambi attivi a livello basso. Il sottostante diagramma temporale (riferito a un DAC a 12 bit) riassume in modo semplice come trasferire il dato dal PIC al convertitore DAC. Naturalmente il discorso vale anche per un DAC a 10 bit come nel nostro caso.

Il microprocessore deve abilitare il DAC portando il CS a livello logico basso, dopodiché è possibile inviare i dati. Su ogni fronte di salita del clock viene trasferito all'uscita un bit della stringa; come si nota, si



ELENCO COMPONENTI:

R56, R60, R62: 4,7 kohm 1%
 R57, R58: 100 kohm 1%
 R59: Trimmer multigiri 100 ohm
 R61: 100 kohm 1%
 R63: 1,2 Mohm 1%
 R64÷R66: 100 kohm 1%
 R67: Trimmer multigiri 2 kohm
 R68: Trimmer multigiri 100 ohm
 R69: 4,7 kohm 1%
 R70: 1,2 Mohm 1%

R71: Trimmer multigiri 2 kohm
 R72÷R83: 100 kohm 1%
 C36÷C43: 100 nF 100 VL ceramico
 C44: 1 µF 25 VL tantalio
 C45÷C53: 100 nF 100 VL ceramico
 U10, U12: MCP602
 U11: AD7395
 U13: MCP1525
 U14÷U17: MCP602

Varie:

- Strip maschio 90° 22 pin
- Connettore maschio per flat cable 26 poli
- Zoccolo 4+4 (6 pz.)
- Zoccolo 7+7
- Circuito stampato

Se non specificato, tutte le resistenze si intendono ad 1/4W 5%

invia per primo il bit più significativo, seguito da tutti gli altri fino ad arrivare al meno significativo.

Conclusa la trasmissione del dato, si riporta il CS a livello logico alto. Ora si può abilitare, attraverso LDA o LDB, il registro in cui memorizzare la stringa inviata, la quale verrà convertita in un valore analogico in uscita dal canale A o dal canale B: ciò dipende dalla selezione fatta. La tensione sull'uscita del DAC dipende, oltre che dal valore digitale inviato lungo l'interfaccia SPI, anche dalla Vref applicata sul pin 2 del DAC; quest'ultima viene ricavata dall'integrato MCP1525 (U13) che è un regolatore di alta precisio-

ne. Possiamo dire che la tensione di uscita dell'AD7395 risulta legata a quella fornita dall'U13 dalla relazione: $V_{out} = (V_{ref} \cdot D) / 2^{10}$. In essa D è il valore digitale passato al convertitore DAC; quindi, data una risoluzione di 10 bit, il minimo valore ottenibile ($D=1$) è pari a 2,4 millivolt, mentre il massimo (che corrisponde a $D=1.024$) è 2,5 volt esatti.

Come già accennato, il convertitore è dotato di un pin di reset (RS) e uno di ShutDown (SHDN): il primo serve per resettare il DAC portandolo al valore 0 oppure al valore $V_{out}/2$ (mezza scala), in relazione al valore assunto dal pin MSB. Il

secondo porta il DAC nella condizione di *sleep*, nella quale il componente assorbe una corrente dell'ordine dei 100 nA. I criteri sono entrambi attivi a livello basso; nel nostro caso non facciamo uso delle funzioni corrispondenti, infatti li abbiamo lasciati entrambi vincolati alla linea del +5V.

Il segnale in uscita dal DAC non può essere inviato così com'è al modulo di potenza, ma va opportunamente amplificato in tensione per raggiungere i livelli desiderati: a ciò provvedono degli operazionali collegati in cascata al fine di ottenere un guadagno pari a 20,484 volte; in particolare, si ha una prima pream-

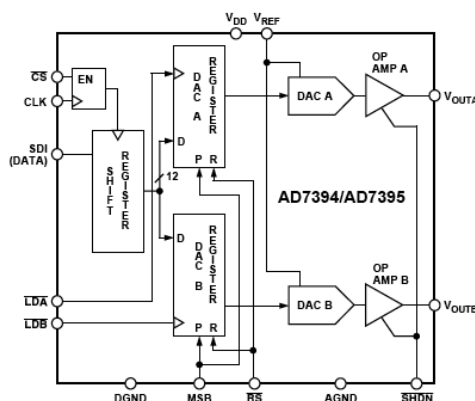
plificazione di 2 volte e successivamente, sulla scheda di potenza, una seconda amplificazione di 10,242. La preamplificazione viene operata da un operazionale sommatore invertente che serve a ricavare tensioni di controllo condizionate dalle vicende delle uscite: più esattamente, ogni sommatore produce potenziali che dipendono dalla Vref1 o Vref2 (tensione di regolazione fine per la minima tensione in uscita dall'alimentatore) dalla Vout1 o Vout2 (tensione proveniente dal primo canale del DAC) e dalla Vadj+ o Vadj- (tensione di retroazione che serve per la regolazione fine del valore massimo in uscita dall'alimentatore). La Vref1 è ottenuta partendo dalla Vref (riferimento costante fornito dall'MPC1525)

e ponendo un partitore simmetrico tra l'uscita dell'amplificatore invertente U10b e quella dell'U10a; siccome quest'ultimo è un buffer non invertente e fornisce esattamente la Vref, mentre U10b guadagna 1, teoricamente il partitore R56/R59/R60 è alimentato con due potenziali simmetrici rispetto a massa. Quindi, portando il cursore del trimmer R59 in posizione centrale Vref1 (riferimento per la sezione positiva) deve valere zero volt.

Vadj+ viene generata dall'operazionale U15, mediante una rete che legge la tensione dell'uscita positiva presente dopo la resistenza di shunt e ne riporta sull'ingresso dell'U16a una piccola

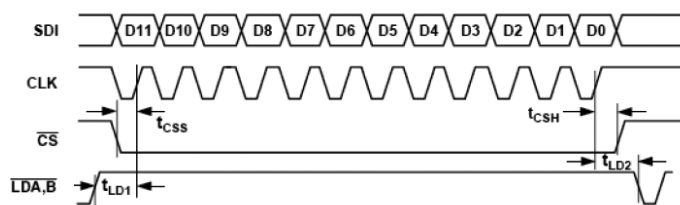
porzione, opportunamente corretta mediante la rete R63/R67. U16a è anch'esso un buffer non invertente che restituisce una componente inalterata nel valore, sommata alle altre due in ugual misura (i resistori sono tutti da 100 kohm); il som-

matore U16b amplifica di due volte la somma dei potenziali, ricavando un potenziale negativo (l'operazionale funziona da sommatore invertente) che viene invertito dall'U14 al fine di ottenere una grandezza



Lo schema a blocchi dell'AD7395 montato nel modulo DAC.

positiva che verrà poi amplificata dall'operazionale in configurazione non invertente (U16) presente sulla scheda di potenza. Osservate che U16a serve a rendere più precisa la somma: porta una tensione proveniente da una fonte a bassissima resistenza, il cui valore non è paragonabile con la R74; se collegassimo il cursore dell'R59 direttamente a tale resistenza, la somma verrebbe falsata perché, come insegna la teoria degli operazionali, il guadagno



Il protocollo di comunicazione tra microcontrollore e AD7395.

di un sommatore invertente relativamente a una certa tensione è dato dal rapporto tra il valore del resistore di retroazione e di quello che porta la tensione da sommare. Se non vi fosse il buffer, la componente dovuta a Vref sarebbe affetta dal-

l'aggiunta di quella risultante dal partitore R56/R59/R60 sommata al valore di R74 e non dipenderebbe dalla sola R74, come voluto.

Chiarito questo punto, vediamo che, essendo la componente Vadj+ strettamente dipendente da una porzione cambiata di segno della tensione dell'uscita positiva, il sommatore è in grado di compensare le fluttuazioni presenti all'uscita dell'alimentatore a causa delle variazioni di carico e per effetto della caduta sullo shunt usato per rilevare la corrente erogata; supponete, ad esempio, che la tensione si abbassi: la componente ricavata da U15a diviene meno negativa e la somma di essa con Vout1 e la componente in arrivo da U16a determina all'uscita di U14b un potenziale più alto.

V+reg incrementa e forza un aumento del potenziale di riferimento della sezione positiva del *Modulo Potenza* che va a compensare l'abbassamento all'origine della correzione. Al contrario, se per una diminuzione dell'assorbimento l'uscita positiva tende a dare un potenziale più alto del previsto, la componente Vadj+ diviene più negativa e la sua somma con la componente derivante da Vref e la Vout1 si abbassa; così la V+reg uscente dall'U14b si abbassa e determina l'abbassamento della tensione inviata al *Modulo Potenza* e la conseguente compensazione dell'incremento all'origine del fenomeno.

Quello che accade nella zona di regolazione negativa è simile: cambiano solo gli operazionali interessati e le polarità delle tensioni. Senza dilungarci troppo diciamo che la Vadj- è ricavata prendendo una porzione della tensione dell'uscita negativa a valle

dello shunt del ramo corrispondente (*Modulo Potenza*) e invertendola di polarità con il buffer invertente U15b fino a renderla positiva; Vref2 è ottenuta con il partitore R62/R68/R69 collegato tra l'uscita del buffer U12a e quella dell'U12b. Siccome quest'ultimo lavora in configurazione invertente a guadagno unitario ed eroga un potenziale uguale a Vref, ma di segno opposto, come per la corrispondente sezione positiva il trimmer da 100 ohm deve fornire teoricamente 0 volt, eventualmente aggiustabili in base alle esigenze di taratura. Dunque, U17a riporta al sommatore U17b la componente di riferimento Vref2, che viene sommata algebricamente a Vout2 e Vadj-; notate che in questo caso la tensione risultante dall'uscita del sommatore non viene invertita, perché, dovendo pilotare il blocco negativo (U17) del *Modulo Potenza*, è giusto che resti negativa. Gli operazionali usati per il *Modulo DAC* sono degli MCP602 prodotti da Microchip; la loro caratteristica saliente è una tensione di offset di ingresso molto bassa, che li rende adatti al nostro scopo: abbiamo infatti a che fare con potenziali di basso livello che possono essere alterati anche da lievi offset. Ogni integrato contiene due operazionali; siccome nell' U14 uno non viene usato (perché l'inversione serve solo nella sezione di regolazione del ramo positivo) ne portiamo gli

ingressi a un valore fisso, in modo che la sua uscita non si metta ad oscillare. Torniamo un attimo alla sezione di riferimento positivo, per dire che il trimmer R59 permette di ottenere un valore di tensione che varia da un minimo negativo a un massimo positivo; ruotandone il cursore si regola la minima tensione presente all'uscita dell'alimentatore e si compensa l'errore introdotto dal convertitore DAC e l'eventuale offset degli operazionali usati per l'amplificazione del segnale. La massima tensione presente sull'uscita dell'alimentatore viene invece determinata dal trimmer R67. Per il ramo negativo, il minimo potenziale erogato si imposta con R68, mentre il massimo si regola agendo sul cursore dell'R71.

Il Modulo Microprocessore

Bene, sperando che abbiate compreso la funzione del *Modulo DAC*, passiamo a descrivere il *Modulo Microprocessore*. Insieme al *Modulo Pulsantiera*, esso si occupa di gestire e controllare tutta l'elettronica descritta finora.

Guardando lo schema elettrico corrispondente possiamo immaginarlo composto dalle seguenti sezioni: gestore ventola, sonde di temperatura, gestore sensori di corrente per i rami positivo e negativo, bus verso l'LCD e verso il modulo pulsantiera. Riguardo al display, facciamo

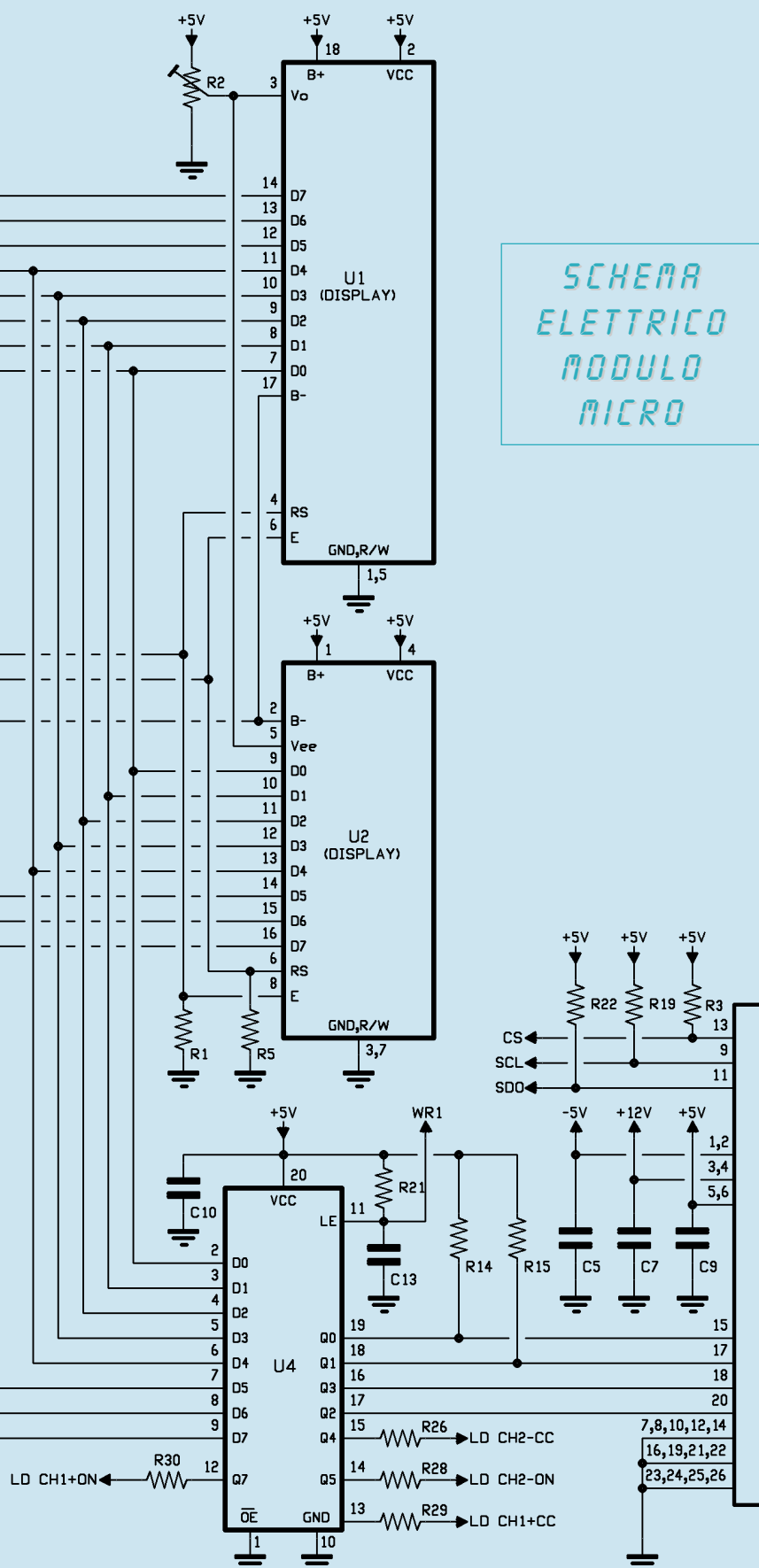
notare che è stata prevista la possibilità di connettere due tipologie di LCD alfanumerici da 2 righe per 16 caratteri.

Cominciamo col descrivere la gestione delle sonde di temperatura, la cui funzione è accertare che il sistema di raffreddamento dei sei darlington finali funzioni correttamente, ovvero che la temperatura sul dissipatore non raggiunga valori che possano compromettere le giunzioni dei componenti. Va infatti considerato che gli elementi di potenza scaldano parecchio, soprattutto quando devono erogare parecchia corrente a basse tensioni di uscita, allorché è massima la loro caduta di tensione collettore/emettitore.

Le sonde previste sono due termistori NTC (Negative Temperature Coefficient, ossia resistenze a coefficiente di temperatura negativo) una per i finali del canale positivo e una per quelli del canale negativo; entrambe sono del tipo Siemens S861/10K. Questo modello a +25°C presenta una resistenza di 10 kohm e può lavorare a temperature comprese nel campo -55 / +155 °C. Trattandosi di una NTC, nella sua curva caratteristica notiamo che più la temperatura scende, più la resistenza aumenta e viceversa; a +155°C la resistenza è dell'ordine dei 2,2 kohm, che diventano 100 a -55°C. Il micro è in grado di rilevare la condizione di sonda scollegata ➤

 IDRISS s.r.l. SISTEMI INFORMATICI IMPIANTI TECNOLOGICI Via Giuseppe Borghi, 21 90144 Palermo Tei/Fax : 0917302587 Mail/MSN: idriss@idriss.it	Progettazione, consulenza, realizzazione, installazione ed assistenza di Sistemi e Prodotti Informatici, Impianti Elettrici e Tecnologici, Impianti di Videosorveglianza Locale e Remota, Sistemi Telefonici Centralizzati, Impianti Antifurto, Sistemi di Videoproiezione Multiscreen, Reti Locali Wired e Wireless, Realizzazione di Aule Multimediali, Sistema di Diffusione P.A., Impianti di Illuminazione Statica e Dinamica. Interventi On-Site. Distributore dei prodotti 
--	--





oppure sonda in corto, condizioni che corrispondono a un valore di resistenza, rispettivamente, infinito e nullo. Il collegamento al PIC è molto semplice: è sufficiente fare un partitore di tensione tra la sonda NTC e un valore di resistenza fisso, ad esempio la sonda NTC R32 e la resistenza R36. Il segnale, prima di essere portato al convertitore A/D del micro, deve essere filtrato dalla cella passa basso realizzata con R8 e C12; R8 serve anche a limitare la corrente entrante nell'ADC del PIC. La tensione presente sul convertitore diminuirà al diminuire della temperatura e crescerà all'aumentare della stessa.

Siccome nelle NTC la variazione di resistenza nei riguardi della temperatura non è lineare, il microcontrollore deve usare una tabella di compensazione per correggere i valori di temperatura letti; la tabella è memorizzata nella sua flash. Durante il campionamento dei valori di temperatura, il PIC ha a disposizione due buffer di memoria indipendenti dove memorizzare in modo ciclico 16 valori di temperatura a 16 bit; tutte le volte che un buffer si riempie, il software ricomincia a scrivere dall'inizio, cancellando i valori esistenti e sostituendoli con quelli nuovi. Una routine indipendente provvede ad eseguire una media pesata tra i 16 valori acquisiti e applica una formula di riscalfatura del valore acquisito, va poi a leggere nella tabella presente in flash il valore di correzione da applicare al valore di temperatura appena calcolato. Affinché l'algoritmo funzioni correttamente è necessario impostare i valori di massimo e minimo; per fare ciò è stata realizzata un'apposita routine da richiamare con una combinazione di tasti allo start-up del micro. La procedura verrà descritta in seguito.

I valori di temperatura letti andranno a determinare la velocità di rota- ➤

Per meglio comprendere il funzionamento del firmware implementato nel microcontrollore, descriviamo una breve porzione di codice inerente l'inizializzazione e la gestione dell'interfaccia seriale SPI. Per inizializzare correttamente la periferica è necessario impostare in modo corretto una serie di bit presenti nei due registri: SSPSTAT ed SSPCON. In particolare SSPCON è il registro di controllo delle periferiche, mentre SSPSTAT è il registro di stato della stessa, i sei bit meno significativi di questo registro sono accessibili solo in lettura. I due bit più significativi sono invece accessibili sia in lettura che scrittura. Il codice seguente mostra come impostare il registro SSPCON per utilizzare la periferica SPI:

```
movlw SSPCON_MASK
movwf SSPCON
movlw SSPSTAT_MASK
call Set_Bank1_RAM
movwf SSPSTAT
call Set_Bank0_RAM
```

Le prime due istruzioni servono per configurare il registro SSPCON, per fare ciò si utilizza una costante la quale viene prima caricata nel registro "W" e poi trasferita nel registro destinazione. "SSPCON_MASK" è una costante (byte) che contiene le impostazioni dei vari bit del registro. Successivamente si devono configurare i due bit più significativi del registro SSPSTAT. La procedura è la stessa del precedente registro, stando però attenti a impostare il corretto banco di RAM prima di passare il valore memorizzato in "W". Il registro SSPSTAT si trova nel banco 1 di RAM. La costante SSPCON_MASK contiene la seguente stringa binaria: quattro bit meno significativi (a partire da destra) impostano la modalità di funzionamento della periferica. Il PIC deve lavorare in modalità MASTER (Il DAC sarà lo SLAVE) con velocità di trasmissione pari a $F_{osc}/16$. Nel nostro caso abbiamo una velocità di trasmissione pari a 312 kHz. Il quinto bit (CKP) serve per identificare lo stato di IDLE della periferica. Nel nostro caso la periferica è nello stato di IDLE quando il clock si trova a livello logico basso. Il sesto bit (SSPEN) abilita la periferica SPI. Quindi configura i pin RC3, RC4, RC5 e RA5 come pin della porta seriale. Il settimo bit (SSPOV) abilita/disabilita il bit che indica se la periferica è andata in overflow. Nel nostro caso il bit di overflow è disabilitato. L'ottavo bit (WCOL) abilita/disabilita l'intercettazione delle collisioni in scrittura. Nel nostro caso le intercettazioni sono disabilitate.

Per quanto riguarda il registro SSPSTAT, i bit di interesse da configurare sono solo i due bit più significativi. Il dato da inviare alla periferica viene trasmesso un bit alla volta e per fare ciò si usa uno shift register che ad ogni colpo di clock invia in uscita sulla seriale un bit del dato. Il settimo bit (CKE) serve per impostare se il bit viene trasferito sul fronte di salita/discesa del

clock. Nel nostro caso i bit del dato sono trasmessi sul fronte di salita del clock. L'ottavo bit (SMP) serve per decidere se il dato presente sul pin SDI deve essere campionato alla fine di ogni bit trasmesso oppure a metà del bit che si sta trasmettendo. Nel nostro caso si è deciso di impostare questo bit a 1 quindi di campionare alla fine della trasmissione di ogni singolo bit.

Il bit meno significativo del registro SSPSTAT (BF) verrà usato per testare la completa trasmissione degli otto bit del dato. Il valore logico "1" assicura la completa trasmissione. Infatti caricando semplicemente il registro SSPBUF si dà inizio alla trasmissione, durante la quale il bit BF va a livello logico "0". Quindi

```
0 0 1 0 0001 LSB MSB
1 1 000000 MSB LSB
```

per capire che la trasmissione è stata portata a termine basta testare questo bit e rimanere in attesa che passi da livello logico "0" a "1". Di seguito viene riportato il codice per inviare i 2 byte a uno dei due DAC presenti nell'integrato AD7395. La routine "Vout_CH1_SPI" viene chiamata dal "main" e a sua volta chiama due subroutine per gestire la trasmissione del dato sulla periferica SPI (Subroutine "Spi_CH1" e Subroutine "Output"):

```
Vout_CH1_SPI
call Spi_CH1 ; Routine per inviare i 2 byte del dato al DAC
bcf LDA ; Le due istruzioni seguenti abilitano il latch A
call Output ; del DAC ad acquisire il dato inviato
bsf SDA
call Output
return
Spi_CH1
bcf CS ; Abilito la periferica SPI del DAC
call Set_Bank1_RAM ; Carico nel registro di trasmissione SSPBUF
movf V_CH1_MSB,W ; il byte più significativo del dato da inviare
call Set_Bank0_RAM ; al DAC
movwf SSPBUF
call Set_Bank1_RAM
btfss SSPSTAT,0 ; Attendo che la trasmissione del byte più
goto $-1 ; significativo sia conclusa
movf V_CH1_LSB,W ; Carico nel registro di trasmissione SSPBUF
call Set_Bank0_RAM ; il byte meno significativo del dato da inviare
movwf SSPBUF ; al DAC
call Set_Bank1_RAM
btfss SSPSTAT,0 ; Attendo che la trasmissione del byte meno
goto $-1 ; significativo sia conclusa
call Set_Bank0_RAM
bsf CS ; Disabilito la periferica SPI del DAC
return
```

Come si può osservare la routine di trasmissione "Spi_CH1" è molto semplice. Per prima cosa si deve abilitare la periferica, cioè il DAC, e per fare ciò si deve portare a livello logico basso il Chip Select della periferica (CS). Dopo di che è sufficiente caricare il byte più significativo da trasmettere nel buffer di trasmissione SSPBUF e rimanere

zione della ventola collegata all'uscita PWM del PIC: per l'esattezza, più sarà alta la temperatura, più velocemente verrà fatta girare la ventola e viceversa. Per agire sulla

velocità di rotazione il microcontrollore provvede ad alimentare la ventola mediante un convertitore Buck di tipo step-down che produce una tensione di uscita (Vout)

minore di quella di ingresso (Vin) ed è formato dai transistor Q3 e Q1, oltre che da un filtro LC passa basso costituito dall'induttanza L1 e dal condensatore C17. Più preci-

in loop in attesa che la trasmissione sia conclusa. Come detto sopra è sufficiente testare il bit "0" del registro SSPSTAT. Appena conclusa la trasmissione del byte più significativo si può passare al byte meno significativo. Conclusa anche la trasmissione del byte meno significativo si riporta a livello logico alto il Chip Select. Conclusa la trasmissione dei due byte è necessario abilitare uno dei due latch interni del DAC in modo che il dato sia trasferito o sull'uscita uno o sull'uscita due del DAC. I due bit di abilitazione (LDA e LDB) non sono collegati direttamente al PIC ma sono connessi all'uscita del latch U4 (Vedi schema elettrico microcontrollore). Quindi per pilotare questi due bit è necessario utilizzare una variabile di stato in cui si memorizzano i valori dei vari bit di uscita; questa variabile viene poi passata al Latch il quale trasferirà il valore della variabile sulle uscite dello stesso. La routine seguente trasferisce il valore della variabile (Ostato1) al latch U4.

```
Output
bcf INTCON,GIE ; Disabilito Interrupt Globali
bsf RD1
nop
bsf RD2
movf Ostato1,W
movwf PORTD
nop
bsf WR1 ; Scrivo lo STATO del LATCH U1 sulla LPF5_B.
nop
bcf WR1
nop
bsf INTCON,GIE ; Abilito Interrupt Globali
return
```

Il significato della variabile Ostato1 è il seguente:

Bit "0": Bit di controllo latch A interno al DAC (LDA).

Bit "1": Bit di controllo latch B interno al DAC (LDB).

Bit "2": Bit di controllo relè 1.

Bit "3": Bit di controllo relè 2.

Bit "4": Bit di controllo LED corto circuito canale negativo.

Bit "5": Bit di controllo ON/OFF canale negativo.

Bit "6": Bit di controllo LED corto circuito canale positivo.

Bit "7": Bit di controllo ON/OFF canale positivo.

Quindi per semplificare le operazioni di set/reset e rendere il codice più leggibile si usa dare un nome a ogni bit della variabile Ostato1.

```
#define LDA Ostato1,0
#define LDB Ostato1,1
#define Rel1 Ostato1,2
#define Rel2 Ostato1,3
#define LED_CC_CH2 Ostato1,4
#define LED_ON_CH2 Ostato1,5
#define LED_CC_CH1 Ostato1,6
#define LED_ON_CH1 Ostato1,7
```

In questo modo per settare un bit basterà scrivere:

```
bsf LDA
```

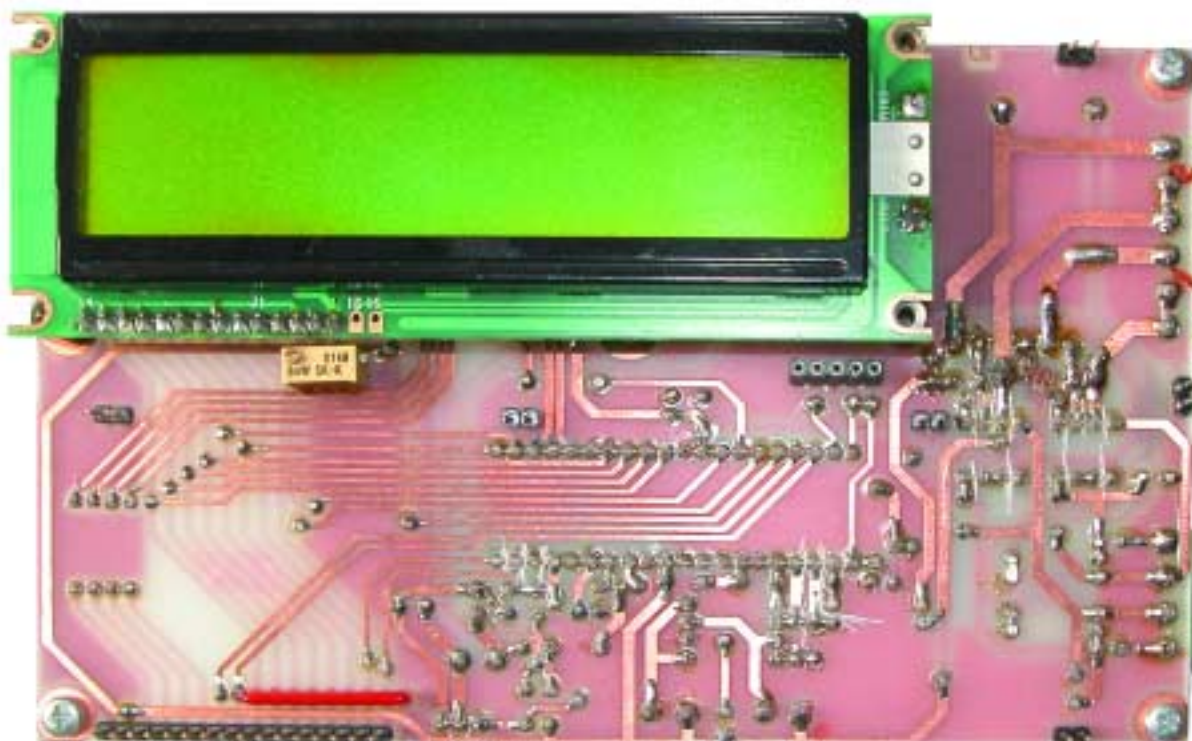
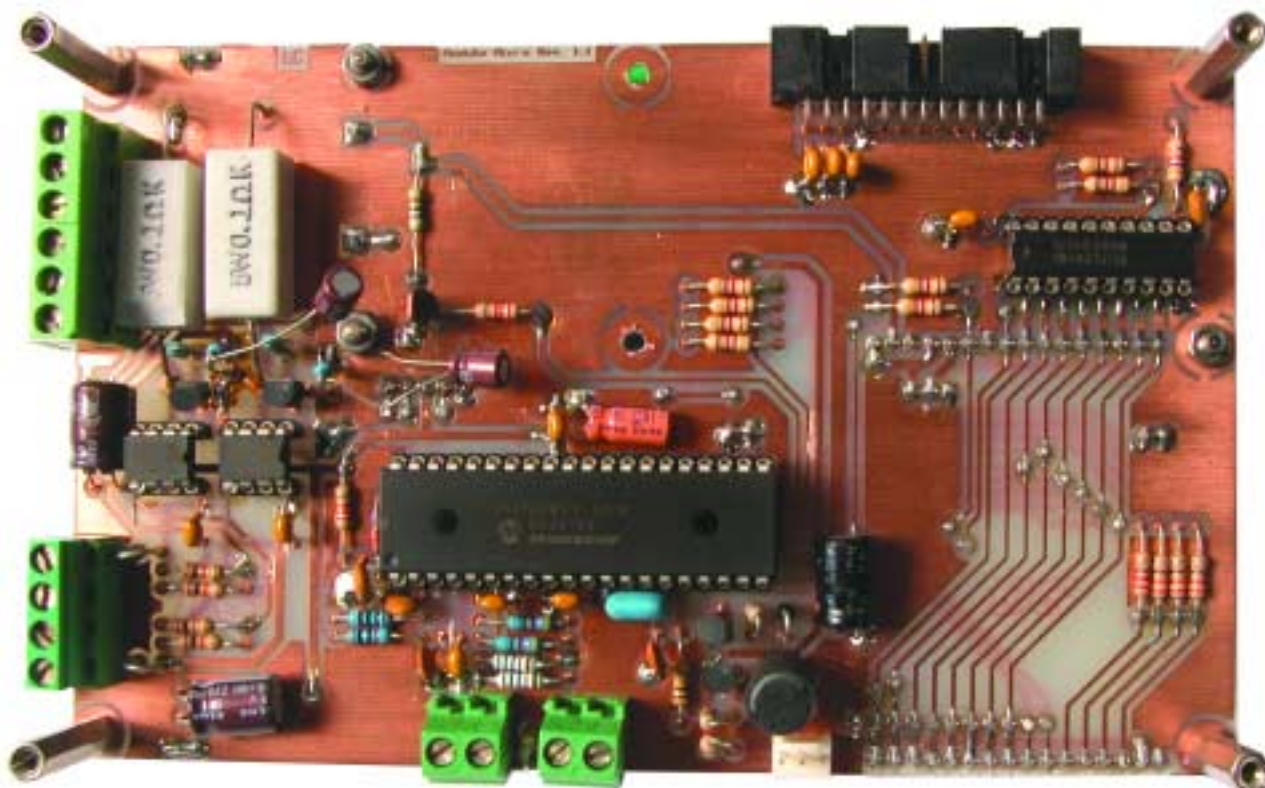
e per resettare:

```
bcf LDA
```

samente, il PIC genera una componente rettangolare in PWM con la quale fa commutare i due transistor; dal collettore del Q1 esce la tensione rettangolare che il filtro seguen-

te livella ottenendo una componente continua e priva di fluttuazioni. Perché la cosa funzioni, la frequenza di taglio della cella L/C deve essere minore di quella dell'onda PWM generata dal micro e riproposta dal Q1. Del regolatore, notate il diodo D2: serve per fare circolare la corrente che l'induttanza, avendola immagazzinata nei periodi di conduzione del Q1, tende a restituire quando il transistor stesso va in interdizione. La regolazione della velocità della ventola non è continua, nel senso che il microcontrollore non agisce analogicamente; prevede invece tre velocità di rotazione, che imposta a seconda della fascia di valori in cui si trova la temperatura attualmente rilevata. Le fasce sono tre: $T < +25^{\circ}\text{C}$; temperatura compresa tra $+25^{\circ}\text{C}$ e $+60^{\circ}\text{C}$, e, infine, temperatura maggiore di $+60^{\circ}\text{C}$. Alle prime due fasce corrispondono velocità che sono, rispettivamente, il 30 ed il 50 % della massima, che poi è quella implementata quando la temperatura dei sensori NTC supera i 60°C . Il microcontrollore non si limita a impartire il comando per ottenere una certa velocità, ma va a verificare che la ventola giri effettivamente come impostato; allo scopo legge il segnale tachimetrico che, tramite l'apposito filo della ventola, viene riportato al piedino RC0 (15). La regolazione della ventola non è casuale, nel senso che il segnale di controllo PWM viene generato al fine di raggiungere una precisa velocità di rotazione: 2.900 giri/min. È per questo che il software del microcontrollore va a leggere gli impulsi provenienti dalla ventola e in base alla loro frequenza ricava l'informazione sulla velocità angolare di rotazione; se risulta corretta mantiene inalterata l'onda PWM, altrimenti ne ritocca il duty-cycle per compensare la differenza. Più esattamente, se la velocità eccede quella voluta viene ristretta la larghezza degli impulsi così da ridurre l'energia ceduta alla ventola; viceversa, se si rileva che la velocità è più bassa si aumenta il duty-cycle al fine di dare al motore della ventola una tensione di valor medio più alto. Il conteggio viene effettuato utilizzando la periferica del PIC Timer1 (a 16 bit) configurata come contatore. A intervalli di tempo di 1 secondo il PIC va a leggere il valore contenuto nel registro TIMER1; il valore letto viene moltiplicato per 30, ottenendo così i giri al minuto della ventola. Il fattore moltiplicativo è 30 perché a ogni giro la ventola genera 2 impulsi, quindi in un minuto si ha esattamente il doppio degli impulsi. Se il PIC non rileva impulsi, ritenendo che la

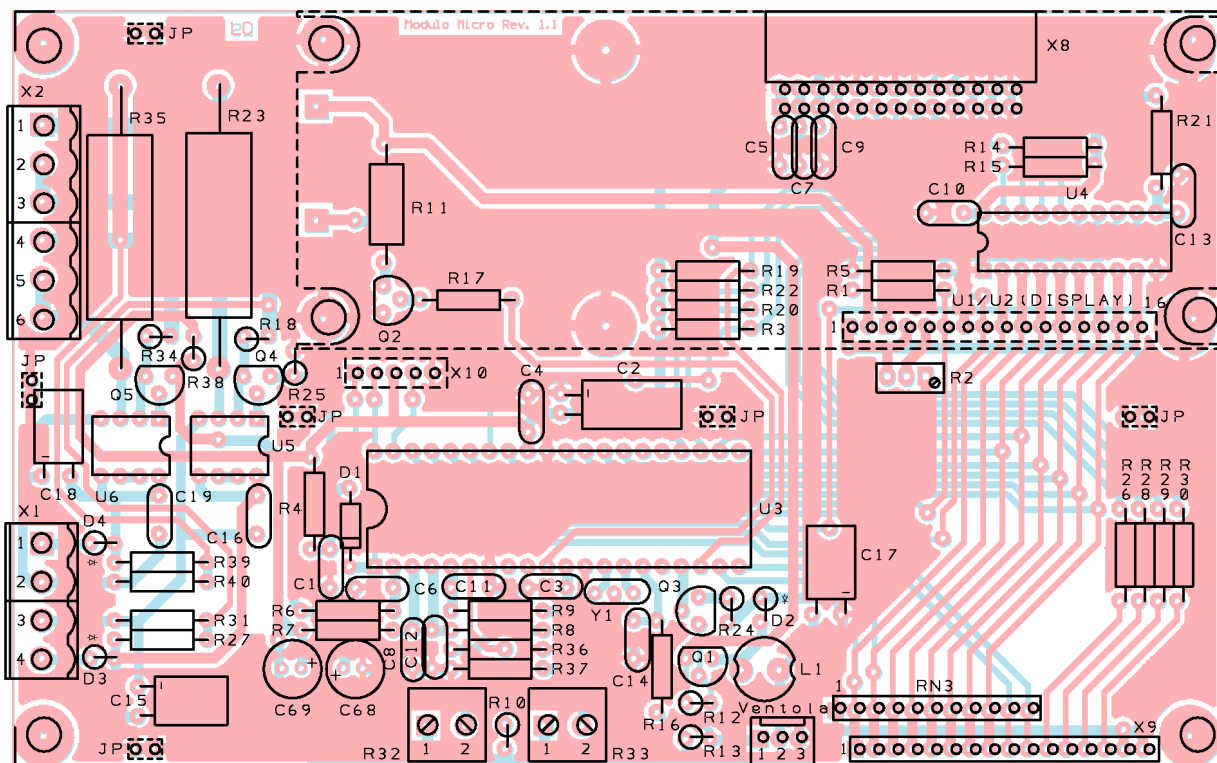
ventola si è fermata (ad esempio perché bloccata dallo sporco o a causa dell'interruzione delle bobine dello statore) avvisa l'utente. Se la ventola o le sonde di temperatura sono guaste il PIC manda in protezione l'alimentatore disabilitando le uscite, le quali vengono riabilite solo dopo aver eliminato le anomalie riscontrate. Nell'unità a microcontrollore si trovano i cir- ➤



cuiti di rilevamento della corrente erogata dall'alimentatore, una per il canale positivo e una per quello

negativo. Descriviamo la sezione del solo canale positivo, visto che il negativo è strutturalmente identico.

Per il canale positivo si usa l'operazionale U5, configurato come generatore di corrente costante. La cor-



ELENCO COMPONENTI:

R1, R3, R5: 4,7 kohm
 R2: Trimmer multigiri 4,7 kohm
 R4, R17: 2,2 kohm
 R6, R7: 10 ohm 1%
 R8÷R10: 4,7 kohm 1%
 R11: 47 ohm 1/2W
 R12: 1,2 kohm
 R13÷R15: 4,7 kohm
 R16: 100 kohm
 R18: 47 ohm 1%
 R19÷R22: 4,7 kohm
 R23: 0,1 ohm 5W
 R24: 680 ohm
 R25: 470 ohm
 R26: 360 ohm
 R27, R31, R38, R40: 10 kohm
 R28÷R30: 360 ohm 1%
 R32, R33: NTC 10 kohm a 25 °C
 R34: 47 ohm 1%
 R35: 0,1 ohm 5W
 R36, R37: 1 kohm 1%
 R39: 470 ohm

RN1: Rete resistiva 4k7 x 8
 C1: 220 pF 25 VL ceramico
 C2: 10 µF 25 VL elettrolitico
 C3÷C5: 100 nF 100 VL ceramico
 C6, C8: 220 pF 25 VL ceramico
 C7: 100 nF 100 VL ceramico
 C9÷C12: 100 nF 100 VL ceramico
 C13, C14: 220 pF 25 VL ceramico
 C15: 220 µF 25 VL elettrolitico
 C16, C19: 100 nF 100 VL ceramico
 C17: 100 µF 25 VL elettrolitico
 C18: 220 µF 25 VL elettrolitico
 C68, C69: 10 µF 25 VL elettrolitico
 D1: 1N4148
 D2: UF4002
 D3, D4: 1N4007
 Q1, Q4: BC557
 Q2, Q3, Q5: BC547
 L1: Induttanza 470 µH
 Y1: Quarzo 20 MHz
 U3: PIC16F877
 U4: 74HC573

U5, U6: LM358

Varie:

- Morsettiera 3 poli (2 pz.)
- Morsettiera 2 poli (4 pz.)
- Strip femmina 2 pin (6 pz.)
- Strip femmina 16 pin (2 pz.)
- Strip maschio 16 pin
- Strip femmina 5 pin
- Strip maschio 3 pin
- Connettore maschio 90° per flat-cable 13+13
- Flat cable 26 pin F/F
- Zoccolo 4+4 (2 pz.)
- Zoccolo 10+10
- Zoccolo 20+20 doppio passo
- Display LCD 20x2
- Ventola 12 V 80 mm con tachimetrico
- Circuito stampato

Se non specificato diversamente, le resistenze sono da 1/4W al 5%.

rente presente in R25 può essere calcolata con la formula seguente:

$$I = V_{\text{shunt}} / R18 = R23 \times I_{\text{load}} / R18.$$

Se, ad esempio, il carico assorbe una corrente di 100 mA, I sarà:

$$I = (0,1 \times 100 \text{ mA}) / 47$$

ossia 212,76 µA. Quindi la tensione presente sull'ingresso RA0 sarà pari a 100 mV; la si ottiene multi-

plicando la corrente I per la resistenza R25. Il convertitore ADC del PIC provvede a ricavare da tale valore di tensione la grandezza corrispondente alla corrente. Continuando con l'esempio, per una tensione di 100 mV il convertitore ADC restituisce un valore decimale pari a 20, che si ricava dalla formula:

$$V_{in}/Res = 100mVdc/5mVdc.$$

Il valore così ottenuto viene moltiplicato per 5 ricavando il valore di corrente assorbito dal carico, cioè 100 mA. Il valore $Res = Risoluzione\ ADC$ viene approssimato a 5mVdc). Per via delle tolleranze dei componenti, la relazione tra corrente assorbita dal carico e tensione presente sull'ingresso dell'ADC non è perfettamente lineare, quindi il software del microcontrollore deve linearizzare i valori letti riferendosi a un'apposita tabella di conversione. Il PIC, dato il valore acquisito, sottrarrà una quantità "x" ottenendo il corretto valore di corrente. Il micro è in grado di leggere e visualizzare correnti fino a 5A. Anche in questo caso ci sono due buffer di memoria indipendenti in cui memorizzare i 16 valori acquisiti e anche in questo caso il PIC farà la media pesata. Il buffer viene riempito in modo ciclico; quando il buffer è pieno si ricomincia a riempirlo dall'inizio sovrascrivendo i dati presenti.

Siccome il convertitore ADC del PIC deve gestire due sonde di temperatura e due di corrente, si è deciso di configurarlo in modo che possa soddisfarle tutte senza degradare troppo l'una o l'altra misura. I due riferimenti V_{ref+} e V_{ref-} sono presi internamente al PIC e in particolare V_{ref+} è uguale a +5V mentre V_{ref-} è uguale a GND. Ne risulta che la risoluzione del convertitore è pari a:

$$Res = 5\ V/2^{10} = 4,88\ mV.$$

Di conseguenza variazioni inferiori non saranno rilevate dallo strumento. Degli otto possibili canali analogici di ingresso ne sono stati usati cinque; gli altri sono inizializzati come porte di I/O digitali. Siamo stati costretti a usare cinque ingressi analogici, anche se in realtà ne bastavano quattro, perché il registro del PIC "ADCON1" non può essere impostato in modo diverso. Il quinto ingresso, cioè RA5, è forzato a GND.

Spendiamo ora due parole sull'alimentazione degli operazionali usati per la lettura della corrente assorbita dal carico: le tensioni devono essere indipendenti e diverse da tutte quelle generate attraverso il trasformatore di potenza, ragion per cui bisogna utilizzare un trasformatore (di bassa potenza) con due secondari da 9 Vac; le tensioni verranno raddrizzate semplicemente con un diodo e un condensatore elettrolitico. In particolare, D3 e C15 oppure D4 e C18. Al pin 8 di entrambi gli operazionali viene portata l'alimentazione positiva, mentre al pin 4 è collegata la massa, punti "A" e "B". In più, come si vede dallo schema, per l'operazionale U5 viene portata, attraverso le resistenze R27 e R31, la tensione stabilizzata positiva dell'alimentatore, mentre le tensioni negative giungono attraverso le resistenze R38 e R40. In questo modo l'alimentazione degli operazionali si sposta seguendo l'andamento delle tensioni di uscita dell'alimentatore. Per gestire le diverse linee di I/O necessarie al funzionamento del sistema, dato che gli 8 bit di bus non sono sufficienti a gestire tutti gli I/O, il PIC fa uso di una soluzione a bus con una serie di latch impiegata come interfaccia; i latch utilizzati sono tre, di cui uno presente sul *Modulo Microprocessore* e gli altri due sulla scheda di interfaccia (*Modulo Pulsantiera*) dove sono presenti i pulsanti di gestione

del sistema. Riassumendo, abbiamo bisogno di:

- un bus a 8 bit; usiamo per intero la PORTD del PIC;

- tre pin di controllo per abilitare i latch (RB3, RB4 e RB5);

- due pin di controllo per gestire uno dei due possibili display LCD.

In totale abbiamo bisogno di 13 pin di controllo per gestire 24 punti di I/O, più il display LCD.

Sulla scheda a microprocessore c'è uno solo dei 3 latch, al quale sono collegati i seguenti segnali di uscita: LDA (seleziona il buffer di memoria canale 1 del DAC), LDB (seleziona il buffer di memoria canale 2 del DAC), Relé1 (pilota il relé per attivare l'uscita del canale positivo dell'alimentatore), Relé2 (pilota il relé per attivare l'uscita del canale negativo dell'alimentatore), led CH2- CC (indicazione corto circuito su canale negativo), led CH2- ON (indicazione uscita negativa attiva), led CH1+ CC (indicazione corto circuito su canale positivo) e, infine led CH1+ ON (indicazione uscita positiva attiva).

Il latch presente sulla scheda a microprocessore viene pilotato dal pin RB3: quando questo assume lo stato logico alto, ciò che è presente sul bus a 8 bit viene riportato sulle uscite del latch (Q0÷Q7).

Naturalmente, il software presente nel PIC deve aver precedentemente preparato i dati da portare sulle uscite prima di attivare il latch. Nel firmware sono stati previsti dei registri di memoria in cui collocare gli stati delle uscite e degli ingressi; il micro lavora su questi registri e non direttamente sul PORTD.

I due latch di ingresso destinati alla lettura di pulsanti vengono controllati dal PIC mediante le proprie linee RB4 e RB5; i segnali di abilitazione dei latch sono attivi a livello basso, quindi portando a zero uno dei due è possibile leggere lo stato dei pulsanti collegati al rispettivo integrato. A riguardo, si noti

che le linee di abilitazione non possono e non devono essere attivate contemporaneamente, altrimenti si determina un conflitto sul bus e può accadere che i latch vengano danneggiati; infatti se l'uscita di uno genera l'1 logico mentre la corrispondente dell'altro impone lo zero, quest'ultima cortocircuita la prima.

Quanto al display LCD, ha bisogno che il microcontrollore gli dedichi due linee di I/O per controllare i pin RS ed E; il primo serve a discriminare se l'informazione presente sul bus è un dato o un'istruzione, mentre il pin E serve per abilitare il display alla ricezione dei dati. Naturalmente l'LCD deve essere abilitato a ricevere i dati solo quando i tre latch sono in alta impedenza, cioè non abilitati. Per regolare il contrasto dell'LCD si usa il trimmer multigiri R2.

Bene, almeno per la teoria possiamo fermarci qui; vediamo adesso le

note costruttive inerenti i moduli qui descritti.

Realizzazione pratica

Tutti i moduli vanno montati su circuiti stampati a doppia faccia dei quali potete scaricare gratuitamente le tracce lato rame dal sito www.elettronica.in.it. Incise e forate le basette, saldate su di esse i componenti in ordine di altezza, ricordando di stagnare da entrambe le facce i componenti che hanno piazzole sui due lati. Per gli integrati prediligete zoccoli con contatto a tulipano, perché consentono, dove richiesto, di stagnarne i pin da entrambe le facce dello stampato, cosa praticamente impossibile nei classici zoccoli con contatto a molla.

Sul Modulo DAC dovete prevedere un connettore maschio da c.s. a 26 poli (due file da 13 pin) a passo 2,54 mm, nel quale andrà poi il flat-cable per l'interconnessione con il

Modulo Microprocessore. Per l'inserimento nella scheda (Modulo Potenza) occorre un connettore a 22 vie single-in-line a passo 2,54 mm, ottenibile con una strip maschio.

Quanto al Modulo Microprocessore, tenete sollevate di un paio di millimetri dalla superficie dello stampato le resistenze R23 ed R35, allo scopo di consentire loro di smaltire il calore che produrranno durante il funzionamento. Per il fissaggio del display LCD occorre prevedere delle colonnine esagonali, mentre le connessioni sono assicurate da uno strip femmina a 16 vie. Le resistenze NTC andranno all'esterno del circuito, collegate mediante due morsettiere bipolari a passo 5 mm.

Giunti a questo punto, mettete da parte i moduli già pronti e attendete la prossima puntata, conclusiva, nella quale descriveremo la pulsantiera, il montaggio e le procedure di taratura dell'alimentatore.

RADIOELETRONICA E INFORMATICA

MOSTRA MERCATO

 **COMPUTER**
 **ELETTRONICA & COMPONENTISTICA**
 **RADIANTISMO**
 **HFET**
 **EDITORIA**

 **TV-SAT**
 **TELEFONIA**
 **RADIO D'EPOCA**

info@natalissimo.com www.natalissimo.com infoline: 338 4188572

IN CONCOMITANZA CON:

 **FOTOAMATORE**
BORSABOCCAMBIO
 **TUTTOCOLLEZIONISMO**
 **MILI FERRARA**

1° DECENNALE

FERRARA FIERE

11-12 FEBBRAIO 2006

ORARIO DI APERTURA: 9,00 - 18,00



P.F. Promozioni Fieristiche - tel. 051.557730 cell. 338.4188572 fax 051.5280939 www.natalissimo.com info@natalissimo.com



RADIO MATORE HI-FICAR

41^a Fiera nazionale del radioamatore, elettronica, hi-fi car, informatica.

29-30 Aprile
1 Maggio 2006



Pordenone Fiere

www.fierapordenone.it

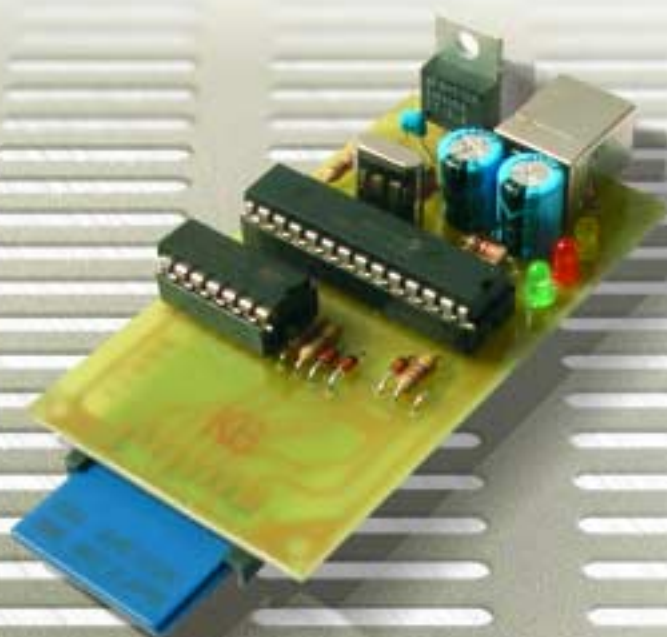
FRIULCASSA
CASA DI RISPARMIO REGIONALE
SPINOTTO



29-30 Aprile : 09.00 > 18.30
1 Maggio : 09.00 > 18.00

Letttore e scrittore SD-Card su USB

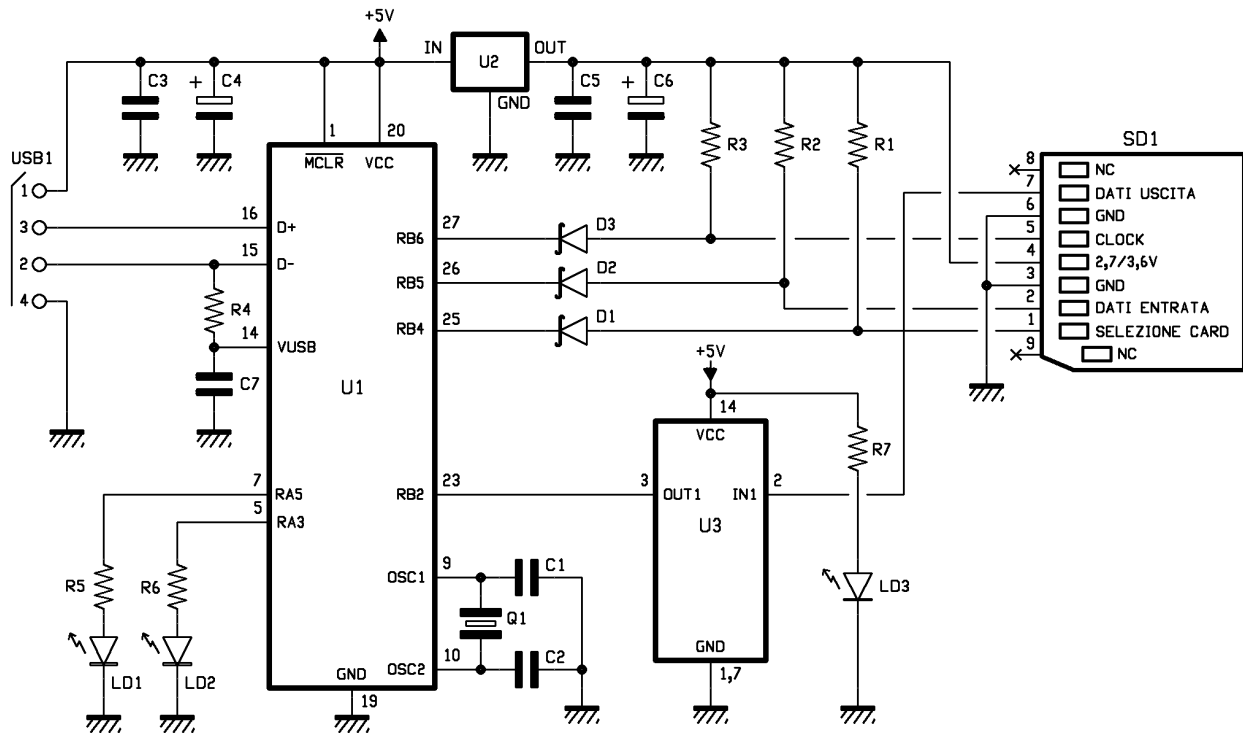
di Carlo Tauraso



Consente di scrivere dati
in memorie
Secure Digital
e di rileggerli
da computer,
mediante la porta USB.
Vediamo
come il dispositivo
è stato sviluppato
sui piani hardware
e software,
grazie all'adozione
di un microcontrollore
dotato di
Universal Serial Bus.

Da qualche tempo si trovano in commercio come dispositivi autonomi o in dotazione alle fotocamere digitali, dei lettori e scrittori per SD-Card ad interfaccia USB o, più raramente, IEEE1394. Per quanto si tratti di apparecchi economici, vi proponiamo di valutare l'autocostruzione, possibilità che vi permette, spendendo all'incirca la stessa quantità di denaro occorrente ad acquistare un lettore/scrittore commerciale, di scoprire come funziona e in che modo è strutturato un dispositivo del genere. In questo articolo vi proponiamo lo schema elettrico e le note progettuali

che ci hanno portato a realizzare il nostro lettore/scrittore; leggetelo, perché è una buona occasione per sperimentare i concetti teorici sulle SD-Card che abbiamo esposto nell'apposito corso pubblicato nei fascicoli n° 94÷98 della rivista. L'obiettivo che ci siamo prefissi è quello di realizzare un piccolo circuito che permetta di scrivere e leggere i singoli blocchi di una SD-Card utilizzando un'interfaccia USB; il risultato finale è stato raggiunto attraverso l'utilizzo di un PIC18F2550, pochi altri componenti ed uno specifico software sviluppato in Delphi. Si tratta senz'altro di un esperimento didattico ➤



co, ma con molte possibili evoluzioni, visto che da un lato integriamo il nostro PIC con una memoria decisamente enorme e dall'altro lo interfacciamo con un sistema di comunicazione rapido ed efficiente quale l'USB. È una maniera utile ed interessante di vedere dal vivo quanto analizzato in teoria nel corso SD-Card. Iniziamo dallo schema circuitale.

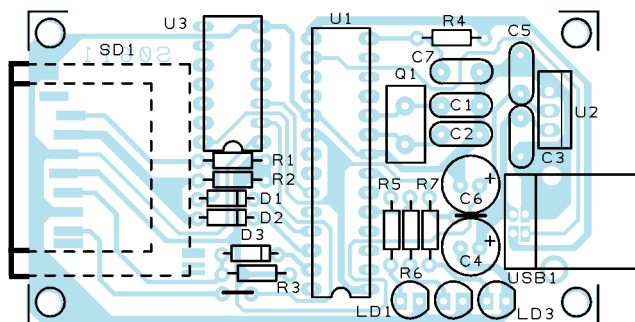
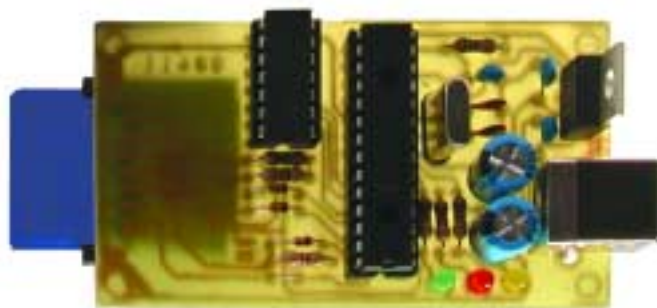
Schema elettrico

Il circuito è decisamente semplice; l'unico problema che abbiamo dovuto affrontare riguarda la realizzazione di un sistema che potesse gestire al meglio la comunicazione tra il PIC, che lavora con segnali da 0 a 5 volt, e la SD-Card, che invece necessita di tensioni comprese tra 0 e 3,6 V. Il tutto è stato risolto attraverso una configurazione circuitale che abbiamo in parte già utilizzato in altri progetti (vedi SMS su PC del n° 92). In pratica, per le linee che vanno dal PIC alla card, si uti-

lizzano un diodo schottky ed una resistenza di pull-up. In questo modo, quando il micro presenta verso la SD l'uno logico, il diodo risulta interdetto e la tensione sul pin della card è quella dovuta al resistore di pull-up, ossia 3,3 V, anche se il PIC fornisce 5 V. Quando il micro genera il livello logico basso, il diodo va in conduzione portando a circa zero volt anche il pin della card. Per quanto riguarda la connessione in direzione inversa, cioè tra card e PIC, la cosa è leggermente differente: per rendere la traduzione dei livelli logici semplice ma allo stesso tempo efficiente e precisa, abbiamo utilizzato un buffer/line driver in tecnologia HCT. Si tratta di un integrato molto economico che permette di risolvere il problema in maniera elegante; in questo caso usiamo la versione più diffusa: la 74HCT125. Per abilitarne le uscite vengono utilizzati quattro pin denominati OE1÷OE4 (Output Enable):

il segnale in input viene presentato sull'output quando la linea OE è allo stato 0. Dato che vogliamo rendere il passaggio input/output più rapido possibile, abbiamo collegato il pin OE direttamente a massa, quindi ogni line-driver è sempre attivo. Le linee di input sono pienamente compatibili con i segnali provenienti dalle SD-Card, perché gli integrati basati su logica ACT/HCT accettano in ingresso livelli TTL e presentano in uscita livelli che, a seconda della tensione fornita al pin di alimentazione, possono essere TTL o del tipo 0/3 V. In particolare, quando sono alimentati a 5 volt "vedono" in ingresso un livello a 3 V come un normale TTL a 5 volt e forniscono in uscita 5 V, tensione che va benissimo per comandare direttamente le linee di input del PIC18F2550.

La tensione di alimentazione per il PIC viene presa direttamente dall'USB, interfaccia che, secondo lo standard cui risponde, può erogare



ELENCO COMPONENTI:

R1÷R3: 4,7 kohm
 R4: 1,5 kohm
 R5÷R7: 1 kohm
 C1, C2: 15 pF ceramico
 C3, C5: 100 nF multistrato
 C4, C6: 220 µF 25 VL elettrolitico
 C7: 220 nF multistrato
 U1: PIC18F2550 (MF611)
 U2: LM1086-3.3
 U3: 74HCT125
 D1÷D3: BAT85
 Q1: quarzo 20 MHz
 LD1: led 3 mm verde
 LD2: led 3 mm rosso
 LD3: led 3 mm giallo
 SD1: connettore SD-CARD

Varie:

- Connettore USB-B
- Zoccolo 14+14
- Zoccolo 7+7
- Circuito stampato codice S611

re 5 V e una corrente di 500 mA. Dai 5 volt, un regolatore LM1086-CT-3.3, molto simile agli ormai conosciutissimi 78Lxx, ricava un potenziale ben stabilizzato di 3,3 V, più che sufficiente per far funzionare correttamente il controller integrato nella card.

Per evidenziare le operazioni svolte dal micro abbiamo dotato la scheda di tre led di colore differente: quello giallo indica che il circuito è alimentato; il rosso segnala che è in corso un'operazione di lettura o scrittura, infine, quello verde indica che la scheda è pronta ad elaborare un comando inviato dall'host. Nel caso entrambi i led rosso e verde risultino accesi, vuol dire che si è verificata

una condizione d'errore che ha bloccato un trasferimento di dati da o verso la SD, pertanto si prevede il reset del sistema.

Vi sarete accorti che nello schema non è presente alcuna memoria temporanea FRAM o EEPROM; ciò perché sfruttiamo la velocità del bus USB per inviare direttamente all'host i valori letti sulla card; analogamente, durante la scrittura riusciamo a far sì che l'host invii

direttamente alla SD i dati che intende scrivere. La versatilità del bus USB ci permette quindi di evitare di appesantire il nostro circuito con un ulteriore integrato, rendendolo ancora più semplice da realizzare. D'altro canto, per rendere il sistema abbastanza efficiente avremmo dovuto utilizzare un buffer veloce, visto che su una EEPROM tradizionale ogni byte necessita di una pausa di stabilizza- ➤

LISTATO 1

```

; *****
; TABELLA PARAMETRI GENERALI
; *****

#define EPO_BUFF_SIZE          8
#define MAX_NUM_INT            1
#define MAX_EP_NUMBER          1
#define NUM_CONFIGURATIONS     1
#define NUM_INTERFACES         1

#define MODE_PP                 _PPBMO
#define UCFG_VAL                _TRINT|MODE_PP ; Low-Speed

#define USB_USE_HID
  
```

LISTATO 2

```
; *****  
; TABELLA PARAMETRI CLASSE HID  
; *****  
  
#define HID_INTF_ID 0x00  
#define HID_UEP UEP1  
#define HID_BD_OUT ep1Bo  
#define HID_INT_OUT_EP_SIZE 8  
#define HID_BD_IN ep1Bi  
#define HID_INT_IN_EP_SIZE 8  
#define HID_NUM_OF_DSC 1
```

zione di 10 ms. La lettura e la scrittura di dati su una SD-Card possono avvenire solo per blocchi da almeno 512 byte, pertanto per ogni operazione di lettura e scrittura il PIC attende l'arrivo di 512 byte, con i quali forma un pacchetto che invia sequenzialmente alla SD, creando di fatto un'interfaccia diretta tra il bus SPI e quello USB. La comunicazione con la SD avviene in modalità SPI, mediante due linee dati seriali (una in ingresso ed una in uscita) sincronizzate da un segnale di clock e una linea di *Chip Select*, utile, nel caso collegassimo più card contemporaneamente, ad indicare con quale dei dispositivi si vuole comunicare.

Il firmware

La maggior complessità di questo

progetto è concentrata nel firmware del PIC e nel front-end costruito in Delphi. Per quanto riguarda il firmware, abbiamo utilizzato il PICBasic al fine di renderlo particolarmente semplice da spiegare. Chiaramente, per tutti coloro che hanno seguito il corso PIC-Basic la spiegazione risulterà ancora più comprensibile, in quanto rappresenta un'immediata sperimentazione pratica di quel che è stato esposto nelle diverse puntate. Iniziamo con un'analisi delle parti salienti del descrittore del dispositivo, ossia del file *SDUSBdsc.asm*. Per coloro che non hanno idea di che cosa sia un descrittore, è utile precisare che il PIC18F2550 verrà riconosciuto dal PC come una periferica HID (Human Interface Device) e per poter comunicare in

maniera corretta attraverso l'interfaccia USB, è necessario far sì che il PC lo identifichi come tale.

Il descrittore è una struttura che fornisce all'host USB tutte le informazioni generali sul dispositivo e sulle sue modalità di funzionamento. Bisogna infatti ricordare che sul bus USB chi comanda è sempre il PC. Inoltre, il

descrittore stabilisce quali e quante sono le porte di comunicazione utilizzabili (i famosi endpoint) pertanto risulta fondamentale se si vuole che il software possa inviare e ricevere i dati dal PIC.

Dal Listato 1 vediamo i punti salienti: stabiliamo la grandezza (in byte) del buffer per l'*Endpoint0* (per i dispositivi low-speed, l'unico valore possibile è 8), la dimensione dell'array che tiene traccia dei settaggi alternativi relativi a ciascuna interfaccia (si ricordi che un dispositivo può avere più interfacce ciascuna con diversi settaggi che l'host può selezionare), il massimo numero di endpoint utilizzati nel progetto (uno soltanto, non si deve tenere in considerazione l'endpoint 0), il numero di configurazioni e interfacce del dispositivo, la modalità di gestione dei buffer associati a ciascun endpoint, la configurazione del registro *UCFG* (USB Configuration Register) e la classe di appartenenza del dispositivo (HID). In particolare, per quanto riguarda il registro *UCFG* abbiamo deciso di utilizzare il transponder interno del chip, ed una configurazione di tipo "Low-Speed". Nella tabella che definisce i parametri della

LISTATO 3

```
; *****  
; TABELLA 3 DESCRITTORE INTERFACE  
; *****  
Interface1  
    retlw (HIDDescriptor1-Interface1)/2 ; bLength  
    retlw DSC_INTF ; bDescriptorType  
    retlw 0x00 ; bInterfaceNumber  
    retlw 0x00 ; bAlternateSetting  
    retlw 0x02 ; bNumEndpoints  
    retlw 0x03 ; bInterfaceClass  
    retlw 0x01 ; bInterfaceSubClass  
    retlw 0x02 ; bInterfaceProtocol  
    retlw 0x05 ; iInterface
```

classe, abbiamo precisato l'utilizzo dell'Endpoint1 IN/OUT, con un buffer ad 8 byte (Listato 2). Nel descrittore *Interface* precisiamo il numero degli endpoint che verranno utilizzati. Si ricordi che in questo conteggio non viene aggiunto l'Endpoint0, che è quello di "servizio". Utilizzeremo, inoltre, due porte di comunicazione ed anticipiamo che una sarà in

uscita ed una in entrata (Listato 3). Nel descrittore degli endpoint (Listato 4) stabiliamo che quello in trasmissione avrà una lunghezza massima di 8 bit (1 byte) mentre quello in ricezione occuperà un massimo di 5 byte. In effetti abbiamo stabilito che il software invierà al PIC dei comandi del tipo *CODICE + 4 PARAMETRI*, ispirandoci alla struttura definita nelle specifiche SD-Card per i CMD17 e CMD24, cioè i comandi di lettura e scrittura che vengono seguiti dall'indirizzo a 32 bit del blocco da leggere o scrivere.

LISTATO 4

```

; *****
; TABELLA 5 DESCRITTORE ENDPOINT
; *****
Endpoint1
    retlw    (Endpoint2-Endpoint1)/2    ; bLength
    retlw    DSC_EP                      ; bDescriptorType
    retlw    0x81                        ; bEndpointAddress
    retlw    0x03                        ; bmAttributes
    retlw    0x01                        ; wMaxPacketSize (low-b)
    retlw    0x00                        ; wMaxPacketSize (high-b)
    retlw    0x0A                        ; bInterval

Endpoint2
    retlw    (EndConfig1-Endpoint2)/2    ; bLength
    retlw    DSC_EP                      ; bDescriptorType
    retlw    0x01                        ; bEndpointAddress
    retlw    0x03                        ; bmAttributes
    retlw    0x05                        ; wMaxPacketSize (low-b)
    retlw    0x00                        ; wMaxPacketSize (high-b)
    retlw    0x0A                        ; bInterval

EndConfig1

```

Dall'altro lato la card risponderà attraverso delle sequenze di singoli byte. L'intervallo di polling è fissato su 10 millisecondi, più che sufficienti per gli scopi che ci siamo prefissati.

Nel descrittore *Report* (Listato 5, a pagina seguente) abbiamo utilizzato una struttura gerarchica che chi ha seguito il corso PIC-USB dovrebbe conoscere molto bene. Sono evidenziate le definizioni dei campi di input e output dimensionati secondo quanto abbiamo detto nel paragrafo precedente. Come si vede, si tratta di strutture a 8 bit per

le quali sono prefissati dei limiti logici ordinari (0-255). Non è il caso di dilungarsi sulla sequenza degli *Usage*, già analizzati a fondo nella seconda puntata del corso. Per quanto riguarda il descrittore *String*, abbiamo utilizzato i valori descritti nella pagina seguente, nel Listato 6. Tali valori vengono riportati su un apposito pannello del software non appena il dispositivo termina il processo di enumerazione. Per venire

incontro ai neofiti di programmazione USB, diciamo che tale processo serve a far sì che host e device "facciano conoscenza" e possano quindi instaurare un canale di comunicazione. Ricordiamo, inoltre, che il nome del file contenente i descrittori deve essere inserito nel file *usbdesc.asm* affinché il compilatore PBP lo includa correttamente nella struttura del *.HEX* che inseriremo, poi, nella memoria del PIC.

In particolare, per la corretta compilazione del progetto software abbiamo utilizzato un file *p18f2550.inc* opportunamente modificato, che prevede una divisione della frequenza di oscillazione in ingresso pari a 5 ed una divisione della frequenza di clock del sistema pari a 4. In questo modo, nel circuito possiamo usare un quarzo a 20 MHz mantenendo (grazie all'attivazione del PLL) un clock interno di 24 MHz.

Per la programmazione del PIC utilizziamo il *Melabs Programmer* della microEngineering Labs, mantenendo le configurazioni visibili nella finestra di dialogo mostrata dalla Fig. 1.

Detto questo, è giunto il momento di passare alla parte di codice vera e propria. Innanzitutto diamo uno sguardo al Listato 7, dal quale ➤



Fig. 1

LISTATO 5

```
ReportDescriptor
retlw 0x06 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x01 ; Usage Page (low-b) ("Vendor Defined Page 1")
retlw 0xFF ; Usage Page (high-b) ("Vendor Defined Page 1")
retlw 0x09 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x01 ; Usage ("Vendor Defined Usage 1")
retlw 0xA1 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x01 ; Collection ("Application")
retlw 0x09 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x02 ; Usage ("Vendor Defined Usage 2")
retlw 0xA1 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x00 ; Collection ("Physical")
retlw 0x06 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x02 ; Usage Page (low-b) ("Vendor Defined Page 2")
retlw 0xFF ; Usage Page (high-b) ("Vendor Defined Page 2")
retlw 0x09 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x03 ; Usage ("Vendor Defined Usage 3")
retlw 0x09 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x04 ; Usage ("Vendor Defined Usage 4")
retlw 0x15 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x00 ; Logical Minimum (0)
retlw 0x26 ; Byte di prefisso (bTag,bType,bSize)
retlw 0xFF ; Logical Maximum (low-b) (255)
retlw 0x00 ; Logical Maximum (high-b)
retlw 0x75 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x08 ; Report Size (8 bits)
retlw 0x95 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x01 ; Report Count (1 campo dati)
retlw 0x81 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x02 ; Input (Data, Var, Abs)
retlw 0x09 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x05 ; Usage ("Vendor Defined Usage 5")
retlw 0x15 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x00 ; Logical Minimum (0)
retlw 0x26 ; Byte di prefisso (bTag,bType,bSize)
retlw 0xFF ; Logical Maximum (low-b) (255)
retlw 0x00 ; Logical Maximum (high-b)
retlw 0x75 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x08 ; Report Size (8 bits)
retlw 0x95 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x05 ; Report Count (5 campi dati)
retlw 0x91 ; Byte di prefisso (bTag,bType,bSize)
retlw 0x02 ; Output (Data, Var, Abs)
retlw 0xC0 ; End Collection ("Physical")
retlw 0xC0 ; End Collection ("Application")
end_ReportDescriptor
```

vediamo un po' di dichiarazioni: per la comunicazione con la Card utilizziamo la modalità SPI, che comprende un sottoinsieme dei comandi del protocollo SD standard ed è anche più semplice da

implementare; utilizza, infatti, solo 3 linee per comunicare, una per il clock (SCK), una per i dati in ingresso (SDI) ed un'altra per i dati in uscita (SDO) oltre, naturalmente, ad un segnale di CS (Chip Select) che ci servirà principalmente per entrare in modalità SPI e per stabilire l'inizio e la fine di una transazione.

LISTATO 6

```
DEVICE NAME: Lettore/Scrittore SDCARD
MANUFACTURER: Microchip
SERIAL NUMBER: TAU333
CONFIGURATION: CFG1
INTERFACE: EP1/INOUT
```

Abbiamo, inoltre, precisato i pin utilizzati per l'accensione e lo spegnimento dei due led di segnalazione. Infine, segue la sequenza di variabili necessarie per l'elaborazione. La coppia IND0, IND1 serve per stabilire l'indirizzo del blocco da leggere o scrivere e compone la sequenza di 32 bit che viene inviata dall'host come lista parametri. Il vettore CMD, costituito da 5 byte, conterrà la sequenza *CODICE + 4 PARAMETRI*. La scelta è dettata da quanto stabilito nel descrittore report.

Come mostra il Listato 8, il codice di inizializzazione è piuttosto semplice.

La PORTA viene messa tutta in uscita visto che ne utilizziamo due linee per comandare i led, mentre per la PORTB viene previsto il pin RB2 come linea di entrata per i dati provenienti dalla card. Viene effettuato lo spegnimento di entrambi i led, quindi

si passa al codice di avvio del dispositivo prima e della Card poi. Inizialmente viene richiamata la procedura di enumerazione del dispositivo e viene effettuata una pausa di stabilizzazione di mezzo secondo. Al termine della procedura, il PIC viene riconosciuto dal sistema operativo del PC ed è pronto a porsi in comunicazione con il software che utilizzeremo per la lettura e la scrittura su Card. Al termine di tale procedura avviamo una fase di inizializzazione della Card,

che ci permetterà prima di passare alla modalità SPI e poi di stabilire il limite dimensionale dei blocchi di lettura e scrittura. Riassumendo, possiamo identificare due fasi fondamentali: RESET CARD (CMD0) e INIZIALIZZAZIONE (CMD1).

Nel Listato 9 vedete come si presenta il relativo codice.

Notate che il reset è preceduto da un'apposita sequenza di cicli di clock a "vuoto" (Dummy Clock) come stabilito nelle specifiche SD-Card. In particolare, il CMD0 viene inviato mantenendo la linea CS a 0: questa procedura permette di entrare nella modalità SPI. Una volta entrati si avvia l'inizializzazione del dispositivo e si attende che la card termini tale fase, verificando

LISTATO 7

```
*****
*Connessione con la SDCard
*****
SCK var PORTB.6 'CLOCK-CARD PIN5 (27 PIC)
SDI var PORTB.5 'DATI-ENTRATA-CARD PIN2 (26 PIC)
SS var PORTB.4 'SELEZIONE-CARD PIN1 (25 PIC)
SDO var PORTB.2 'DATI-USCITA-CARD PIN7 (23 PIC)
*****
* LED
*****
ROSSO var PORTA.3 'LED ROSSO
VERDE var PORTA.5 'LED VERDE
*****
* Applicazione
*****
RISP1 var byte 'RISPOSTA TIPO R1 DA CARD (8bit)
RISP2 var word 'RISPOSTA TIPO R2 DA CARD (16bit)
IND1 var word 'INDIRIZZO SDCARD WORD ALTA BIT 16-31
IND0 var word 'INDIRIZZO SDCARD WORD BASSA BIT 0-15
CAR var byte 'CARATTERE DA SCRIVERE
CONTA VAR BYTE 'contatore byte ricevuti
CMD VAR BYTE[5] 'Sequenza Comando+IND0+IND1
CONTA1 var word 'CONTATORE
CONTA2 var word 'CONTATORE
CTL var byte 'BYTE DI CONTROLLO FRAM
QX var byte 'ANALISI 4 Bit Risposta Dati
```

LISTATO 8

```
ADCON1 = %00001111 'PIN RA Digitali
TRISA = %00000000 'PORTA in uscita
TRISB = %00000100 'RB6=CLOCK RB5=VERSO CARD
'RB4=SELEZ CARD RB2=DA CARD
'Reset led

PORTA = 0
ROSSO = 1
VERDE = 0

USBInit 'Processo di enumerazione alla fine ildispositivo
'entra nello stato Configurato
Pause 500 'Attesa
```

stabilisce il raggiungimento del limite di ripetizione oltre il quale si presume che la card si rifiuta di rispondere o il processo di inizializzazione non è andato a buon fine. Viene saltata la fase di definizione del numero di byte che comporranno il blocco di lettura e scrittura

l'arrivo della risposta che essa invia.

Ricordiamo che in modalità SPI la card è in grado di inviare esclusivamente due tipologie di risposte, una a 8 bit (*RISP1*) ed una a 16 bit (*RISP2*), che, come avete visto negli articoli di approfondimento pubblicati sulle SD-Card, hanno ciascuna una particolare struttura di segnalazione degli errori. Facciamo notare che in queste sequenze è stato previsto un timeout per evitare che in caso di errore il siste-

ma si blocchi. In questo caso, attraverso un contatore (CONTA1) si

tura della card, perché si usa la lunghezza standard pari a 512 byte ➤

LISTATO 9

```
RESET:
SS=1
FOR CONTA1 = 1 TO 10
SHIFTOUT SDI, SCK, MSBFIRST, [0xFF] 'Invio cicli di clock a vuoto
NEXT CONTA1
SS=0
PAUSE 50
*****
* CMD0 mantenendo SS a 0
*****
SHIFTOUT SDI, SCK, MSBFIRST, [$40,$00,$00,$00,$00,$95] 'Invio CMD0
SHIFTIN SDO, SCK, MSBPRES, [RISP1] 'Leggo risposta R1 da Card
CONTA1 = 0
WHILE RISP1 <> 1
SHIFTIN SDO, SCK, MSBPRES, [RISP1] 'Leggo risposta R1 da Card
CONTA1 = CONTA1 + 1
IF CONTA1 >= 255 THEN 'Superato Time-Out esco
GOTO ERRORE
ENDIF
WEND
SS=1
PAUSE 50
SS=0
```

LISTATO 10

```
*****
* Invio ripetutamente CMD1 finchè risposta = 0
*****
CONTA1 = 0
RISP1 = 1
WHILE RISP1 <> 0
SS=1
SHIFTOUT SDI,SCK,MSBFIRST,[ $FF]
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
SS=0
PAUSE 50
SHIFTOUT SDI,SCK,MSBFIRST,[ $41,$00,$00,$00,$00,$FF,$FF] 'Invio CMD1
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
CONTA1 = CONTA1 + 1
IF CONTA1 >= 255 THEN 'Superato Time-Out Esco
GOTO ERRORE
ENDIF
WEND
```

(Listato 10).

Terminata questa prima procedura accendiamo il led verde, segnalando che il dispositivo è inizializzato e pronto a ricevere comandi dal software che gira nel computer. In

dei quali conterrà il codice del comando da eseguire.

Nel Listato 11 si vede chiaramente la sequenza logica utilizzata per stabilire quali istruzioni eseguire a seconda del primo byte ricevuto. In

avvia la lettura (nel protocollo si usa il *CMD17*) mentre il 24 inizia la scrittura (nel protocollo si usa il *CMD24*). Nel caso arrivi un qualsiasi altro comando, lo scartiamo e ritorniamo a leggere una nuova sequenza da 5 byte. Si noti che viene richiamata regolarmente la funzione *USBService* che gestisce il modulo d'interfaccia USB. Il codice risultante, infatti, non è

più basato sulla gestione dei segnali di interrupt, ma esclusivamente su una continua interrogazione dei registri associati alla porta finalizzata a identificare le varie operazioni che devono essere svolte.

Per quanto riguarda la procedura di lettura, abbiamo stabilito di ricevere l'indirizzo del blocco da leggere negli ultimi quattro byte della sequenza di cinque. A questo punto avviamo il comando di lettura sulla card, trasferendo i 512 byte sul bus USB attraverso l'istruzione *USBOUT*. Le istruzioni risultanti sono visibili nel Listato 12.

LISTATO 11

```
AVVIOP:
ROSSO = 0          'Spegnimento LED Rosso
VERDE = 1          'Accensione LED Verde

RICEVI:
USBService
CONTA = 5
USBIN 1,CMD,CONTA,RICEVI          'Ricevi sequenza comando
USBService
IF CMD[0] = 17 THEN               'CMD17 = Lettura Settore
GOTO LEGGI
ENDIF
IF CMD[0] = 24 THEN               'CMD24 = Scrittura Settore
GOTO SCRIVI
ENDIF
GOTO AVVIOP
```

particolare, entriamo in un loop di attesa sulla porta USB, attraverso la quale ci aspettiamo di ricevere una sequenza di cinque byte, il primo

particolare, abbiamo intenzionalmente mantenuto la codifica utilizzata nella descrizione del protocollo SPI per SD-Card, quindi il 17

Accendiamo il led rosso per segnalare che l'elaborazione del comando è in corso. In pratica viene inviato il *CMD17* alla card passando

TELSTAR di Argieri M.T.
Via V.Gioberti, 37/B - 10128 TORINO (TO)
Tel./Fax 011/54.55.87
telstar.rtv@inwind.it

COMPONENTI ELETTRONICI per HOBBYISTI -
KITS - STRUMENTAZIONE - BATTERIE -
SALDATORI - TELECAMERE - MONITOR -
EFFETTI LUCE ... e altro ancora

RIVENDITORE AUTORIZZATO KITS e PRODOTTI:

veleman-kit

FUTURA ELETTRONICA

ELETRONICA

G.P.E.

ELSE KIT

LISTATO 12

LEGGI:

```
ROSSO = 1
VERDE = 0
IND0.BYTE1 = CMD[1]
IND0.BYTE0 = CMD[2]
IND1.BYTE1 = CMD[3]
IND1.BYTE0 = CMD[4]

'Accensione LED Rosso
'Spegnimento LED Verde
'Valorizzazione Indirizzo da leggere

'* LETTURA 512 BYTE DA SDCARD
SS=1
SHIFTOUT SDI,SCK,MSBFIRST,[ $FF]
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
SS=0
SHIFTOUT SDI,SCK,MSBFIRST,[ $51,IND1.BYTE1,IND1.BYTE0,IND0.BYTE1,IND0.BYTE0,$FF] 'Invio CMD17
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
CONTA1 = 0
WHILE RISP1<> 0
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
CONTA1 = CONTA1 + 1
IF CONTA1 >= 255 THEN 'Superato Time-Out Esco
GOTO ERRORE
ENDIF
WEND

'Ricezione START BLOCK
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
CONTA1 = 0
WHILE RISP1<> $FE
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
CONTA1 = CONTA1 + 1
IF CONTA1 >= 255 THEN 'Superato Time-Out Esco
GOTO ERRORE
ENDIF
WEND

'Ricezione BLOCCO DATI
FOR CONTA2= 0 TO 511
SHIFTIN SDO,SCK,MSBPRE,[CAR]
IN VIA: USBService
USBOUT 1,CAR,1,IN VIA 'Invio tramite USB
NEXT CONTA2

'Ricezione CRC
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
SHIFTIN SDO,SCK,MSBPRE,[RISP1]

'Appena la card ha finito di leggere verifico lo stato della card
'Invio il CMD13
SS=1
SHIFTOUT SDI,SCK,MSBFIRST,[ $FF]
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
SS=0
SHIFTOUT SDI,SCK,MSBFIRST,[ $4D,$00,$00,$00,$00,$FF]
'Ricevo lo status a 16 bit risposta formato 2
SHIFTIN SDO,SCK,MSBPRE,[RISP2\16]
CONTA2=0
WHILE RISP2.BYTE0 <> 0
SHIFTIN SDO,SCK,MSBPRE,[RISP2\16]
CONTA2 = CONTA2 + 1
IF CONTA2 >= 255 THEN
GOTO ERRORE2
ENDIF
WEND
GOTO AVVIOP 'Ritorna al ciclo di attesa comando
```

Inizializzazione dell'indirizzo del blocco da leggere su card attraverso gli ultimi 4 byte della sequenza di 5.

Indirizzo del blocco da leggere passato alla card come parametro di CMD17.

Ricezione dello Start-Block 11111110.

Ricezione di 1 byte e scrittura su FRAM

Ricezione dello status della card attraverso la risposta a 16 bit

come parametri gli ultimi 4 byte ricevuti dal PC. Nel momento in cui il comando viene accettato, la card invia una sequenza di bit particolare chiamata Start-Block, dopodiché inizia la trasmissione in sequenza dei 512 byte; ad ogni byte corri-

sponde immediatamente un invio. Al termine, attraverso il CMD13 viene letto il registro di Status della card, verificando se l'operazione è andata a buon fine oppure si è verificato qualche errore. Successivamente viene fatto un

jump alla label *AVVIOP* dove è localizzato il codice che si occuperà di attendere un nuovo comando dall'host. Il led rosso viene spento, quello verde riacceso per segnalare all'utente la possibilità di inviare una nuova richiesta di lettura o ➤

LISTATO 13

SCRIVI:

```
ROSSO = 1      'Accensione LED Rosso
VERDE = 0      'Spegnimento LED Verde
IND0.BYTE1 = CMD[1] 'Valorizzazione Indirizzo da scrivere
IND0.BYTE0 = CMD[2]
IND1.BYTE1 = CMD[3]
IND1.BYTE0 = CMD[4]
```

```
'-----
'* SCRITTURA 512 BYTE SU SDCARD
'-----
```

```
USBService
SS=1
SHIFTOUT SDI,SCK,MSBFIRST,[$FF]
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
'Invio CMD24 scrittura blocco su SDCard
SS=0
SHIFTOUT SDI,SCK,MSBFIRST,[$58,IND1.BYTE1,IND1.BYTE0,IND0.BYTE1,IND0.BYTE0,$FF]
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
CONTA2 = 0
WHILE RISP1 <> 0
  SHIFTIN SDO,SCK,MSBPRE,[RISP1]
  CONTA2 = CONTA2 + 1
  IF CONTA2 > 10000 THEN
    GOTO ERRORE
  ENDIF
WEND
'Invio Start Block %11111110=$FE
SHIFTOUT SDI,SCK,MSBFIRST,[$FE]
```

Passaggio dell'indirizzo del blocco da scrivere

Verifica se la card è pronta a ricevere dati

scrittura. Veniamo quindi all'etichetta *SCRIVI*, che contiene il codice della scrittura di un blocco da 512 byte su Card (Listato 13).

Accendiamo il solito led rosso e valorizziamo opportunamente *IND0* e *IND1* con l'indirizzo del settore da scrivere. Attraverso il *CMD24* inizializziamo la transazione di scrittura ed inviamo lo *Start Block* (Listato 14). L'host ci invia 104 pacchetti da 5 byte ciascuno: il primo è composto dall'identificativo del comando da eseguire seguito dall'indirizzo del relativo settore. I restanti 103 sono costituiti dai valori da scrivere sulla card. In particolare, l'ultimo pacchetto ha 3 byte di chiusura, pertanto come si vede nel listato scarichiamo soltanto i primi due nella card.

Concludiamo, quindi, l'operazione di scrittura. (Listato 15). Segnaliamo l'operazione di AND logico per l'estrazione del cosiddetto "Data Response Token" che

viene inviato per ogni blocco dati scritto. Questa sequenza di bit diventa molto importante nel caso di operazioni più onerose, come la scrittura di più blocchi contigui, che, in questo progetto, però non utilizziamo. Se, infatti, si verifica un errore, il PIC deve bloccare la transazione attraverso un *CMD12*.

comando attraverso l'USB.

Il software per PC

Bene, dopo aver esaminato esaurientemente lo schema elettrico del lettore/scrittore di SD-Card ed il firmware caricato nel microcontrollore PIC18F2550 deputato a gestirlo, consideriamo, ora, il software che

LISTATO 14

RX5: FOR CONTA1 = 1 TO 102

```
USBService
CONTA = 5
USBIN 1,CMD,CONTA,RX5
USBService
```

'Inizio Scrittura su SDCARD

```
FOR CONTA = 0 TO 4
  CAR = CMD[CONTA]
  SHIFTOUT SDI,SCK,MSBFIRST,[CAR]
NEXT CONTA
NEXT CONTA1
```

RX6: USBService

```
CONTA = 5
USBIN 1,CMD,CONTA,RX6
USBService
CAR = CMD[0]
SHIFTOUT SDI,SCK,MSBFIRST,[CAR]
CAR = CMD[1]
SHIFTOUT SDI,SCK,MSBFIRST,[CAR]
```

Ricezione blocco 5 byte

Scrittura ultimi 5 byte ricevuti

Chiusura del settore

Nel nostro piccolo, invece, non facciamo altro che segnalare l'errore accendendo entrambi i led (etichette *ERRORE*, *ERRORE2*) e resettan-

faremo girare nel nostro PC per comandare la lettura e la scrittura della SD-Card. È stato sviluppato in Delphi ed utilizza principalmen-

te l'oggetto *TJvHidDevice-Controller* che abbiamo presentato anche nel corso PIC-USB: si tratta di un componente nato da un progetto del JEDI (Joint Endeavour of Delphi Innovators) ossia una comunità internazionale composta da migliaia di sviluppatori Delphi che ha come scopo quello di diffondere l'utilizzo di tale ambiente di sviluppo e del suo fratello minore Kylix. L'interfaccia del programma è molto semplice da utilizzare e permette di sperimentare immediatamente le funzionalità che abbiamo introdotto nel firmware.

Il form principale (Fig. 2) presenta tre pannelli: uno contiene i dati relativi al dispositivo USB, un altro permette di scegliere la dimensione della card, ed il terzo raccoglie i pulsanti necessari alle funzioni di lettura e scrittura. Il tutto è corredato da una semplice griglia composta da 512 celle suddivise in 32 righe da 16 elementi. In questo modo è possibile controllare e modificare i singoli valori. Per evitare errori è stata prevista una procedura di con-

LISTATO 15

```
SHIFTOUT SDI,SCK,MSBFIRST,[\$FF,\$FF]
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
QX = %00000000
QX = RISP1 & \$0F
IF QX <> %00000101 THEN
GOTO ERRORE
ENDIF
'Ricevo il bit busy della card mentre scrive
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
CONTA2=0
WHILE RISP1 = 0
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
CONTA2 = CONTA2 + 1
IF CONTA2 >= 255 THEN
GOTO ERRORE
ENDIF
WEND
'Invio il CMD13
SS=1
SHIFTOUT SDI,SCK,MSBFIRST,[\$FF]
SHIFTIN SDO,SCK,MSBPRE,[RISP1]
SS=0
SHIFTOUT SDI,SCK,MSBFIRST,[\$4D,\$00,\$00,\$00,\$00,\$FF]
SHIFTIN SDO,SCK,MSBPRE,[RISP2\16]
CONTA2=0
WHILE RISP2.BYTE0 <> 0
SHIFTIN SDO,SCK,MSBPRE,[RISP2\16]
CONTA2 = CONTA2 + 1
IF CONTA2 >= 255 THEN
GOTO ERRORE2
ENDIF
WEND
GoTo AVVIOP
```

Ultimi 4 bit risposta del comando di scrittura:
0101 DATI ACCETTATI
1011 DATI RIFIUTATI ERRORE CRC
1101 DATI RIFIUTATI ERRORE SCRITTURA

Ricezione status della card

'Continua all'infinito

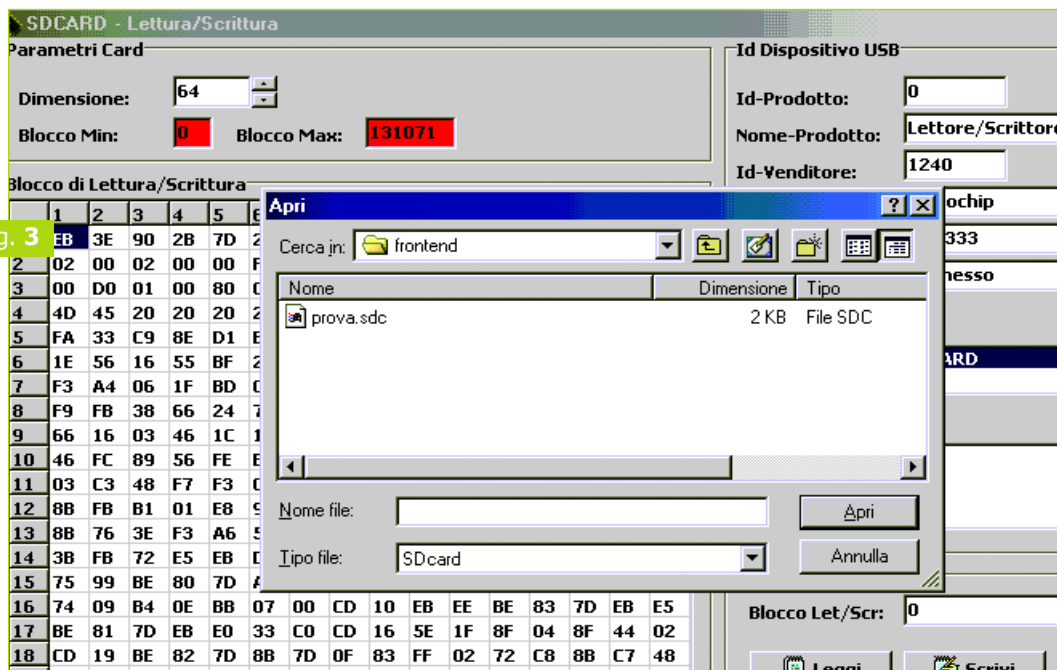
trollo della congruità dei dati inseriti, non appena viene avviata una lettura, scrittura, esportazione o importazione. Nel caso sia stato

inserito un valore non ammissibile viene visualizzato un messaggio che riporta i numeri di riga e di colonna relativi alla cella che non ha superato il controllo.

Se colleghiamo alla porta USB il nostro circuito (con la card già inserita) a seguito del processo di enumerazione, vengono caricati nell'apposito pannello i campi identificativi dello stesso (ricordate i valori inseriti nel descrittore *String?*). Nella lista dei dispositivi compare la voce *Lettore/Scrittore SDCard*. In particolare, viene aggiornato lo status della comunicazione (*Collegato* o *Scollegato*) e se osserviamo la nostra schedina vedremo l'accensione prima del led giallo che segnala l'arrivo della tensione d'alimentazione e dopo qualche frazione di secondo quella del led verde. Ciò significa che la card ha superato la fase di reset, l'entrata in modalità SPI ed il processo di inizializzazione. Notate che qualora si colleghi il circuito senza la card si accenderanno entrambi i led verde ➤

Fig. 2

Fig. 3



e rosso segnalando una condizione d'errore.

Dopo questa prima fase ci si può posizionare sul pannello *Parametri Card* per selezionare attraverso le due freccette la dimensione della card che intendiamo utilizzare. In questo modo viene ricalcolato il limite massimo dei settori accessibili. Il numero si riferisce alla quantità di sezioni da 512 byte utilizzabili per conservare dati.

A questo punto ci si può posizionare sul pannello *Procedure*. Qui inseriamo il numero del settore che vogliamo elaborare; naturalmente la modifica di questo campo è collegata al limite dimensionale della card che abbiamo stabilito prima.

Attraverso i pulsanti del pannello possiamo avviare cinque funzioni fondamentali; analizziamole singolarmente.

Leggi: facendo clic su questo pulsante viene avviata una lettura del settore che abbiamo precisato nel campo (*Blocco Let/Scr*); l'avanzamento della procedura è controllabile attraverso la barra rossa sottostante; via-via che la sequenza di byte arriva al PC i valori vengono inseriti nella griglia e sono direttamente visibili e modificabili.

Scrivi: avvia un'informazione di scrittura del settore che abbiamo precisato nel campo (*Blocco Let/Scr*) scaricando i dati direttamente dalla griglia.

Cancella: svuota la griglia inserendo il valore FFh in tutte le celle.

Importa: carica nella griglia, i dati da un file esterno.

Esporta: scarica i dati dalla griglia ad un file esterno.

Per quanto riguarda le due ultime funzioni si tenga presente che i file utilizzati hanno un'estensione del tipo *.sdc*, tuttavia sono semplicemente dei file di testo. La loro struttura è semplicissima: in pratica

ad ogni riga corrisponde un singolo valore da inserire in griglia. La sequenza viene generata leggendo ciascuna riga da destra a sinistra e dall'alto in basso, come si farebbe su un foglio di carta. Facendo clic sul pulsante *Importa* viene visualizzata una finestra di dialogo come quella illustrata nella Fig. 3.

È quindi possibile selezionare il file e caricarlo facendo clic sul pulsante *Apri*. Viene effettuato un controllo sui dati che vengono inseriti segnalando eventualmente la riga errata. La procedura di esportazione funziona in maniera essenzialmente complementare. Anche qui viene visualizzata una finestra di dialogo per inserire il nome del file che si

Per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono facilmente reperibili in commercio ad eccezione del microcontrollore programmato che costa 18,00 Euro (cod. MF611). Il master del circuito stampato utilizzato, il software in Delphi e il firmware in formato hex possono essere scaricati gratuitamente dal sito www.elettronica.in.it. Il file sorgente del firmware è invece riservato agli abbonati alla rivista.

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

vuole creare. Dopodiché, facendo clic sul pulsante *Salva* la griglia viene trasferita nel file selezionato.

Note costruttive

Spiegato anche il programma di gestione dal PC, non ci resta che darvi qualche consiglio riguardante la realizzazione del circuito. Tutti i componenti prendono posto su una piccola basetta stampata, ottenibile per fotoincisione dopo aver ricavato la necessaria pellicola da una stampa su carta da lucido o acetato della traccia lato rame scaricabile gratuitamente dal nostro sito Web (www.elettronica.in.it). Ottenuta la basetta, da una parte inserite e saldate le resistenze e i diodi, poi gli zoccoli per il microcontrollore e il 74HC125, i condensatori e il quarzo, quindi il regolatore in TO-220 (da tenere con l'aletta metallica rivolta all'esterno dello stampato) e, in ultimo, il connettore USB di tipo B, da c.s. Quest'ultimo è normalmente rivestito da uno schermo metallico terminante con due linguette da inserire nei fori allo scopo previsti nello stampato e stagnare: tale operazione conferisce maggior robustezza all'insieme ed evita che, tirando inavvertitamente il cavo USB, il connettore possa essere divelto.

Per completare la realizzazione dovete stagnare, ma stavolta dal lato opposto della basetta, quindi direttamente sulle piazzole corrispondenti, dal lato rame, il lettore per la SD-Card; nel saldare i suoi terminali prestate attenzione a non eccedere con lo stagno, altrimenti lo spessore delle saldature può risultare tale da ostacolare o impedire del tutto l'inserimento della card stessa.

Per tutte le fasi del montaggio del circuito seguite l'apposito disegno di disposizione dei componenti, che indica l'esatto orientamento degli elementi polarizzati. Il microcontrollore già programmato va inserito dopo aver completato le saldature, orientandolo come indicato dalla disposizione componenti; lo stesso vale per il 74HC125.

Siamo arrivati alla conclusione di questo piccolo progetto, spaziando tra realizzazione circuitale, sviluppo firmware e software.

Come vedrete in pratica, pur essendo relativamente semplice, il sistema ci permette di prendere confidenza con i nuovi supporti di memorizzazione e di fare un po' di esperienza da accantonare per affrontare progetti più complessi: ad esempio, potrà tornare utile per analizzare i settori chiave della formattazione FAT16, come il boot-sector e la root-directory.

A parte l'impiego didattico, il lettore/scrittore è utile per scaricare in un computer dei dati scritti da un apparato di vario genere nella SD-Card con cui è equipaggiato o per trasferire in una SD dei file presenti nel Personal Computer.

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 stagiate 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.



visita il nostro sito per il dettaglio delle note tecniche

www.mdsrl.it

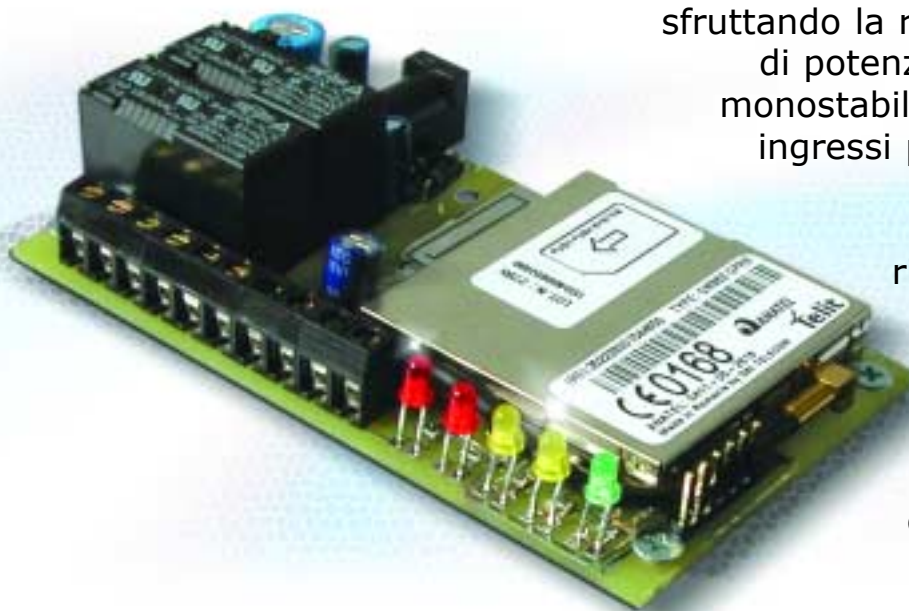
millennium
md
dataware

millennium dataware srl
parco scientifico e tecnologico
15050 rivalta scrivia - tortona (al)
tel. 0131 860.254 fax 0131 860157
www.mdsrl.it info@mdsrl.it

* i prezzi si intendono iva esclusa e saldati sul singolo pezzo - ordini minimo 2 pezzi

Telecontrollo GSM con antenna integrata

di Boris Landoni



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. Prima puntata.

Vi ricordate i telefonini di alcuni anni fa? Avevano tutti un'antenna più o meno lunga e vistosa; di quest'antenna, nei cellulari in commercio oggi, non esiste più traccia. Che abbiano inventato un apparato radio funzionante senza antenna? Nessuna notizia a proposito... Che vengano utilizzate frequenze molto più alte con antenne molto più piccole? Mah, ci risulta che le frequenze utilizzate siano sempre le stesse... E allora? Provate ad aprire un cellulare moderno o un'antenna a stilo per impieghi GSM: gratta-gratta, scoprirete l'antenna; già, non è nient'altro che una piccola spirale

ottenuta con una pista opportunamente sagomata del circuito stampato. Il pensiero è subito corso al nostro telecontrollo GSM (fascicolo n° 99) e alle notevoli difficoltà incontrate per cercare un'antenna compatta e di basso costo; allora ci siamo detti: "... e noi chi siamo, gli ultimi arrivati?" No di certo! Abbiamo quindi progettato una nuova versione di questo telecontrollo integrando l'antenna sullo stampato. Non solo. abbiamo anche provato e deciso di utilizzare - visti i buoni risultati - un modulo della Telit, precisamente il modello GM862-GPRS. Già che eravamo in ballo abbiamo

anche aggiunto nuove funzionalità in modo da migliorarne ulteriormente le prestazioni. Abbiamo così messo a punto un telecomando più performante ad un costo sicuramente più contenuto: un circuito, insomma, alla portata di tutti.

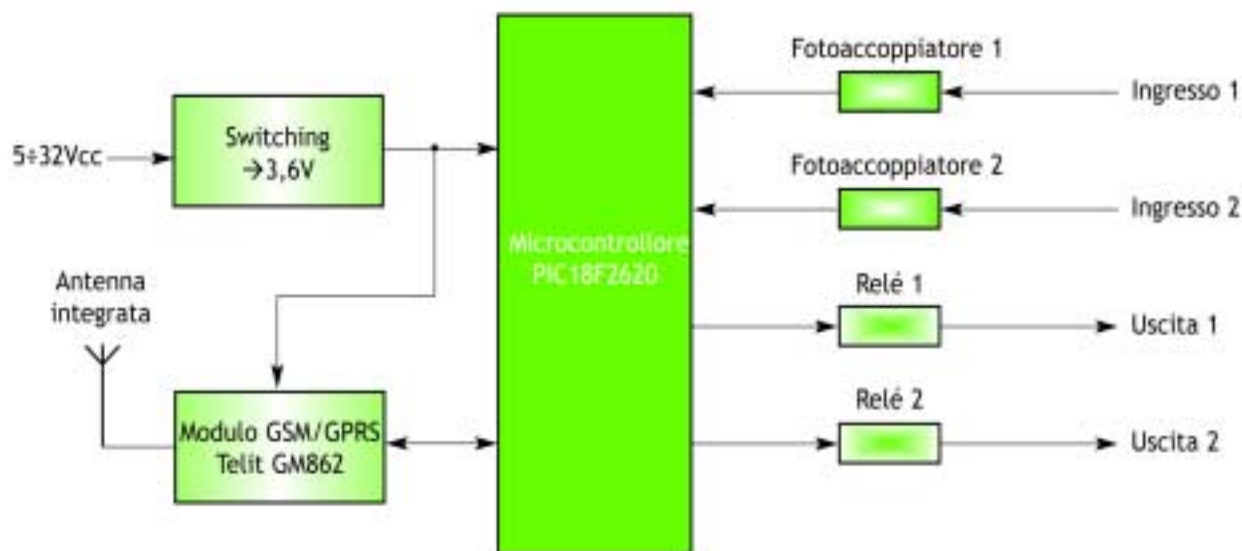
Come funziona

Il telecomando dispone di due relé ed altrettanti fotoaccoppiatori; i relé consentono di attivare localmente apparati comandabili elettricamente ed anche di realizzare la funzione “apricancello”, mentre gli ingressi optoisolati consentono di rilevare tensioni esterne garantendo un iso-

dunque un modulo di controllo a distanza bidirezionale, utilizzabile ovunque serva azionare carichi elettrici e leggere condizioni tramite tensioni; l'apparato dispone di due uscite a relé attivabili mediante appositi SMS inviati da un telefono GSM precedentemente abilitato. Inoltre, sempre via SMS, l'unità avverte un massimo di otto utenti del verificarsi di un cambiamento della tensione presente sugli ingressi o di un avvenuto spegnimento a causa della mancanza della tensione di alimentazione, condizione, questa, utile a segnalare che i dispositivi sotto controllo sono stati

azionare una centralina per apricancello comandabile da contatto sia normalmente chiuso, che normalmente aperto. Prima di addentrarci nel dettaglio tecnico riepiloghiamo le due funzioni dell'unità: nel modo telecomando, gestisce i relé a seconda dei comandi ricevuti, esclusivamente sotto forma di SMS, da un massimo di 8 numeri telefonici inseriti con gli appositi comandi; agli stessi otto numeri invia SMS e telefonate quando ritiene attivati (in base all'impostazione fatta in fase di caratterizzazione) gli ingressi a livello di tensione. L'invio della telefonata può

Schema a blocchi



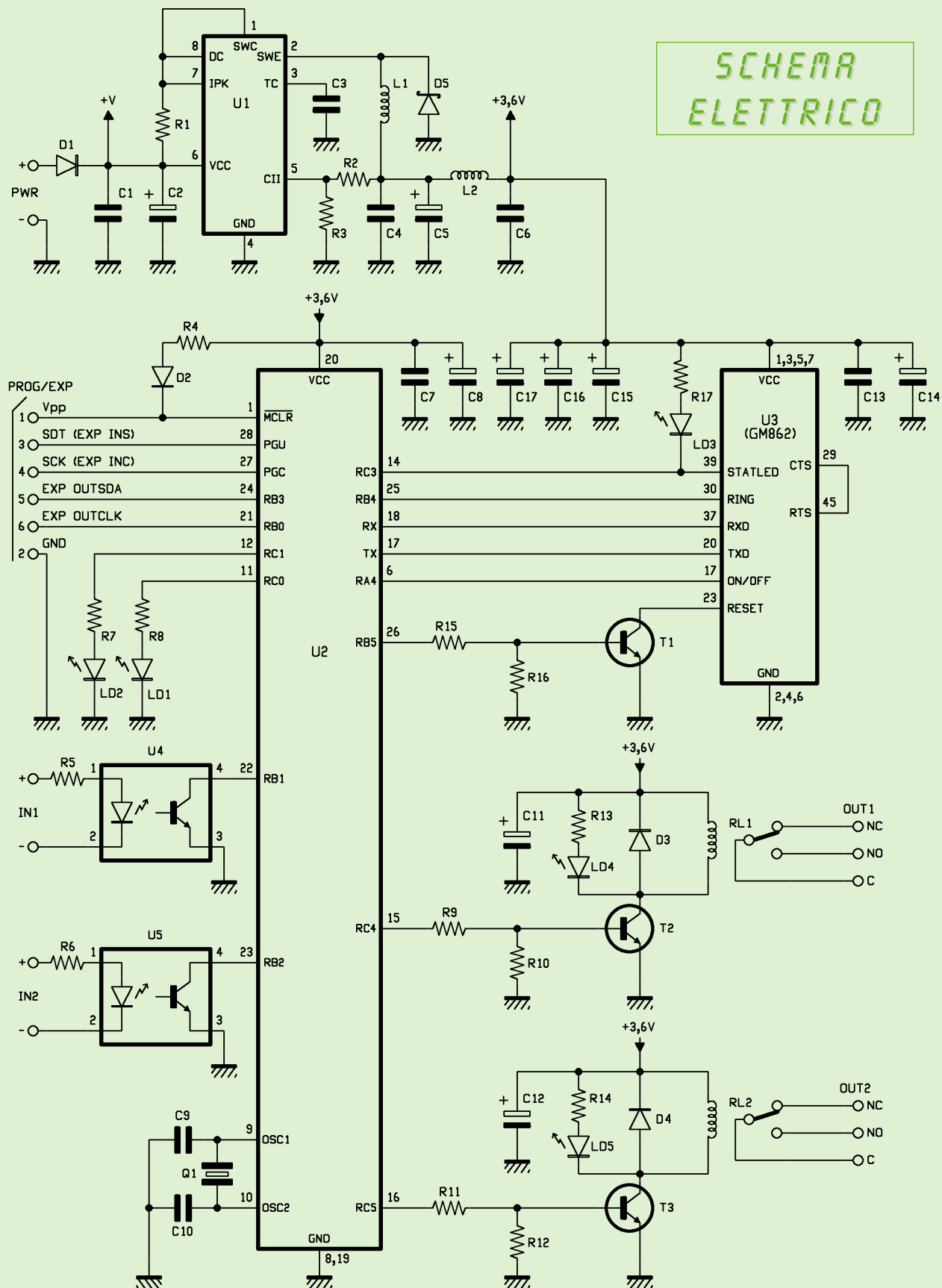
Il circuito è composto da un modulo GSM/GPRS gestito da un microcontrollore, da un convertitore switching con tensione di ingresso compresa tra 5 e 32 volt e da due input/output.

lamento galvanico tra telecomando ed apparato esterno.

Il dispositivo può essere configurato affinché invii degli SMS o effettui delle telefonate quando gli ingressi assumono lo stato definito nelle impostazioni; tale prerogativa aggiunge al telecomando la capacità di funzionare anche da teleallarme, abbinato, ad esempio, ad un impianto antifurto. Il circuito è

privati della tensione e possono quindi presentare anomalie o richiedere un intervento a distanza. Completa il set di funzioni la modalità apricancello, che consiste in questo: chiamando il telecomando da uno dei telefoni abilitati, lo scambio del relé 1 si chiude per il tempo definito, in fase di caratterizzazione, con l'apposito comando (fino a 59 secondi) e può quindi

essere richiesta in aggiunta o in alternativa alla spedizione dell'SMS. Riguardo alle chiamate, va detto che il circuito non trasmette alcunché, ma si limita ad effettuare qualche squillo al telefono della persona chiamata: scopo di ciò è richiamare l'attenzione e segnalare il verificarsi di un'anomalia con maggiore tempestività rispetto all'SMS; non va infatti ➤



dimenticato che i messaggi di testo non sempre vengono recapitati in tempo reale.

I relé di uscita possono essere attivati tanto in modo impulsivo, quanto in bistabile: il comando è unico e per ottenere le due modalità basta definire opportunamente il parametro tempo; se si invia un SMS per forzare l'attivazione monostabile di un relé già attivato in modo bistabile, il comando ha l'effetto di far tornare a riposo il relé stesso per il tempo definito nel messaggio. Scaduto l'intervallo, il relé torna eccitato, ossia ripristina la modalità bistabile, modalità che può essere disattivata solo con un SMS che contenga la disattivazione bistabile, oppure privando il circuito dell'alimentazione quando non sia attivata la funzione di ripristino dopo blackout; quest'ultima, alla successiva accensione riporta i relé nelle condizioni in cui si trovavano prima dell'evento. Resta inteso che abili-

- Bibanda GSM 900/1.800 MHz;
- 2 uscite a relé;
- 2 ingressi a livello di tensione ($3 \div 30$ V);
- SMS in caso di attivazione ingressi;
- Chiamata in caso di attivazione ingressi;
- Numeri abbinabili per allarme: 8;
- Funzione apricancello a costo zero;
- Numeri abbinabili per apricancello: 200;
- Carico applicabile alle uscite: 250V, 10A;
- Alimentazione: $5 \div 32$ Vcc, 300 mA max.

Specifiche tecniche

tando tale funzione, un relé attivo in modo bistabile può essere interdetto solo con il comando di disattivazione. Quanto alla modalità apricancello, il telecomando si comporta un po' come il ricevitore del tradizionale radiocomando: ricevendo la chiamata, attiva RL1 per un tempo definito da un apposito messaggio di configurazione ($1 \div 59$

secondi). Per casi particolari, abbiamo previsto la possibilità di configurare il relé affinché, a seguito di una telefonata, si attivi in modo bistabile: in questo caso ogni chiamata proveniente da numeri abilitati porterà il relé nella condizione opposta a quella in cui si troverà al momento. Per l'apricancello abbiamo previsto la possibilità di abbinare ➤

Radiomicrofoni professionali

Trasmettitore con microfono a clips da utilizzare nei sistemi FR217 e FR217D in sostituzione del radiomicrofono palmare. Ideale per convegni, dimostrazioni, spettacoli teatrali, ecc. Frequenza di lavoro: 181.660 MHz.

Trasmettitore a clips

FR217CL Euro 92,⁰⁰

Radiomicrofono in VHF formato da un trasmettitore palmare e da un ricevitore diversity. Questa particolare tecnologia consente di evitare l'effetto evanescenza. Frequenza di lavoro: 181,660 MHz. Il set viene fornito con una valigetta di plastica nera.

Set Radiomicrofono VHF

FR217 Euro 110,⁰⁰

Set Radiomicrofono UHF

FR217UHF Euro 205,⁰⁰

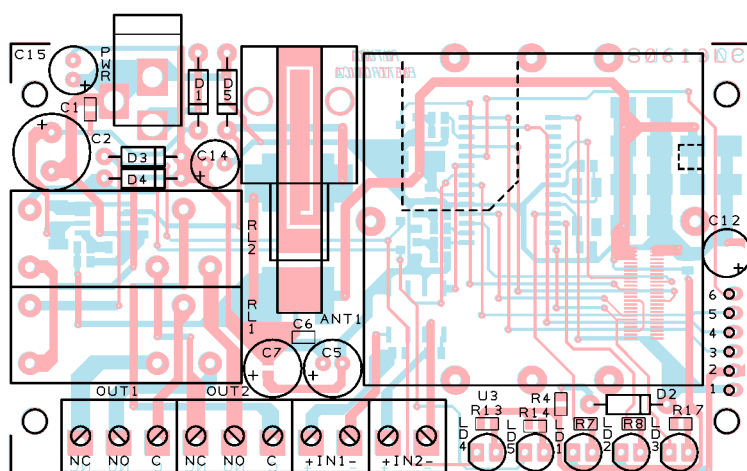
CARATTERISTICHE TECNICHE:
Stabilità in frequenza: $\pm 0.005\%$ / 25°C ; Modulazione: FM (F3E); Risposta in frequenza: 50Hz - 15kHz; Potenza RF: 10mW; Alimentazione: 1 x 9V; Assorbimento: $< 20\text{mA}$; Dimensioni: 106 x 66 x 23mm; Peso: 80g.

CARATTERISTICHE TECNICHE:
RICEVITORE
Frequenza di lavoro: banda VHF; Stabilità in frequenza: $\pm 0.005\%$ / 25°C ; Modulazione: FM (F3E); Distorsione T.H.D.: $< 0.8\%$; Rapporto S/N: $> 90\text{dB}$; Risposta in frequenza: 50Hz - 15kHz; Uscita segnale: 0-300mV, 10kOhm; Connettore: 6.35mm jack; Alimentazione: DC 12V/300mA (adattatore di rete incluso); Dimensioni: 152 x 98 x 36mm; Peso: 226g.
MICROFONO CON TRASMETTITORE
Stabilità in frequenza: $\pm 0.005\%$ / 25°C ; Modulazione: FM (F3E); Risposta in frequenza: 50Hz - 15kHz; Potenza di uscita RF: 10mW; Alimentazione: 1 x 1.5V AA; Assorbimento: 60mA; Autonomia: superiore a 12 ore con batteria alcalina; Lunghezza: 233mm; Peso: 210g.

CARATTERISTICHE TECNICHE:
RICEVITORE
Frequenza di lavoro: banda UHF; Stabilità in frequenza: $\pm 0.005\%$ / 25°C ; Modulazione: FM (F3E); Distorsione T.H.D.: $< 0.6\%$; Dinamica: $> 110\text{dB}$; Rapporto S/N: $> 100\text{dB}$; Risposta in frequenza: 50Hz - 15kHz; Oscillatore: controllato al quarzo; Sensibilità RF: 3µV / 20dB Sinad; Squelch: controllo esterno; Uscita segnale: bilanciata 60mV/600 Ohm, sbilanciata 750mV 5kOhm; Connettore: 6.35mm jack; Alimentazione: DC 12V/500mA (adattatore di rete incluso); Dimensioni: 222 x 114 x 36mm; Peso: 550g.
MICROFONO CON TRASMETTITORE
Stabilità in frequenza: $\pm 0.005\%$ / 25°C ; Modulazione: FM (F3E); Deviazione massima: $\pm 40\text{kHz}$; Risposta in frequenza: 50Hz - 15kHz; Potenza di uscita RF: 10mW; Alimentazione: 1 x 9V; Assorbimento: 30mA; Autonomia: superiore a 8 ore con batteria alcalina; Lunghezza: 235mm; Peso: 210g.

Tutti i prezzi sono da intendersi IVA inclusa.

Disponibili presso i migliori negozi di elettronica o nel nostro punto vendita di Gallarate (VA).
Caratteristiche tecniche e vendita on-line: www.futuraneet.it
Via Adige, 11 - 21013 Gallarate (VA) - Tel. 0331/799775 - Fax. 0331/778112 - www.futuraneet.it



ELENCO COMPONENTI:

- R1: 0,1 ohm 1W
- R2: 2,2 kohm
- R3: 1,2 kohm
- R4÷R5, R9: 4,7 kohm
- R7, R8, R17: 470 ohm
- R10, R12, R16: 10 kohm
- R11: 4,7 kohm
- R13, R14: 1 kohm
- R15: 4,7 kohm
- C1, C4: 100 nF multistrato SMD
- C2: 470 µF 35 VL elettrolitico
- C3: 100 pF ceramico SMD
- C5: 220 µF 16 VL elettrolitico
- C6, C7, C13: 100 nF multistrato SMD
- C8: 470 µF 6,3 VL tantalio SMD
- C9, C10: 10 pF ceramico SMD
- C11, C12: 100 µF 16 VL elettrolitico
- C14÷C19: 470 µF 6,3 VL tantalio SMD
- Q1: quarzo 20 MHz SMD
- U1: MC34063
- U2: PIC18F2620 (MF613)
- U3: Modulo Telit GM862-GPRS
- U4, U5: TLP181
- D1÷D4: 1N4007
- D5: 1N5819
- T1÷T3: BC817
- LD1, LD2: led 3 mm giallo

re il dispositivo a ben 200 numeri, cosa che ci permette di proporlo anche come attuatore a distanza per grandi condomini, aziende o enti pubblici. C'è poi una seconda lista di numeri relativa alla funzione telecomando; occupa otto posizioni che corrispondono a quelli che chiamiamo "numeri privilegiati", ossia gli otto abilitati al comando di entrambi i relé e alla ricezione delle notifiche inerenti lo stato degli ingressi. Ciò vuol dire che, in fase di configurazione, tali numeri telefonici dovranno essere memorizzati nelle posizioni 1÷8.

Schema elettrico

Detto ciò, diamo adesso uno sguardo allo schema elettrico del dispositi-

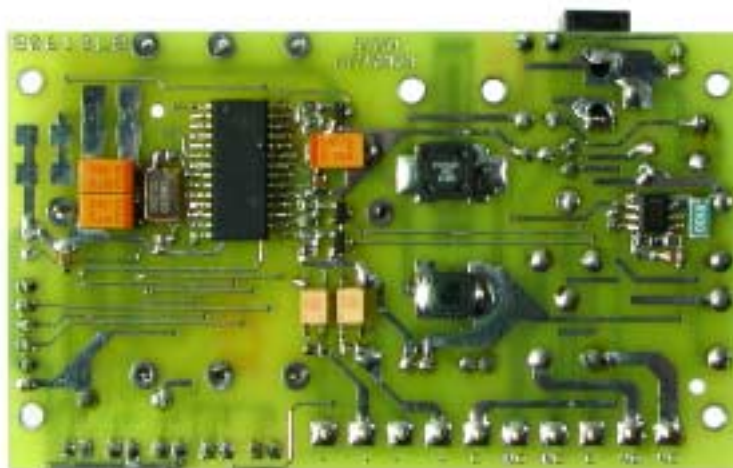
tivo. Partiamo dall'alimentazione, dicendo che il telecomando funziona con una tensione continua, anche non stabilizzata (applicata ai punti + e - PWR) di valore compreso tra 5 e 32 V, che viene filtrata a valle del diodo di protezione contro l'inversione di polarità (D1) mediante i condensatori C1 e C2; dalla differenza di potenziale ai capi di questi ultimi il regolatore switching U1 ricava 3,6 V stabilizzati, con i quali alimenta tutto il resto del circuito. U1 è uno step-down MC34063 che incorpora un regolatore PWM, il cui transistor d'uscita carica l'induttanza L1 con impulsi la cui larghezza dipende dalla tensione livellata da C4 e C5 e retrocessa dal partitore resistivo

R2/R3; più la tensione scende per effetto del carico, più si allargano gli impulsi e viceversa. Il tasso di retroazione del partitore R2/R3 stabilisce la tensione d'uscita, che nel nostro caso è 3,6 volt. In pratica il modulatore interno pilota la base di un transistor NPN interno al chip, il cui emettitore eroga impulsi di corrente che attraversano L1 dirigendosi verso C4 e C5; nelle pause il transistor è interdetto e l'induttore tende a mantenere il flusso di corrente ma, essendosi aperto il, circuito che l'alimentava fino all'istante precedente, restituisce l'energia immagazzinata forzando lo scorrimento di corrente nei condensatori e attraverso il diodo Schottky D5. Quest'ultimo serve anche per

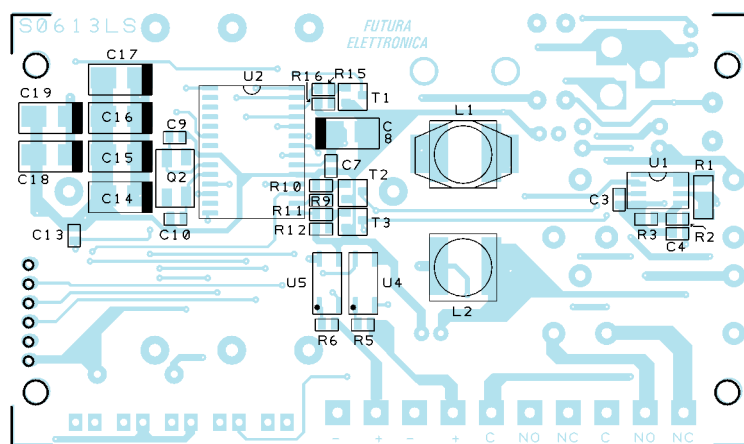
LD3: led 3 mm verde
 LD4, LD5: led 3 mm rosso
 L1: Bobina 22 μ H - 2A
 L2: Bobina 1 μ H - 3A
 RL1: Relè 5V 1 scambio
 RL2: Relè 5V 1 scambio

Varie:

- Morsettiera 2 poli (2 pz.)
- Morsettiera 3 poli (2 pz.)
- plug alimentazione
- Strip maschio 6 pin
- Connettore da c.s. per GM862
- Connettore per antenna GSM
- Circuito stampato codice S613



I componenti prendono posto su entrambe le facce del circuito stampato: dal lato superiore si montano i relè, i led e il modulo GSM/GPRS (sull'apposito connettore miniatura, da saldare badando di non cortocircuitare pin vicini). Dall'altro si sistemano il micro, il regolatore U1 e i condensatori SMD. L'antenna del Telit è su stampato e va collegata (vedi freccia) con uno spezzone di cavo coassiale intestato su un connettore MMCX.



proteggere il transistor interno all'integrato, che altrimenti verrebbe danneggiato dall'extratensione inversa generata dalla L1 nel momento dell'interdizione. Notate i numerosi condensatori di filtro posti sulla linea dei 3,6 volt: il loro scopo è sopprimere i disturbi e impedire gli abbassamenti impulsivi della tensione di alimentazione che si verificherebbero, causa il forte assorbimento di corrente, ad ogni entrata in trasmissione del modulo Telit.

La gestione dell'intero telecontrollo è affidata a un microcontrollore PIC18F2620 Microchip, che, inizializzate le linee di I/O, provvede a monitorare la condizione logica degli ingressi a livello di tensione

(RB1 ed RB2) ma anche le linee RB4, RC3, RX, che servono a ricevere dal modulo cellulare le principali segnalazioni; precisamente, RB4 viene usata per rilevare l'arrivo delle chiamate entranti, mentre RC3 controlla il led di stato. L'uscita corrispondente (STAT-LED) pulsa alla frequenza di 1 Hz quando il GM862 sta cercando la rete radiomobile, mentre fornisce un impulso a zero logico della durata di 0,5 s. seguito da una di 2 secondi quando il modulo ha agganciato il segnale. Dunque, dalla frequenza e durata degli impulsi con cui il Telit fa pulsare il led di "campo", il PIC18F2620 ricava quelle che sono le condizioni di lavoro; più esattamente, se a

seguito dell'entrata in allarme di uno o di entrambi gli ingressi di allarme deve inviare gli SMS o effettuare le chiamate del caso, ma rileva che il modulo cellulare è fuori campo, mantiene in memoria gli allarmi e prepara le operazioni del caso. In pratica, attende che il modulo Telit si riconnetta alla rete GSM/GPRS e solo allora effettua la sequenza di avviso. L'analisi delle condizioni della rete radiomobile permette al telecontrollo di garantire la corretta effettuazione del ciclo di chiamate ed SMS.

Il micro contiene un UART accessibile dai piedini 17 (trasmissione) e 18 (ricezione): mediante il TX interroga ciclicamente il modulo Telit al fine di verificare l'arrivo di ➤

SMS. TX ed RX servono, durante le procedure di chiamata, ricezione e trasmissione messaggi, per la comunicazione tra il microcontrollore e il GM862.

Come accennato, il telecontrollo permette di monitorare lo stato dei due ingressi, per leggere i quali il micro sfrutta i fotoaccoppiatori U4 e U5, che gli permettono di rilevare tensioni (di valore compreso fra 3 e 30 V) in circuiti elettrici di qualsiasi tipo restando galvanicamente isolato da essi; questa scelta consente di tutelare il telecontrollo da scariche o correnti di massa dovute, ad esempio, al fatto che l'apparato al quale viene collegato possa avere un potenziale di massa non corrispondente a quello dell'alimentatore connesso ai punti + e - PWR. L'isolamento galvanico permette anche di pilotare gli input con due differenti circuiti, mantenendoli isolati tra loro, cosa non possibile se IN1 e IN2 facessero riferimento alla stessa massa.

Il rilevamento agli ingressi funziona così: quando ad uno di essi è applicata una tensione di almeno 3 volt, il fotoaccoppiatore relativo ha il led (pin internamente connesso con l'anodo al pin 1 e il catodo al 2) acceso e il fototransistor d'uscita in conduzione, cosicché il collettore (piedino

5) è a circa zero volt, per effetto della caduta sul resistore di pull-up configurato, durante l'inizializzazione degli I/O, per la linea del PIC. Se l'ingresso non è polarizzato, l'optoisolatore è interdetto e il suo pin 4 risulta a livello alto.



LD1 ed LD2 segnalano l'attivazione rispettivamente degli ingressi 1 e 2; per attivazione si intende il raggiungimento, sulle linee

RB1 ed RB2 delle condizioni logiche definite in fase di configurazione del telecontrollo, secondo le modalità che descriveremo dettagliatamente nella prossima puntata. Per ora accontentatevi di sapere che ogniqualvolta si verifica la condizione impostata, il circuito invia un SMS contenente la notifica dell'evento. È anche possibile definire un intervallo di osservazione, ossia stabilire per quanto tempo la situazione di anomalia deve permanere prima di essere considerata allarme e determinare l'invio del messaggio di notifica e l'effettuazione della chiamata al numero o ai numeri tra gli otto definiti per il teleallarme; ciò si imposta, ingresso per ingresso. Per il comando di utilizzatori esterni, il microcontrollore si avvale dei relé

RL1 e RL2, pilotati mediante le linee RC4 ed RC5; più esattamente, quando ne pone una a livello logico alto manda in saturazione il transistor corrispondente, determinando l'attivazione del rispettivo relé, condizione evidenziata dall'accensione del led associato, il quale, essendo alimentato in parallelo alla bobina del relé, si attiva insieme al relé stesso. Delle uscite di RL1/RL2 viene reso disponibile l'intero scambio, così da poter gestire sia circuiti che richiedono un contatto normalmente chiuso, sia che abbiano bisogno di un contatto normalmente aperto. L'osservatore attento avrà certo notato che, sebbene siano del tipo con bobina a 5 volt, RL1 ed RL2 nel nostro circuito vengono fatti funzionare a 3,6 V; ebbene, dalle prove effettuate su numerosi campioni possiamo dire che non ci sono controindicazioni in tal senso. Delle linee di I/O usate dal micro abbiamo finora tralasciato /MCLR, PGU, PGC, RB0 ed RB3: il software le destina a costituire un bus usato sia per la programmazione in-circuit del PIC18F2620 che per l'eventuale espansione; più esattamente, PGU e PGC sono, rispettivamente, il Serial Data (SDA) e il Serial Clock (SCL) del bus I²C per la comunicazione con il programmatore, linee riferite alla massa comune GND. Nella modalità espansione, riservata, peraltro, a

Per il **MATERIALE**

Il progetto descritto in queste pagine è disponibile in scatola di montaggio (cod FT613K) al prezzo di 198,00 Euro. Il kit comprende tutti i componenti, la basetta forata e serigrafata, il modulo GSM ed il microcontrollore già programmato. Il telecontrollo è anche disponibile già montato e collaudato (cod. TDG33) allo stesso prezzo (198,00 Euro) del kit. 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>**

futuri sviluppi, tali I/O ed RB0/RB3 costituiscono i bus di comunicazione seriale con gli eventuali moduli di espansione.

Bene, descritta l'unità dal punto di vista elettrico, vediamo ora come funziona: all'accensione il microcontrollore inizializza le linee di I/O e, per prima cosa, lancia la modalità *fast setup*: si tratta di una funzione che consente di configurare rapidamente il funzionamento del dispositivo. In pratica, se, nei primi tre minuti di esercizio, riceve una chiamata, memorizza nella prima posizione il numero da cui giunge: il telefono corrispondente verrà abilitato sia al comando apri-cancello, sia al telecontrollo (gestione di entrambi i relé e ricezione di chiamate ed SMS relativi agli ingressi). Basta fare uno squillo (ovviamente senza nascondere l'ID...) e il GSM salva il numero nella prima posizione di memoria. Fatto ciò, abbandona il *fast setup* entrando nel normale funzionamento, condizione nella quale il circuito si porta anche se allo scadere dei tre minuti iniziali non riceve alcuna chiamata.

Va detto che per gestire le uscite a relé il circuito deve ricevere SMS, contenenti specifici comandi, da numeri telefonici abilitati a farlo, così come per attivare RL1 nella modalità apri-cancello occorre che il telecontrollo abbia memorizzato almeno un numero da cui ricevere

la telefonata di comando. Ebbene, i modi per configurare il circuito al fine di memorizzarvi i numeri abilitati a comandarlo sono due: il primo consiste nell'inviare gli opportuni SMS secondo la sintassi richiesta e corredati dalla password, indispensabile ad evitare che la configurazione possa essere operata da estranei. Il secondo modo consiste nell'introdurgli, mediante *fast setup*, il numero del telefono cellulare abilitato a fare le successive impostazioni tramite SMS configuranti; in tal caso la gran parte dei messaggi potrà essere inviata senza la password, perché proveniente da un numero già abilitato. La password predefinita del circuito è 12345; va inserita nei messaggi di configurazione ed anche in quello per l'eventuale personalizzazione della password. All'interno di un SMS possono essere contenuti più comandi (ad esempio: attivazione di un relé e rilascio dell'altro) purché non in contrasto tra loro.

La realizzazione

Vediamo, infine, qualche nota pratica: una volta procurata la basetta (è del tipo a doppia faccia e può essere ottenuta per fotoincisione dalle tracce disponibili nel nostro sito www.elettronica.in.it) inserite e montate i componenti usando un saldatore a punta sottile da non più di 30 watt di potenza; disponete i componenti iniziando con il con-

nettore miniatura per il modulo Telit. Sistemate via-via gli altri elementi, quasi tutti SMD; il micro deve essere acquistato già programmato. Non dimenticate di stagnare da entrambe le facce dello stampato i componenti passanti (se acquistate il c.s. non occorre, perché è a fori metallizzati). Prima di inserire la SIM nel GM862, con un comune cellulare disattivate il PIN, perché il software del PIC18F2620 non gestisce il codice d'accesso. Un discorso particolare lo merita l'antenna: siccome è costituita da una pista dello stampato, bisogna collegarla al modulo (dopo averlo inserito nel suo connettore e fissato con le apposite linguette) con un cavetto coassiale, l'anima del quale va stagnata sull'esterno della pista. Potete anche usare un'antenna esterna: basta montare un connettore sul c.s. e collegare ad esso il coassiale che giunge dal modulo. Ovviamente qualora il dispositivo venga inserito in un contenitore, quest'ultimo dovrà essere necessariamente plastico; utilizzando un contenitore metallico dovrete impiegare un cavo adattatore d'antenna con presa da pannello FME ed un'antenna bibanda.

Il prossimo mese concluderemo la descrizione del nostro telecontrollo occupandoci del software e di tutte le procedure di programmazione remota descrivendo dettagliatamente i vari comandi.

RM ELETTRONICA SNC

vendita componenti elettronici
rivenditore autorizzato:



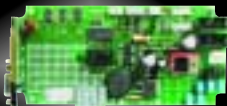
Via Val Sillaro, 38 - 00141 ROMA - tel. 06/8104753

Controllo accessi e varchi con transponder attivi e passivi

CONTROLLO VARCHI A MANI LIBERE

Sistema con portata di circa 3-4 metri realizzato con transponder attivo (MH1TAG). L'unità di controllo può funzionare sia in modalità stand-alone che in abbinamento ad un PC. Essa impiega un modulo di gestione RF (MH1), una scheda di controllo (FT588K) ed un'antenna a 125 kHz (MH1ANT). Il sistema dispone di protocollo anticollisione ed è in grado di gestire centinaia di TAG attivi.

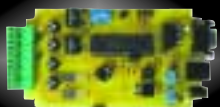
MODULO DI GESTIONE RF



Modulo di gestione del campo elettromagnetico a 125 kHz e dei segnali radio UHF; da utilizzare unitamente al kit FT588K ed ai moduli MHTAG e MH1ANT per realizzare un controllo accessi a "mani libere" in tecnologia RFID. Il modulo viene fornito già montato e collaudato.

MH1 - euro 320,00

SCHEDA DI CONTROLLO



Scheda di controllo a microcontrollore da abbinare ai dispositivi MH1, MH1TAG e MH1ANT per realizzare un sistema di controllo accessi a "mani libere" con tecnologia RFID.

FT588K - euro 55,00

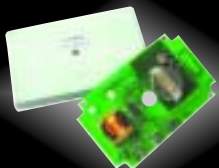
ANTENNA 125 KHZ



Antenna accordata a 125 kHz da utilizzare nel sistema di controllo accessi a "mani libere". In abbinamento al modulo MH1 consente di creare un campo elettromagnetico la cui portata raggiunge i 3-4 metri. L'antenna viene fornita montata e tarata.

MH1ANT - euro 45,00

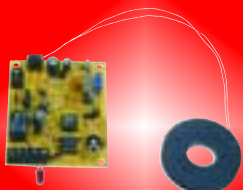
TRANSPONDER ATTIVO RFID



Tessera RFID attiva (125 kHz/433 MHz) da utilizzare nel sistema di controllo accessi a "mani libere". La tessera viene fornita montata e collaudata e completa di batteria al litio.

MH1TAG - euro 60,00

LETTORI E INTERFACCE 125 KHz



SERRATURA CON TRANSPONDER

Chiave elettronica con relè d'uscita attivabile, in modo bistabile o impulsivo, avvicinando un TRANSPONDER al solenoide nel raggio di 5÷6 centimetri. La scheda viene attivata esclusivamente dai TRANSPONDER i cui codici sono stati precedentemente memorizzati nel dispositivo mediante una semplice procedura di abilitazione. Il sistema è in grado di memorizzare sino ad un massimo di 200 differenti codici. L'apparecchiatura viene fornita in scatola di montaggio (contenitore escluso). Non sono compresi i TRANSPONDER.

FT318K - euro 35,00

PORTACHIAVI CON TRANSPONDER

Transponder passivo adatto per sistemi a 125 kHz. Programmato con codice univoco a 64 bit. Versione portachiavi.

TAG-1 - euro 3,50



PORTACHIAVI CON TESSERA ISOCARD

Transponder passivo adatto per sistemi a 125 kHz. Programmato con codice univoco a 64 bit. Versione tessera ISO.

TAG-2 - euro 3,50



SISTEMI CON PC

LETTORE DI TRANSPONDER RS485

Consente di realizzare un sistema composto da un massimo di 16 lettori di transponder passivi (cod FT470K) e da una unità di interfaccia verso il PC (cod FT471K). Il collegamento tra il PC e l'interfaccia avviene tramite porta seriale in formato RS232. La connessione tra l'interfaccia ed i lettori di transponder è invece realizzata tramite un bus RS485. Ogni lettore di transponder (cod FT470K) contiene al suo interno 2 relè la cui attivazione o disattivazione viene comandata via software. Il dispositivo viene fornito in scatola di montaggio la quale comprende anche il contenitore plastico completo di pannello serigrafato.

FT470K - euro 70,00



INTERFACCIA RS485

Consente di interfacciare alla linea seriale RS232 di un PC da 1 ad un massimo di 16 lettori di transponder (cod. FT470K). Il kit comprende tutti i componenti, il contenitore plastico ed il software di gestione.

FT471K - euro 26,00



LETTORI E INTERFACCE 125 KHz

LETTORE DI TRANSPONDER SERIALE RS232

Lettore di transponder in grado di funzionare sia come sistema indipendente (Stand Alone) sia collegato ad un PC col quale può instaurare una comunicazione (PC Link). Munito di 2 relè per gestire dispositivi esterni e di una porta seriale per la connessione al PC. L'apparecchiatura viene fornita in scatola di montaggio (compreso il contenitore serigrafato). I transponder sono disponibili separatamente in vari formati.

FT483K - euro 62,00



Tutti i prezzi si intendono IVA inclusa.

Client FTP con PIC e SD-Card

di Carlo Tauraso

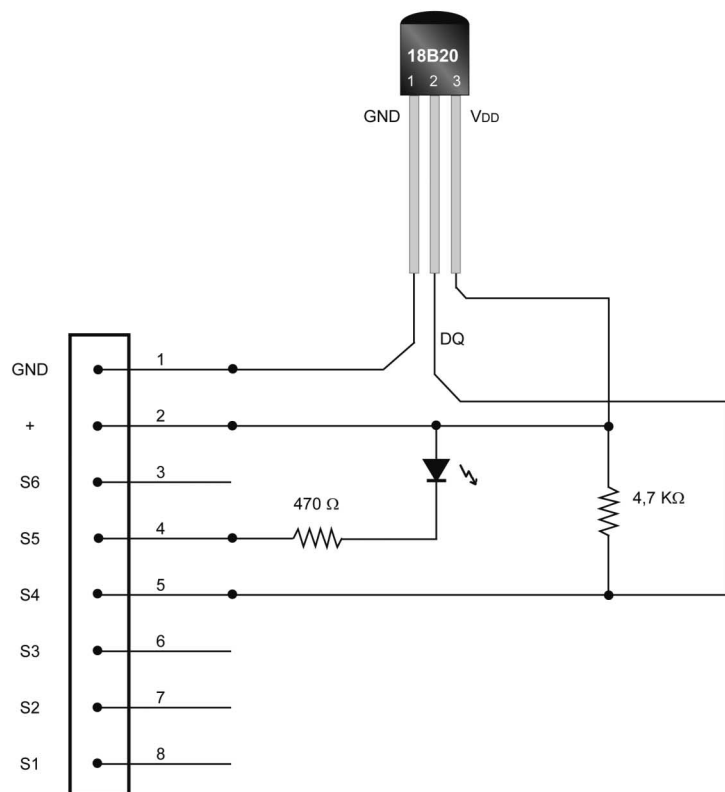


Utilizziamo un microcontrollore Microchip per pubblicare via FTP dati su un server Internet. Per la prima volta adottiamo un'interfaccia di rete che costruiamo interamente a partire da uno dei più popolari chip ethernet: l'RTL8019 di Realtek. Con uno specifico firmware, il circuito proposto può funzionare anche da Web-server. Seconda parte.

Costruire un client per pubblicare dei dati in un server FTP usando un microcontrollore e pochi altri componenti di contorno, è ormai una realtà assodata; chi è convinto della necessità di adottare un Personal Computer può ricredersi leggendo l'articolo con il quale, nel fascicolo di dicembre/gennaio 2005, abbiamo iniziato la descrizione di un apparato da connettere in rete locale o, mediante un router, ad Internet o a una WAN, per pubblicare i dati riguardanti la lettura di alcune sonde di temperatura in un server di rete secondo il protocollo FTP (File Transfer Protocol). Nella

prima puntata abbiamo esposto le problematiche inerenti alla realizzazione di un progetto del genere, non ultime quelle strettamente attinenti all'implementazione dell'interfaccia ethernet; quest'ultima problematica è stata affrontata ricorrendo a un chip prodotto dalla Realtek e siglato RTL8019: si tratta di un convertitore ISA/ethernet capace di assicurare una velocità di comunicazione di 10 Mbit/s, velocità che, sebbene sembri bassa (oggi le schede di rete assicurano data-rate di 100 o 1.000 Mbit/s.) nel nostro caso è più che sufficiente. Dopo aver visto l'hardware del client, in questa punta- ➤

Fig. 1



ta analizziamo il software; nello specifico, parliamo della parte che permette la lettura della temperatura da parte delle sonde e la preparazione dei dati per l'invio al server FTP passando dall'interfaccia di rete.

L'integrazione con la sonda

Il circuito di base può essere facilmente integrato

attraverso lo strip a 8 pin di cui è dotato. Predisponiamo quindi una piccola basetta sulla quale inseriamo la sonda DS18B20 ed un led di

sistema che ci permetterà di monitorare l'effettivo funzionamento. Abbiamo utilizzato RA3 come linea di acquisizione della temperatura, mentre RA4 funge da driver per il led di segnalazione. Lo schema risultante è quello riportato nella Fig. 1.

La sonda DS18B20 utilizza un protocollo di comunicazione chiamato 1-Wire e permette di misurare la temperatura ambiente in un range che va da -55 a +150 °C (da -10°C a +85°C con una accuratezza di +/- 0,5°C). Durante il campionamento il risultato della misura viene reso disponibile in due registri, secondo la struttura di Fig.2; in essa notiamo che un registro contiene il byte più significativo, la cui prima porzione è composta dai bit S, i quali permettono di stabilire il segno della temperatura rilevata: (S=0 indica i valori positivi, mentre S=1 vuol dire che la lettura è da intendersi negativa). Notate che i primi bit dell'MSB possono essere 4 o 5: il software del PIC configura la sonda per 4 bit di segno e 12 di dati. Di questi ultimi (quattro risiedono

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
LS Byte	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
MS Byte	S	S	S	S	S	2^6	2^5	2^4

Fig. 2

Tabella 1

Definizione	Descrizione
void OWTX(unsigned char DATO);	Invia un byte alla sonda. Serve per inviare comandi come ad esempio la sequenza di avvio della conversione della temperatura.
unsigned char OWRX(void);	Riceve un byte dalla sonda. Serve per ricevere in risposta dalla sonda i valori di temperatura.
unsigned char OWReset(void);	Permette di effettuare un reset del dispositivo. La funzione ritorna un valore pari a 1 se la sonda risponde correttamente inviando il bit di presenza, mentre ritorna un valore pari a 0 se la sonda non è presente.
unsigned char OWRX1(void);	Permette di gestire la ricezione di un singolo bit dal dispositivo. Permette di stabilire il termine dell'operazione di conversione da parte della sonda.

nell'MSB e 8 nel byte meno significativo o LSB) i primi otto bit esprimono la parte intera del valore assoluto della temperatura, mentre i restanti quattro costituiscono i decimali. Il valore complessivo si ottiene sommando i valori effettivi di ciascuna cifra o posizione binaria prima della virgola (parte intera) e dopo (decimali): ad esempio, se gli ultimi dodici bit sono del tipo 0000 0001 0000, la temperatura letta è di 1 grado centigrado; otteniamo il valore considerando che la parte

intera (a sinistra della virgola) vale 1 e quella decimale (i quattro bit a destra della virgola) “pesa” 0. Se invece abbiamo 0111 1101 0000 la sonda ha rilevato 125 °C: infatti i quattro bit a destra danno zero, mentre la parte a sinistra vale $1+4+8+16+32+64$, ossia 125. Ancora, 0111 1101 1000 corrisponde a una temperatura di 125,5 °C; infatti 1000 a destra della virgola corrisponde a $2^{-1}+0+0+0$, ovvero a $1/2=0,5$. Una porzione decimale del tipo 0001 vale invece 2^{-4} , ossia 0,0625.

Tutti gli esempi appena fatti valgono a condizione che i quattro bit del segno siano a 0000; se sono 1111 le temperature rappresentate dai restanti bit sono negative.

Le letture possono anche essere espresse in formato esadecimale, rappresentando l'insieme dei due byte nel loro ordine e ricordando che quello meno significativo è il

LISTATO 1

```
SCHEDAInit();
TickInit();
XEEInit(EA_BAUD(CLOCK_FREQ, 400000));
PGMInit();
StackInit();
FTPIInit();
SDBeginWrite(CONTAS);
inizio = TickGet();
```

opportunamente i dati della sonda. Questo accorgimento ha sostanzialmente uno scopo: distribuire tra server e client la logica di elaborazione del nostro applicativo, alleggerendo il firmware del client FTP. Per comunicare con la sonda Dallas abbiamo utilizzato una libreria che contempla il protocollo 1-Wire, secondo il quale ogni operazione viene iniziata dal dispositivo master (nel nostro caso è il PIC). La prima fase da considerare è il reset del dispositivo che si compone di un impulso proveniente dal master seguito da un segnale di presenza

ne rilascia nuovamente il bus, pertanto la resistenza di pull-up riporta la linea dati a livello alto. Per i nostri scopi è sufficiente estrarre i primi due byte (che contengono la temperatura) quindi durante la lettura scaricheremo i rimanenti 7 byte. La libreria in questione si compone di 4 funzioni, che sinteticamente raggruppiamo nella tabella 1. Nella stessa libreria troviamo due file: *onewire.c* contiene il codice sorgente e *onewire.h* contiene, invece, le definizioni corrispondenti.

La fase di Inizializzazione

Il flusso logico che permette di far funzionare correttamente il circuito del client-FTP può essere suddiviso idealmente in tre fasi fondamentali: 1) inizializzazione dei dispositivi e dello Stack TCP/IP; 2) ciclo di campionamento con caricamento SD; 3) sessione FTP temporizzata (ogni 6 ore) con scarico dati.

All'interno di queste fasi viene richiamata regolarmente una routine che gestisce le attività dello stack chiamata *StackTask()*. In questo modo, pur effettuando un'operazione di registrazione dati, il dispositivo è in grado di rispondere ad eventuali interazioni richieste attraverso l'interfaccia di rete. Un tipico esempio è dato dalle richieste ICMP, come ad esempio quella generata durante il ping.

Abbiamo, chiaramente ridotto al minimo tali possibilità per evitare che ciò comporti una perdita di prestazioni generale. Premettiamo che non analizzeremo concretamente tutto il funzionamento dello stack ➤

LISTATO 2

```
static void SCHEDAInit(void)
{
    CMCON = 0b00000111;
    ADCON1 = 0x07;
    TRISA = 0x00;
    while(SDInit());
    CONTAS = 0x00;
    CONTAR = 0x00;
    while (OWReset()!=1);
    DelayMs(1000);
    TOCON = 0;
    INTCON_GIEH = 1;
    INTCON_GIEL = 1;
}
```

PORTA configurata come linee digitali in output.

Inizializzazione in modalità SPI della SD-Card.

Azzeramento contatore settori e record scritti su SD.

Reset della sonda DS18B20 con pausa.

Attivazione dei segnali di interrupt.

decimale più a destra, mentre il più significativo è il primo dei bit di segno. Ad esempio, un valore pari a 00A2h corrisponde a una temperatura di 10,125°C, mentre FFF8h indica -0,5°C.

È importante tenere ben presente il formato della sonda, perché il microcontrollore non lo converte ma lo lascia tale e quale: i dati vengono usati come sono nella preparazione del file binario che verrà trasferito al server FTP. Sarà poi l'applicazione Web ad interpretare

da parte della sonda. Durante questa fase il microcontrollore porta a livello basso la linea per un minimo di 480 µs. e successivamente si pone in ricezione rilasciando il bus. La resistenza di pull-up porta, quindi, la linea a livello logico alto; non appena la sonda rileva questo cambiamento, attende per un intervallo di tempo che va da un minimo di 15 ad un massimo di 60 µs. e poi invia un impulso di presenza ponendo la linea a livello basso per un intervallo che va da 60 a 240 µs. Al termi-

LISTATO 3

```
#pragma interrupt HighISR save=section(".tmpdata")
void HighISR(void)
{
    TickUpdate();
}
#pragma code highVector=0x08
void HighVector (void)
{
    _asm goto HighISR _endasm
}
#pragma code
```

La funzione HighISR richiama a sua volta una funzione della libreria Tick.c che aggiorna i contatori per la temporizzazione delle transazioni.

Il vettore degli Interrupt prevede esclusivamente un salto alla procedura di gestione chiamata HighISR implementata sottoforma di funzione C18.

TCP/IP, riservandoci di farlo in una serie di puntate di approfondimento dedicate a tale scopo.

Qui diamo le sole informazioni necessarie a capire l'utilizzo di alcune funzioni all'interno dell'implementazione del protocollo FTP. Vediamo come avviene l'inizializzazione della nostra scheda. Il codice del main è visibile nel Listato 1. Come si vede, richiamiamo tutta una serie di funzioni che svolgono ognuna una inizializzazione specifica. La prima è la *SCHEDAInit()* che raccoglie le istruzioni di preparazione del PIC (vedere Listato 2). Attraverso il registro CMCON, disabilitiamo i comparatori che insistono sulla PORTD, che nel nostro caso viene usata principalmente per la comunicazione con l'RTL. Configuriamo quindi tutte le linee della PORTA come uscite, disabilitando i moduli A/D e definendo la direzione di output tramite il registro TRISA. A questo punto iniziamo la SD-Card con la funzione *SD-Init()*. Quest'ultima ritorna un valore 0 soltanto quando la card entra in modalità SPI correttamente. Altrimenti si realizza un ciclo

infinito che blocca l'avvio dell'intera scheda. In questo modo si evita l'inizio delle operazioni di campionamento se il dispositivo di memorizzazione non è regolarmente inserito nello slot o non risponde ai comandi in maniera appropriata.

LISTATO 4

```
TMROL = TICK_COUNTER_LOW;
TMROH = TICK_COUNTER_HIGH;
TOCON = 0b00000000 | TIMER_PRESCALE;
TOCON_TMROON = 1;
INTCON_TMROIF = 1;
INTCON_TMROIE = 1;
```

Notate che nel progetto abbiamo utilizzato una SD da 64Mb che gestiamo come un'enorme matrice, senza creare in essa strutture di tipo FAT16; così facendo, se la card viene inserita in un lettore per PC non è possibile leggerne i dati, a meno di non accedere ai singoli settori. Inizializzata la card, azzeriamo il contatore dei valori memorizzati e assegnamo il primo settore di scrittura. Tali variabili sono necessarie per stabilire sia il termine della transazione di scrittura per

di volta in volta. Se consideriamo che il campionamento avviene con una frequenza di un rilievo al secondo e che il valore corrispondente è lungo 16 bit, ciascun settore conserva i dati relativi all'andamento della temperatura per circa 4

LISTATO 5

```
SSPSTAT &= 0x3F;
SSPCON1 = 0x00;
SSPCON2 = 0x00;
SSPCON1 |= MASTER;
SSPSTAT |= SLEW_ON;
TRISC_RC3 = 1;
TRISC_RC4 = 1;
SSPCON1 |= SSPENB;
SSPAD = baud;
```

La variabile baud corrisponde al parametro passato dalla XEEInit. In questo caso viene effettuato un calcolo del baud rate sulla base della definizione seguente:
#define EE_BAUD(CLOCK, BAUD) (((CLOCK / BAUD) / 4) - 1)
Il clock utilizzato per la linea SCL viene sincronizzato sulla base di SSPADD. Per maggiori chiarimenti si vedano i datasheet del PIC18F458 alla sezione MSSP.

ciascun settore scritto su SD, sia per determinare il momento in cui bisogna avviare la sessione FTP per l'upload verso il server. In questo progetto utilizziamo la scrittura per singoli blocchi da 512 byte, pertanto è necessario aggiornare l'indirizzo del settore

minuti. La procedura continua richiamando la *OWReset()* che, come abbiamo visto nel precedente paragrafo, permette di stabilire se la sonda termometrica risulta correttamente collegata alla scheda ed è funzionante. Anche in questo caso abbiamo preferito bloccare la scheda nel caso in cui qualcosa non vada per il verso giusto; infatti il ciclo "while" non terminerà mai se non dopo aver ricevuto il bit di presenza dalla sonda. Dopo il reset abbiamo introdotto una pausa di stabilizzazione, per evitare un accesso troppo immediato ai registri della sonda, che comporterebbe la lettura del valore di power-up +85°C. Azzeriamo quindi il registro contatore per il TIMER0 ed abili-

tiamo gli interrupt del nostro PIC.

Si tenga ben presente che lo stack TCP/IP fa uso di una temporizzazione precisa dettata dal TIMER0; in particolare, il codice di gestione dell'interrupt non fa altro che richiamare una funzione chiamata *TickUpdate()* come si vede dal listato 3. Questa funzione gestisce proprio l'aggiornamento di alcuni contatori interni utilizzati per calcolare lo scorrere del tempo. Ciò è molto importante per realizzare una vera cooperazione

LISTATO 6

```
typedef struct _APP_CONFIG
{
    IP_ADDR      MyIPAddr;           -----> INDIRIZZO IP
    MAC_ADDR     MyMACAddr;          -----> INDIRIZZO MAC
    IP_ADDR      MyMask;             -----> MASCHERA DI SOTTORETE
    IP_ADDR      MyGateway;          -----> GATEWAY
    struct
    {
        unsigned int bIsDHCPEnabled : 1; -----> FLAG PER INDIRIZZAMENTO DINAMICO
    } Flags;
} APP_CONFIG;
```

dalla libreria *XEEPROM.c* che viene inclusa nello stack TCP/IP fornito da Microchip e sostanzialmente si differenzia da quella presente nelle librerie standard del compilatore C18 per il fatto che permette di utilizzare l'indirizza-

zione del modulo MSSP. Quest'ultimo elemento ha un nome (MSSP Address Register) che potrebbe essere fuorviante: ciò perché nel momento in cui il registro MSSP viene configurato come SLAVE, contiene l'indirizzo del dispositivo. Nel nostro caso, invece, il modulo funziona da MASTER e i 7 bit meno significativi di tale registro contengono il valore che viene caricato nel BRG (Baud Rate Generator). Quest'ultimo componente è un contatore che si decrementa due volte per ogni ciclo di clock del sistema (TCY) e viene utilizzato per sincronizzare la linea SCL. Naturalmente, viene ricaricato in maniera automatica proprio con il valore conservato nel registro *SSPADD*.

Per maggiori chiarimenti sulla materia vi suggeriamo di dare un'occhiata ai datasheet del PIC18F458, nella sezione *MSSP*. Arriviamo, quindi, alla *PGMInit()* una funzione che ha lo scopo di valorizzare i parametri della nostra applicazione; così facendo abbiamo voluto concentrare in un unico punto il caricamento dei parametri ➤

LISTATO 7

```
typedef struct _MAC_ADDR
{
    BYTE v[6];
} MAC_ADDR;
typedef union _IP_ADDR
{
    BYTE      v[4];
    DWORD     Val;
} IP_ADDR;
```

tra i vari livelli ISO/OSI di cui si compone lo stack. In particolare, ciascun secondo viene suddiviso in più "tick" a seconda della frequenza dell'oscillatore utilizzato. Dopo l'inizializzazione della scheda utilizziamo la *TickInit()* proprio per avviare tale elaborazione. È un passaggio obbligato per far funzionare bene lo stack (vedi listato 4).

Il TIMER0 viene inizializzato con i relativi contatori, poi si configura il prescaler in maniera da stabilire che la frequenza sorgente venga suddivisa per 256. Il TIMER0 viene avviato attivando il relativo segnale di interrupt. La *XEEInit* è una funzione che permette di avviare la comunicazione con la 24LC256 di cui è dotata la scheda. L'abbiamo estratta

mento a 16 bit. L'inizializzazione riguarda il modulo MSSP (Master Synchronous Serial Port) di cui è dotato il PIC18F458. Esso viene configurato come I²C Master precisando un baud-rate compatibile con le memorie a 400 kHz. Le linee utilizzate per il clock (RC3) e per i dati (RC4) vengono configurate in input. Le relative istruzioni sono presentate nel Listato 5.

Si faccia attenzione che il registro *SSPADD* ha una duplice funzione a seconda della modalità di inizializ-

LISTATO 8

```
BYTE c;
BYTE *p;

AppConfig.Flags.bIsDHCPEnabled = FALSE;
p = (BYTE*)&AppConfig;
XEEBeginRead(EEPROM_CONTROL, 0x01);
for ( c = 0; c < sizeof(AppConfig); c++ )
    *p++ = XEERead();
XEEEndRead();
```

LISTATO 9

```
typedef enum _SM_FTP
{
    SM_ARPID_CMD,          ----->Invia pacchetto ARP al server
    SM_ARPID_REP,          ----->Riceve pacchetto ARP con indirizzo MAC del server
    SM_CONNETTI_CMD,       ----->Apri Socket su porta comandi (21)
    SM_CONNESSO_U,         ----->Fase comando USER
    SM_CONNESSO_P,         ----->Fase comando PASS
    SM_CONNESSO_PP,        ----->Fase comando PORT
    SM_CONNESSO_S,         ----->Fase comando STOR
    SM_CONNETTI_DAT,       ----->Apri Socket su porta dati (1012)
    SM_ATTESA_CMD,         ----->Attesa apertura Socket porta comandi
    SM_ATTESA_DAT,         ----->Attesa apertura Socket porta dati
    SM_TRASMETTI,          ----->Trasmissione dati
    SM_CHIUDI,             ----->Chiusura Socket dati e comandi
    SM_FINE,               ----->Fine sessione FTP
} SM_FTP;
```

di rete fondamentali per l'interfacciamento tramite il chip ethernet l'RTL8019. Si faccia attenzione che consideriamo come primari l'indirizzo IP, l'indirizzo MAC, la maschera di sottorete e l'indirizzo del gateway. Vengono, invece, precisati come secondari l'indirizzo del Server FTP, la porta dati e le credenziali di accesso.

Dividiamo, in pratica, i parametri che caratterizzano il nostro dispositivo da quelli utilizzati per gestire una sessione di comunicazione. Questa differenza può apparire sottile ai più, ma rappresenta una prima separazione tra la logica di funzionamento e quella di sicurezza, ossia, se preferite, tra i parametri caratteristici del client e quelli usati per connettersi, tramite la rete, al server FTP.

Nel nostro caso utilizziamo una connessione non cifrata, quindi, potenzialmente insicura. Entrambi gli insiemi di informazioni vengono conservati nella EEPROM; tuttavia, mentre quelli primari sono caricati all'avvio, gli altri saranno letti soltanto quando risulterà necessario. La separazione logica ci ritornerà utile quando cominceremo a parlare di chiavi di cifratura e di interfacce riconfigurabili dinamicamente attraverso l'uso di Smart-Card. Tutti i dati primari per il funzionamento dell'interfaccia di rete vengono conservati nella struttura dati del Listato 6.

Si noti che ciascuno dei campi corrispondenti ha un tipo ridefinito secondo le dichiarazioni descritte nel Listato 7.

Il MAC è in pratica definito come un vettore di 6 byte, mentre per l'indirizzo IP si definiscono sia la distinzione in singoli byte che il suo valore complessivo a 32 bit. Il flag per l'indirizzamento dinamico consente, all'occorrenza, la possibilità per la periferica di rete di acquisire la propria configurazione IP da un server DHCP; a riguardo notate che, sebbene lo stack TCP/IP Microchip preveda questa possibilità (IP dinamico, ovvero assegnazione dell'IP address da parte del server) noi non la utilizziamo perché preferiamo un indirizzamento di tipo statico.

Se teniamo presente questa struttura, è molto semplice capire la sequenza di codice necessaria ad inizializzare l'applicativo (Listato 8). Per prima cosa disabilitiamo l'indirizzamento dinamico e leggiamo il primo byte della EEPROM. La libreria *XEEPROM.c* prevede la possibilità di gestire le transazioni con la memoria in questione attraverso letture e scritture sequenziali senza la necessità di gestire l'aggiornamento dell'indirizzo al quale si accede. Il file "bin" inserito nella memoria ha un'etichetta iniziale pari a 0x55 e contiene in sequenza tutti i dati primari. Il caricamento avviene molto semplicemente attra-

verso il puntatore *p* che indirizza la struttura dati vista nel precedente riquadro. Per il limite dimensionale della stessa si utilizza la *sizeof* direttamente nell'intestazione del ciclo *for*, in maniera da evitare la definizione di un'ulteriore variabile temporanea. Se tutto funziona correttamente siamo pronti ad avviare lo

stack TCP/IP, l'inizializzazione del quale avviene attraverso la funzione *StackInit()* che richiama le funzioni di avvio di ciascuno dei componenti inseriti nei vari livelli ISO/OSI.

Lasciamo da parte l'analisi di questa parte di codice, che occuperebbe da sola almeno tre numeri della rivista e passiamo, invece, all'inizializzazione del modulo che si occuperà della gestione del protocollo FTP. È importante ricordare

LISTATO 10

```
smFTP = SM_ARPID_CMD;
timeout = 0;
nrbyte = 0;
nrsettori = 0;
scarico = 1;
```

che questo protocollo fa parte del livello *Application* e che nella struttura dello stack deve interfacciarsi con il sottostante livello *Transport*. Si noti che, analogamente a come abbiamo sviluppato questo protocollo, potremo realizzare qualsiasi altro protocollo al livello più elevato dello stack. Per integrare in maniera corretta il relativo codice con quello sviluppato per la gestione dello stack, bisogna strutturarli come una macchina a stati finiti: in pratica, ogni volta che il controllo di esecuzione passerà a questo modulo verranno eseguite delle istruzioni specifiche a seconda dello stato in cui si trova la macchina virtuale. Per rendere le cose più

semplici, abbiamo identificato 13 possibili stati, che permettono di precisare ciascuna fase della sessione FTP. La relativa enumerazione è visibile nel Listato 9, nel quale si possono chiaramente identificare le singole sequenze viste nella precedente puntata: per prima cosa il nostro dispositivo invia un pacchetto ARP allo scopo di ricevere l'indirizzo MAC del server; poi sulla porta 21 viene aperto un socket per il trasferimento di comandi e risposte tra server e client. Notate che sono stati previsti uno stato di attesa della corretta apertura di tale socket, e la gestione di un eventuale time-out (*SM_ATTESA_CMD*).

Successivamente il client effettua la procedura di login, trasferendo le credenziali di accesso attraverso i comandi *USER* e *PASS*. Nel momento in cui il client è connesso, invia un comando *PORT* che permette di creare un canale di comunicazione per il trasferimento dati tramite una porta (evitiamo l'utilizzo di quella predefinita, ossia la 20) determinata dal client. Anche in questo caso si prevede un opportuno stato di attesa fino alla creazione del relativo socket.

A questo punto viene inviata al server l'istruzione *STOR*, che avvia il trasferimento dei dati, terminato il quale si trasmette il comando *QUIT*; vengono chiuse le due connessioni dati e

comandi e si inizializza nuovamente lo stato di trasporto per prepararlo alla nuova sessione. L'avvio del modulo corrispondente è relativo

funzione simile e complementare, tuttavia mentre i contatori sono relativi alle transazioni di lettura, esse calcolano i record e i settori

LISTATO 11

```
union
{
    unsigned short Val;
    struct
    {
        unsigned char LSB; /*Last significant byte*/
        unsigned char MSB; /*Most significant byte*/
    } bytes;
} tempera; /*Temperatura divisa in due campi*/
```

vamente semplice: non si fa altro che precisare lo stato iniziale della macchina virtuale ed azzerare alcuni contatori (vedere il Listato 10). Utilizziamo la variabile *timeout* per calcolare l'intervallo di tempo delle fasi di attesa durante la sessione FTP; come si vede, in questa funzione troviamo una coppia di variabili che ricorda molto i contatori *CONTAS* e *CONTAR* del main. In effetti tali variabili svolgono una

letti dalla card durante la fase di lettura e scaricamento.

Siamo quasi al termine della fase di inizializzazione; dobbiamo soltanto fare in modo che la SD-Card sia pronta ad essere riempita di dati, pertanto avviamo una transazione di scrittura attraverso la *SDBeginWrite*, passando come parametro il valore della variabile *CONTAS* che contiene il settore iniziale. Un'ultima considerazione va ➤

LISTATO 12

```
while(1)
{
    StackTask();

    ----- > SESSIONE FTP

    fine = TickGet();
    if (TickGetDiff(fine, inizio) >= 10)
    {
        inizio=fine;
        OWRReset();
        OWTX(0xCC);
        OWTX(0x44);
        while (OWRX1());
        OWRReset();
        OWTX(0xCC);
        OWTX(0xBE);
        tempera.bytes.LSB = OWRX();
        tempera.bytes.MSB = OWRX();
        for (CONTAG=1;CONTAG<=7;CONTAG++)
            OWRX();
        SDWrite(tempera.bytes.MSB);
        SDWrite(tempera.bytes.LSB);
        CONTAR = CONTAR + 2;
        if (CONTAR >= 512)
        {
            SDEndWrite();
            CONTAR=0;
            CONTAS++;
            if (CONTAS<=84)
                SDBeginWrite(CONTAS);
        }
    }
}
```

La temperatura viene rilevata attraverso la sequenza evidenziata. Il valore a 16 bit viene trasferito alla variabile "tempera" e si saltano i 7 byte successivi.

Il valore a 16 bit viene scritto sulla SD-Card aggiornando il contatore dei record scritti.

Nel caso sia terminata la scrittura di un intero settore della card si finalizza l'operazione e si aggiorna il contatore dei settori scritti. Se non è stato superato il limite relativo alle 6 ore di campionamento viene avviata una nuova transazione.

EVR-500B
Registratore digitale+FM+MP3
512 MB-144 ore
USB
VOX

MATVI

Microspia VHF - Telefonica e Ambientale - Alimentazione dalla linea telefonica - bassa caduta !

COMPACT
Telefono cellulare semplificato con funzioni d'allarme e monitoraggio ambientale
Sensori: movimento, acustico, vibrazione, esterno
Tracciatura delle celle Dual-Band

PD-1
Rilevatore tascabile Cellulari e microspie
GSM, DCS, UMTS
1+3000 Mhz
Portata 5+10m
Sensibilità a 2 livelli,
Vibra-beep-led
Batteria ricaricabile

Bias S.p.A.
Strada del Lavoro, 33 47892 Gualdicciole
REPUBBLICA DI S. MARINO
Tel. 0549.999408. Fax 0549.999431
www.bias.sm

LISTATO 13

```
typedef enum _SD_RES
{
    SD_OK = 0,
    SD_ERR1 = 0x31,
    SD_ERR2 = 0x32
} SD_RES;
```

fatta per la funzione *TickGet()*: abbiamo visto che la temporizzazione delle operazioni svolte dallo stack TCP/IP viene realizzata attraverso il TIMER0 e il relativo segnale di interrupt; Microchip rende disponibile la funzione *TickGet()* per utilizzare gli stessi contatori in altri casi.

Ad esempio noi la sfruttiamo per calcolare la frequenza di campionamento della temperatura.

La variabili inizio e fine saranno utilizzate per definire l'intervallo di tempo intercorso tra due interrogazioni consecutive della sonda.

Nella nostra procedura di inizializzazione non facciamo altro che valorizzare l'estremo iniziale dell'intervallo; ad ogni ciclo si determinerà l'estremo finale e dalla differenza dei due si potrà valutare se l'intervallo di tempo sarà stato superato oppure no.

tick, sincronizzando la frequenza di campionamento attorno al secondo. Il calcolo viene fatto utilizzando un'altra funzione messa a disposizione dallo stack "TickGetDiff" che vuole in ingresso due parametri relativi al valore dei contatori in due momenti diversi dell'elaborazione. Se la differenza supera il numero di "tick" necessari, l'estremo finale diventa quello iniziale del prossimo intervallo di tempo da calcolare. A questo punto viene effettuato il reset della sonda e attraverso la funzione OWTX() viene trasmesso il comando di lettura della temperatura (CCh - 44h). Durante la conversione il firmware attende di ricevere un bit a 1 che segnala la presenza del valore campionato nei registri della sonda. Dopo l'ulteriore reset si invia il comando che permette di leggere la relativa sequenza di 9 byte. I primi due vengono salvati nella variabile *tempera* la cui struttura è visibile nel Listato 11.

La suddivisione nei singoli byte che compongono la variabile permette di indirizzare i bit più significativi e quelli meno significativi in maniera più agevole, visto che l'accesso al

LISTATO 14

```
...
while(SDInit());
...
```

Questo ciclo continua finchè la card non viene inizializzata correttamente. La presenza di errori fa sì che risulti vera l'espressione logica che comanda il ciclo.

Detto questo, passiamo alla fase di campionamento vera e propria.

Campionamento

Dopo aver inizializzato le variabili necessarie ed aver predisposto l'ambiente hardware, il firmware entra in un ciclo infinito che contiene due procedure fondamentali: il campionamento e la sessione FTP. La prima viene eseguita nel momento in cui la differenza tra gli estremi inizio e fine supera i 10

valore registrato nella sonda avviene byte a byte. Dopo aver saltato i 7 byte rimanenti bisogna caricare il valore a 16 bit nella card: ciò si può fare agevolmente attraverso una funzione che abbiamo predisposto (SDWrite) funzione che prende in ingresso il byte da scrivere.

Dopo ogni scaricamento è necessario aggiornare il contatore relativo ai record trasferiti su SD-Card. Nel caso in cui tale numero raggiunga il limite di 512 significa che è stato

LISTATO 15

```
SD_RES SDInit(void)
{
    TRISBbits.TRISB7 = 0;
    PORTBbits.RB7 = 1;           //SS=1
    for (CONTA=0; CONTA<10; CONTA++)
    {
        SHOUT(0xFF);
    }
    PORTBbits.RB7 = 0;           //SS=0
    Delay1KTCYx(250);

    SHOUT(0x40);
    SHOUT(0x00);
    SHOUT(0x00);
    SHOUT(0x00);
    SHOUT(0x00);
    SHOUT(0x95);

    RISPI=SHIN();
    CONTA=0;
    while ( RISPI != 0x01 && CONTA <= 50000 )
    {
        RISPI=SHIN();
        CONTA++;
    }
    if (CONTA > 50000)
        return SD_ERR1;

    PORTBbits.RB7 = 1;           //SS=1
    Delay1KTCYx(250);
    PORTBbits.RB7 = 0;           //SS=0

    CONTA=0;
    RISPI=0x01;
    while ( RISPI != 0x00 && CONTA <= 50000 )
    {
        PORTBbits.RB7 = 1;           //SS=1
        SHOUT(0xFF);
        RISPI=SHIN();
        PORTBbits.RB7 = 0;           //SS=0
        Delay1KTCYx(250);
        SHOUT(0x41);
        SHOUT(0x00);
        SHOUT(0x00);
        SHOUT(0x00);
        SHOUT(0x00);
        SHOUT(0xFF);
        SHOUT(0xFF);
        RISPI=SHIN();
        CONTA++;
    }
    if (CONTA > 50000)
        return SD_ERR1;
    else
        return SD_OK;
}
```

"Dummy Clock" : vengono inviati 80 impulsi di clock per iniziare la comunicazione con il controller della SD.

"CMD0" : viene inviato il comando 0 (GO_IDLE_STATE) mantenendo la linea di selezione a livello logico basso. Si attende la risposta a 1 gestendo l'eventuale timeout.

"CMD1" : viene ripetutamente inviato il comando 1 (SEND_OP_COND) finché la risposta è 0 gestendo l'eventuale timeout. Da questo punto in poi la card è in modalità SPI e attende i comandi di lettura e scrittura.

scritto un settore completo e quindi bisogna finalizzare l'operazione. Lo si fa attraverso la funzione *SDEndWrite*.

Il contatore dei record viene azzerato; quello che conserva il numero di settori scritti è invece incrementato. Abbiamo predisposto l'upload via FTP ogni 6 ore, pertanto con un facile calcolo si ricava che il numero di settori da scrivere per ogni sessione è pari a 85. Durante ogni connessione trasferiremo dal client

al server file delle dimensioni di circa 43 kbyte. Naturalmente questo è solo uno dei possibili metodi di funzionamento, visto che modificando la frequenza di campionamento e il numero di settori da conservare si può variare la grandezza dei file inseriti nel disco fisso del server. Nel caso in cui non sia stato raggiunto il limite di settori per avviare la sessione FTP, è necessario iniziare una nuova transazione di scrittura attraverso la

"SDBeginWrite" alla quale passiamo il numero del settore da registrare. Si faccia attenzione che, a differenza di altre procedure sviluppate in progetti precedenti, qui non trasferiamo l'indirizzo a 32 bit da utilizzare nella transazione di scrittura, ma esattamente il numero del settore. Sarà poi la funzione richiamata a calcolare l'indirizzo di destinazione. Il codice complessivo relativo all'operazione è visibile nel Listato 12.

Diamo ancora un'occhiata alle funzioni che permettono la lettura/scrittura dei dati sulla card. Per rendere lo sviluppo sufficientemente semplice da riutilizzare, abbiamo raccolto tutte le funzioni relative alle operazioni di accesso alla SD-Card in un'unico file chiamato *SD.c*. Le dichiarazioni corrispondenti sono invece state inserite nel file *SD.h*. Le funzioni in questione si appoggiano a loro volta a

due piccoli moduli scritti in assembler chiamati *SHIN.asm* e *SHOUT.asm*. Si tratta della riscrittura ottimizzata delle istruzioni *SHIFTIN* e *SHIFTOUT* del PICBasic, che sicuramente conoscerete.

L'accesso alla SD-Card avviene gestendo lo stato dell'operazione, per cui nel caso la procedura comporti un errore di tipo 1 o 2 la funzione ritorna un apposito valore intero, secondo la definizione

Tabella 2

Funzione	Parametro Input	Parametro Output	Descrizione
SDInit	Nessuno	Stato Operazione	Inizializza la card portandola in modalità comandi SPI. Ritorna un valore a 0 se l'operazione termina correttamente.
SDBeginWrite	Settore Iniziale	Stato Operazione	Avvia una transazione di scrittura a partire dal settore indicato nel parametro d'ingresso. Ritorna un valore a 0 se l'operazione termina correttamente.
SDWrite	Byte da scrivere	Stato Operazione	Viene utilizzata solo dopo aver richiamato una SDBeginWrite terminata correttamente. Scrive sulla SD il dato passato in ingresso. Ritorna un valore a 0 se l'operazione termina correttamente.
SDEndWrite	Nessuno	Stato Operazione	La transazione di scrittura viene finalizzata. Questa operazione è necessaria dopo aver inviato 512 byte tramite la SDWrite. Viene inviato il CRC e verificato lo status della card dopo la scrittura. Ritorna un valore a 0 se l'operazione termina correttamente.
SDBeginRead	Settore Iniziale	Stato Operazione	Avvia una transazione di lettura a partire dal settore indicato nel parametro d'ingresso. Ritorna un valore a 0 se l'operazione termina correttamente.
SDRead	Nessuno	Byte Letto	Viene utilizzata solo dopo aver richiamato una SDBeginRead terminata correttamente. Legge la SD sulla base dell'indirizzo predisposto durante l'inizializzazione dell'operazione. Il dato letto viene passato attraverso il parametro di uscita.
SDEndRead	Nessuno	Stato Operazione	La transazione di lettura viene finalizzata. Questa operazione è necessaria dopo aver letto 512 byte tramite la SDRead. Anche in questo caso viene verificato lo status della card dopo la lettura.

descritta nel Listato 13. Allo scopo di poter utilizzare dei cicli *while* che terminano esclusivamente nel momento in cui l'operazione va a buon fine, abbiamo utilizzato il valore 0 per segnalare l'assenza di errori. Proprio come avviene nella procedura *SCHEDAInit()* esposta nel Listato 14. Vediamo nel concreto quali sono le funzioni utilizzabili per la lettura/scrittura a blocchi su SD-Card (Tabella 2).

Come si vede, le sequenze sono facilmente riutilizzabili in altri progetti che sfruttano le SD-Card come

dispositivi di memorizzazione. Tutti coloro che hanno seguito le puntate di approfondimento sulle memorie Flash troveranno in questo codice il naturale completamento del discorso che avevamo iniziato considerando il trasferimento dati con l'accesso a dei buffer esterni. Nel Listato 15 vediamo in dettaglio la funzione di inizializzazione della card. Lo sviluppo in C18 ci permette di apprezzare chiaramente le diverse fasi che abbiamo già incontrato in passato. Le modifiche necessarie a poter riutilizzare que-

ste sequenze in altri progetti riguardano esclusivamente i pin utilizzati come linee di comunicazione con la SD-Card. Bene, anche per questo mese siamo giunti al termine del nostro spazio. Nella puntata conclusiva di questo progetto affronteremo l'analisi dello sviluppo del protocollo FTP e finalmente vedremo come utilizzare il software per generare il file *.bin* da inserire nella EEPROM della nostra unità. Chiuderemo quindi il discorso proponendo un'applicazione pratica del progetto descritto.

Per il MATERIALE

I componenti utilizzati in questo progetto sono facilmente reperibili in commercio. Il software per la creazione del file binario da inserire nella EEPROM ed il firmware in formato hex possono essere scaricati gratuitamente dal sito della rivista (www.elettronica.in.it). Il microcontrollore è anche disponibile già programmato (cod. MF612) al prezzo di 25,00 Euro.

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

La rivista a portata di clic

a cura della *Redazione*

Una nuova veste e tante interessanti proposte per il rinnovato sito di Elettronica In.

Tra le iniziative più significative segnaliamo la disponibilità di tutti i nostri prodotti in formato digitale.

Ciò significa, ad esempio, poter scaricare e consultare, in qualsiasi momento del giorno e della notte, numeri arretrati, Corsi, Monografie, Software, ecc. E per gli abbonati appassionati di programmazione una gradita sorpresa!

Molti di voi se ne saranno già accorti, altri non ancora. Per questo motivo vogliamo segnalare da queste pagine le radicali modifiche che, mese dopo mese, stiamo apportando al sito della rivista (www.elettronica.in). Modifiche che riguardano non solo l'aspetto grafico ma anche (e soprattutto) i contenuti. Le mere pagine informative e di presentazione della nostra attività che caratterizzavano il vecchio sito stanno lasciando il posto ad una serie di servizi che, grazie ad Internet, risultano facilmente ed immediatamente fruibili. Vogliamo innanzitutto segnalare la possibilità per i let-

tori della rivista di scaricare gratuitamente quei contenuti che appesantirebbero inutilmente la rivista su carta; ci riferiamo in particolare ai master dei circuiti stampati, al firmware dei microcontrollori utilizzati nei progetti ed all'eventuale software. Questi disegni e listati, quando sono riportati su carta, costringono il lettore a riprendere fotograficamente lo stampato per ricavarne il master oppure a trascrivere riga per riga i listati. Con Internet è tutto molto più semplice, veloce e sicuro. Oltretutto in questo modo si liberano pagine sulla rivista da destinare a informazioni sicuramente più interessan- ➤



2003

ti di un lungo listato. A proposito del firmware implementato nei microcontrollori, ci preme segnalare che è nostra intenzione mettere a disposizione di tutti i lettori i listati in formato HEX riservando ai soli abbonati i file sorgente. In questo modo chiunque potrà realizzare

valenza commerciale. Questi file sorgente fanno parte di progetti denominati Top Projects, disponibili a pagamento e scaricabili dal sito nella loro interezza. I Top Projects ci forniscono lo

l'apparecchiatura proposta sulle pagine della rivista mentre solamente gli abbonati potranno accedere al codice per approfondire lo studio del progetto ed apportare eventuali modifiche. Ci è sembrato doveroso premiare in questo modo coloro che seguono fedelmente la rivista da anni e che ci danno la loro fiducia incondizionata. Sempre a proposito del firmware implementato nei microcontrollori, da questa iniziativa restano esclusi alcuni progetti particolarmente "pesanti" che hanno impegnato i nostri laboratori per molto tempo e che hanno una

spunto per segnalare un'altra importante novità: la disponibilità di tutti i nostri prodotti in formato digitale, dai numeri arretrati delle riviste ai Corsi, dalle Monografie ai Libri. Ci preme segnalare in particolare la disponibilità di numerosi Corsi che raccolgono in un unico file quanto pubblicato in passato a puntate sulla rivista. Ciascun Corso, al termine della pubblicazione, viene raggruppato in un apposito fascicolo disponibile in formato digitale e scaricabile dal sito. Oltre alle varie puntate, i Corsi comprendono anche tutti gli elementi hardware (demo-board o altro) e software (programmi, firmware, ecc) riguardanti quello specifico Corso. Sicuramente molto interessanti sono anche le

TOP PROJECT

Mediamente una volta al mese sulle pagine della rivista vengono proposti dei progetti molto complessi che hanno impegnato i nostri laboratori per molto tempo e che hanno una importante valenza commerciale. Ciascun progetto - denominato TOP PROJECT - può essere acquistato nella sua interezza: il relativo file ZIP (scaricabile da sito) contiene infatti il master degli stampati ed il firmware completo (sia i file hex che i file sorgente). Quanti sono interessati all'argomento, con un costo tutto sommato molto contenuto, potranno acquisire in pochissimo tempo quelle conoscenze che altrimenti richiederebbero lunghi e approfonditi studi. Ci riferiamo, ovviamente, essenzialmente ai firmware implementati nel micro di cui viene fornito il sorgente commentato nei punti più significativi. L'utilizzo delle versioni digitali è consentito esclusivamente per uso privato. Sono espressamente vietate la pubblicazione, la diffusione tramite Internet o mediante qualsiasi altro mezzo dei files scaricati o di parte di essi. Tali fatti costituiscono violazione del copyright e verranno perseguiti a termini di legge.



Progetto completo Localizzatore GSM-GPS con Q2501

Localizzatore remoto alimentato a batteria dalle dimensioni particolarmente ridotte grazie all'impiego del modulo GSM/GPS della Wavecom. Viene spiegato anche come utilizzare Internet e le cartografie disponibili in rete per visualizzare la posizione del localizzatore all'interno di una mappa. L'aspetto....

Codice: TOP01 - File zip completo (11008 kb) € 600,00

[Dettagli](#)



Progetto completo Microspia GSM professionale

Un concentrato di tecnologia in pochi centimetri cubi. Indispensabile per l'ascolto ambientale all'interno di veicoli, può essere utilizzata anche in abitazioni ed uffici. Il dispositivo utilizza nella versione GSM un modulo GR47. Il file ZIP (scaricabile immediatamente dopo il pagamento) comprende tra....

Codice: TOP02 - File zip completo (10218 kb) € 600,00

[Dettagli](#)



Progetto completo registratore audio con SD-Card

L'aspetto più importante di questo progetto è rappresentato dal firmware implementato nel microcontrollore PIC: scritto in PICBasic, include le procedure per effettuare le operazioni di scrittura su SD-Card attraverso la sostituzione delle istruzioni SHIFTIN e SHIFTOUT, aumentando notevolmente la velocità....

Codice: TOP03 - File zip completo (3382 kb) € 150,00

[Dettagli](#)



Progetto completo registratore GPS su SD-Card

L'aspetto più interessante di questo progetto è senza dubbio il firmware implementato nel microcontrollore PIC: scritto in PICBasic, include le procedure per effettuare le operazioni di scrittura su SD-Card facendo uso delle istruzioni SHIFTIN/SHIFTOUT ed eliminando la necessità di buffer esterni....

Codice: TOP04 - File zip completo (1219 kb) € 100,00

[Dettagli](#)



Unità remota GSM audio/video con modulo GM862-PCS

Un progetto all'avanguardia che sfrutta il primo modulo GSM direttamente interfacciabile ad una telecamera. Nel progetto viene esaminato approfonditamente il funzionamento del modulo Teli GM862-PCS e viene illustrato un circuito che ne sfrutta tutte le capacità. Il cuore del dispositivo di controllo....

Codice: TOP05 - File zip completo (N/D) [Scarica la copia omaggio](#)

[Dettagli](#)



Progetto completo Localizzatore remoto GPS/GSM per PC o palmare

Localizzatore remoto per veicoli che utilizza le reti GPS e GSM composto da una unità remota da installare sull'automobile e da una stazione base. Il ricevitore, sfruttando i segnali emessi dai satelliti della costellazione GPS, determina la posizione del veicolo ed invia i relativi dati, mediante i....

Codice: TOP06 - File zip completo (5330 kb) € 60,00

[Dettagli](#)

MONOGRAFIE

Le monografie comprendono tutti i progetti e gli articoli teorici riguardanti uno specifico argomento. Queste raccolte sono disponibili in un unico fascicolo in formato PDF scaricabile on-line al quale sono allegati anche tutti gli elementi hardware e software riguardanti i progetti presentati nella monografia.

L'utilizzo delle versioni digitali è consentito esclusivamente per uso privato. Sono espressamente vietate la pubblicazione, la diffusione tramite Internet o mediante qualsiasi altro mezzo dei files scaricati o di parte di essi. Tali fatti costituiscono violazione del copyright e verranno perseguiti a termini di legge.



Raccolta articoli e progetti DHX

Teoria e pratica del protocollo DHX12 utilizzato negli impianti di illuminazione per controllare sia la luminosità delle lampade che i movimenti dei sistemi più sofisticati, dagli scanner ai dispositivi a testa mobile, dai laser alle macchine per il fumo. Una full-immersion per conoscere a fondo questo protocollo ed apprendere le routine firmware da implementare nei micro utilizzati in queste applicazioni.

Codice: DHX-MON - File zip completo (16400 kb) € 15,00

[Dettagli](#)



Raccolta articoli e progetti di applicazioni meteo

Una serie di progetti dedicati agli appassionati di meteorologia. Dai dispositivi per la pubblicazione in Internet di dati meteo, ai sistemi per l'invio di informazioni sul tempo tramite SMS, all'interfacciamento con centraline commerciali per l'attivazione di dispositivi di vario genere.

Codice: METEO-MON - File zip completo (3052 kb) € 18,00

[Dettagli](#)



Sistemi di controllo remoto GSM con cellulari Siemens

Cinque progetti che utilizzano un comune cellulare Siemens della famiglia 35 per realizzare dei dispositivi di controllo remoto. Oltre all'hardware viene presentato ed illustrato anche il firmware (hex e sorgenti) implementato nei microcontrollori che si interfacciano al cellulare. Con le informazioni contenute in questa monografia chiunque sarà in grado di realizzare dispositivi di questo tipo.

Monografia campione disponibile gratuitamente

Codice: SIE-MON - File zip completo (1934 kb) [Scarica la copia omaggio](#)

[Dettagli](#)

Le schermate riportate in queste pagine illustrano alcuni dei servizi implementati nel nuovo sito della rivista. In alto a sinistra la pagina degli arretrati con le copertine ed i dettagli di ciascun numero. I fascicoli possono essere acquistati sia nella veste tradizionale che nel formato digitale (disponibile per il download 24 ore su 24).

Monografie che comprendono tutti i progetti e gli articoli teorici riguardanti uno specifico argomento. Queste raccolte sono disponibili in un unico fascicolo in formato PDF, scaricabile on-line, al quale sono allegati anche tutti gli elementi hardware e software riguardanti i progetti presentati nella monografia. Ad esempio, quanti sono interessati ai progetti che utilizzano il protocollo DMX, potranno scaricare un file ZIP da 16,5 MB contenente un articolo teorico e ben cinque progetti pratici, dai dimmer ai mixer DMX completi di

PRODOTTI SCARICABILI GRATUITAMENTE

Per dare la possibilità ai nostri Clienti di valutare i prodotti in vendita, rendiamo disponibili gratuitamente (chiediamo solamente di registrarsi) alcuni dei file disponibili per il download, uno per categoria. Ricordiamo che anche questi prodotti, pur essendo forniti gratuitamente, sono destinati ad un uso personale e sono soggetti a copyright: ne è vietata la pubblicazione e la diffusione con qualsiasi mezzo (stampa, internet, TV ecc.). Le violazioni verranno perseguite a termini di legge.

FASCICOLI ARRETRATI IN FORMATO DIGITALE - ABBONAMENTO DIGITALE ESTERO

I fascicoli arretrati sono disponibili in formato PDF ad alta risoluzione e sono del tutto simili alla rivista cartacea, con tutti gli articoli, le rubriche, la pubblicità, ecc. I file hanno dimensioni comprese tra 4 e 10 Mb a seconda del fascicolo; con una connessione ADSL possono essere scaricati in 1-2 minuti. Ricordiamo che l'abbonamento digitale alla rivista (disponibile per il momento solo per l'estero) prevede la possibilità per l'abbonato di scaricare 10 numeri del tutto simili a quello disponibile in omaggio in questa sezione.

Ulteriori dettagli sulla rivista omaggio
Scarica gratis questa copia

CORSI

Ciascun Corso presentato in passato sulla rivista, al termine della pubblicazione, viene raggruppato in un apposito fascicolo disponibile in formato digitale e scaricabile dal sito. Oltre alle varie puntate, i Corsi comprendono anche tutti gli elementi hardware (circuiti, schede o altro) e software (programmi, firmware, ecc) riguardanti quel specifico Corso. Il file disponibile in omaggio raccoglie tutti gli articoli riguardanti il Corso di programmazione del modulo web-server SitePlayer.

Ulteriori dettagli sul corso omaggio
Scarica gratis questo corso

MONOGRAFIE

Le monografie comprendono tutti i progetti e gli articoli teorici riguardanti uno specifico argomento. Queste raccolte sono disponibili in un unico fascicolo in formato PDF scaricabile on-line al quale sono allegati anche tutti gli elementi hardware e software riguardanti i progetti presentati nella monografia. In questa monografia omaggio sono presenti cinque progetti di telecontrollo remoto GSM che utilizzano telefoni cellulari Siemens; nel file sono presenti gli articoli relativi, i master degli stampati ed il firmware (hex + sorgenti).

Ulteriori dettagli sulla monografia omaggio
Scarica gratis questa monografia

TOP PROJECT

Il Top Project è un progetto particolarmente complesso che ha impegnato i nostri laboratori per molto tempo e che ha una importante valenza commerciale. Quanti sono interessati all'argomento, con un costo tutto sommato molto contenuto, potranno acquisire in pochissimo tempo quelle conoscenze che altrimenti richiederebbero lunghi e approfonditi studi. Ci riferiamo, ovviamente, essenzialmente al firmware implementato nel micro di cui viene fornito il sorgente commentato nei punti più significativi. Il Top Project offerto in omaggio descrive un'unità remota GSM audio/video con modulo GM862-PCS.

Ulteriori dettagli sul Top Project omaggio
Scarica gratis questo Top Project

remota GSM audio/video con modulo GM862-PCS. La maggior parte dei prodotti digitali è già disponibile per il download; quelli mancanti lo saranno a breve: contiamo di ultimare questa sezione del sito entro qualche settimana. Le altre novità riguardano le News dai produttori di semiconduttori, l'elenco di tutte le Fiere dell'Elettronica con le informazioni più importanti, le offerte di lavoro ed i collegamenti ai siti amatoriali di elettronica. Ma questo non è che l'inizio: abbiamo in cantiere numerose altre iniziative che vedranno la luce nei prossimi mesi. A questo proposito siamo anche disponibili a recepire idee e suggerimenti da parte dei

CORSI

Ciascun Corso presentato in passato sulla rivista, al termine della pubblicazione, viene raggruppato in un apposito fascicolo disponibile in formato digitale e scaricabile dal sito. Oltre alle varie puntate, i Corsi comprendono anche tutti gli elementi hardware (circuiti, schede o altro) e software (programmi, firmware, ecc) riguardanti quel specifico Corso. Il file disponibile in omaggio raccoglie tutti gli articoli riguardanti il Corso di programmazione del modulo web-server SitePlayer.

Ulteriori dettagli sul corso omaggio
Scarica gratis questo corso

MONOGRAFIE

Le monografie comprendono tutti i progetti e gli articoli teorici riguardanti uno specifico argomento. Queste raccolte sono disponibili in un unico fascicolo in formato PDF scaricabile on-line al quale sono allegati anche tutti gli elementi hardware e software riguardanti i progetti presentati nella monografia. In questa monografia omaggio sono presenti cinque progetti di telecontrollo remoto GSM che utilizzano telefoni cellulari Siemens; nel file sono presenti gli articoli relativi, i master degli stampati ed il firmware (hex + sorgenti).

Ulteriori dettagli sulla monografia omaggio
Scarica gratis questa monografia

TOP PROJECT

Il Top Project è un progetto particolarmente complesso che ha impegnato i nostri laboratori per molto tempo e che ha una importante valenza commerciale. Quanti sono interessati all'argomento, con un costo tutto sommato molto contenuto, potranno acquisire in pochissimo tempo quelle conoscenze che altrimenti richiederebbero lunghi e approfonditi studi. Ci riferiamo, ovviamente, essenzialmente al firmware implementato nel micro di cui viene fornito il sorgente commentato nei punti più significativi. Il Top Project offerto in omaggio descrive un'unità remota GSM audio/video con modulo GM862-PCS.

Ulteriori dettagli sul Top Project omaggio
Scarica gratis questo Top Project

st'ultimo con i file HEX ed i file sorgente).

Tutte questi prodotti sono acquistabili on-line con pagamento mediante carta di credito e vengono resi disponibili per lo scarico al termine della procedura di pagamento. Per dare la possibilità ai nostri Clienti di valutare i prodotti disponibili, offriamo la possibilità di scaricarne gratuitamente alcuni:

- Rivista Elettronica In n. 98;
- Corso SitePlayer;
- Monografia comprendente numerosi progetti di Telecontrollo GSM realizzati con i cellulari Siemens S35;
- Top Project 05, riguardante l'unità

Scarga gratis i prodotti demo

- Corso di programmazione per SitePlayer;
- Monografia dedicata ai controlli remoti GSM realizzati con telefoni Siemens;
- Unità remota audio/video GSM con Telit GM862-PCS (Top05).

NEWS DALLE AZIENDE (cliccare sulla data per visualizzare la notizia)

12/01/2006 **National Semiconductor**
National Semiconductor presenta il più Piccolo Sottosistema Audio con Stadio di Uscita in Classe D

11/01/2006 **Hitachi**
Hitachi Media: l'ausiliario interno all'orecchio che consente una comunicazione chiara e comoda

11/01/2006 **Fujitsu Siemens**
Fujitsu Siemens Computers lancia i PC SCALEO con tecnologia Vii™ per la casa digitale

Seleziona | Ultimi 20

OPPORTUNITA' DI LAVORO

Il gruppo Futura Elettronica - di cui la rivista Elettronica In fa parte - è in forte espansione ed è alla ricerca di numerose figure professionali da inserire nel proprio organico. Siamo alla ricerca sia di persone da inserire direttamente nella struttura aziendale che di collaboratori esterni. Se sei interessato a lavorare o collaborare con noi vai all'apposita pagina "RICERCA DI PERSONALE".

tutti i particolari hardware, del software e del firmware relativo (que-

nostri lettori; cerchiamo anche chi, per passione e per amore dell'elettronica, possa collaborare con noi per rendere il sito della rivista sempre più interessante. E se poi volete collaborare anche con dei progetti, date un'occhiata in basso!

Sei un appassionato di elettronica? Ti piacerebbe collaborare con una delle più prestigiose riviste di elettronica applicata? Pensi di essere in grado di progettare e costruire qualcosa di originale e innovativo? Vorresti rendere partecipe delle tue conoscenze ed esperienze un pubblico molto vasto? Ritieni di poter esporre chiaramente e per iscritto i tuoi lavori?

In questo caso sappi che,

Elettronica In

La più prestigiosa rivista italiana di elettronica applicata

CERCA

TECNICI ed APPASSIONATI DI ELETTRONICA, programmazione, robotica e tecnologie alternative cui affidare - come collaboratori esterni - la progettazione (hardware e software) di dispositivi elettronici e la stesura di articoli e corsi didattici. Le proposte debbono essere assolutamente originali ed il livello tecnico adeguato agli standard della rivista.

Contattaci subito!!!

ElettronicaIn - Vispa Edizioni
Via Adige 11, 21013 Gallarate (VA)
fax 0331/ 778799 - redazione@elettronica.in



MARCHEFIERE

ENTE REGIONALE PER LE MANIFESTAZIONI FIERISTICHE

**Quartiere Fieristico
CIVITANOVA MARCHE (MC)**

8-9 aprile 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**

“1^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**

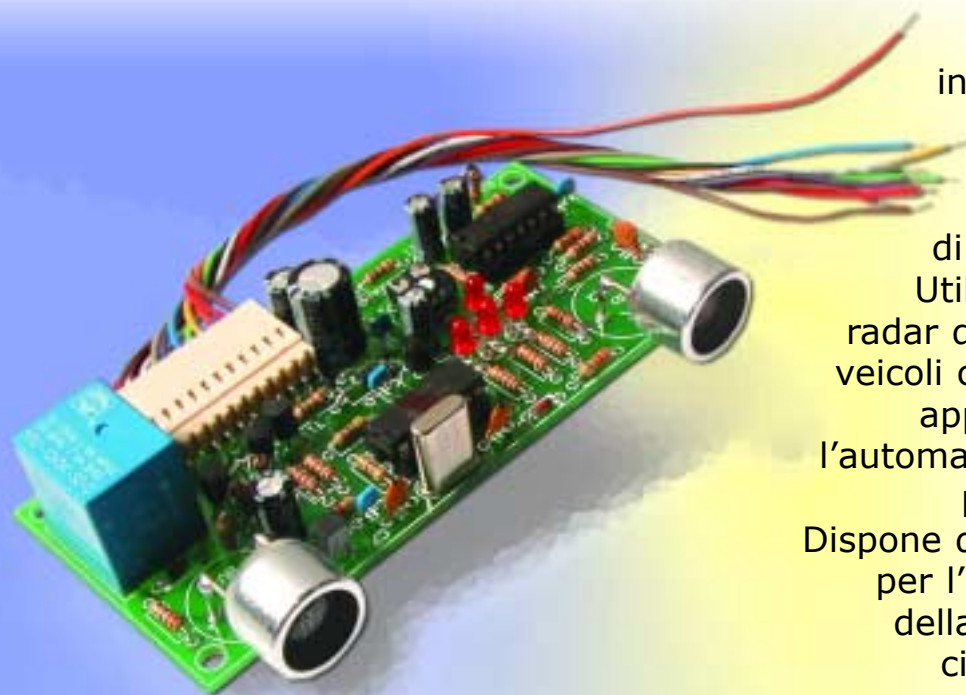
**Orario:
9,00-13,00
15,00-19,30**

**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

Sensore a ultrasuoni universale

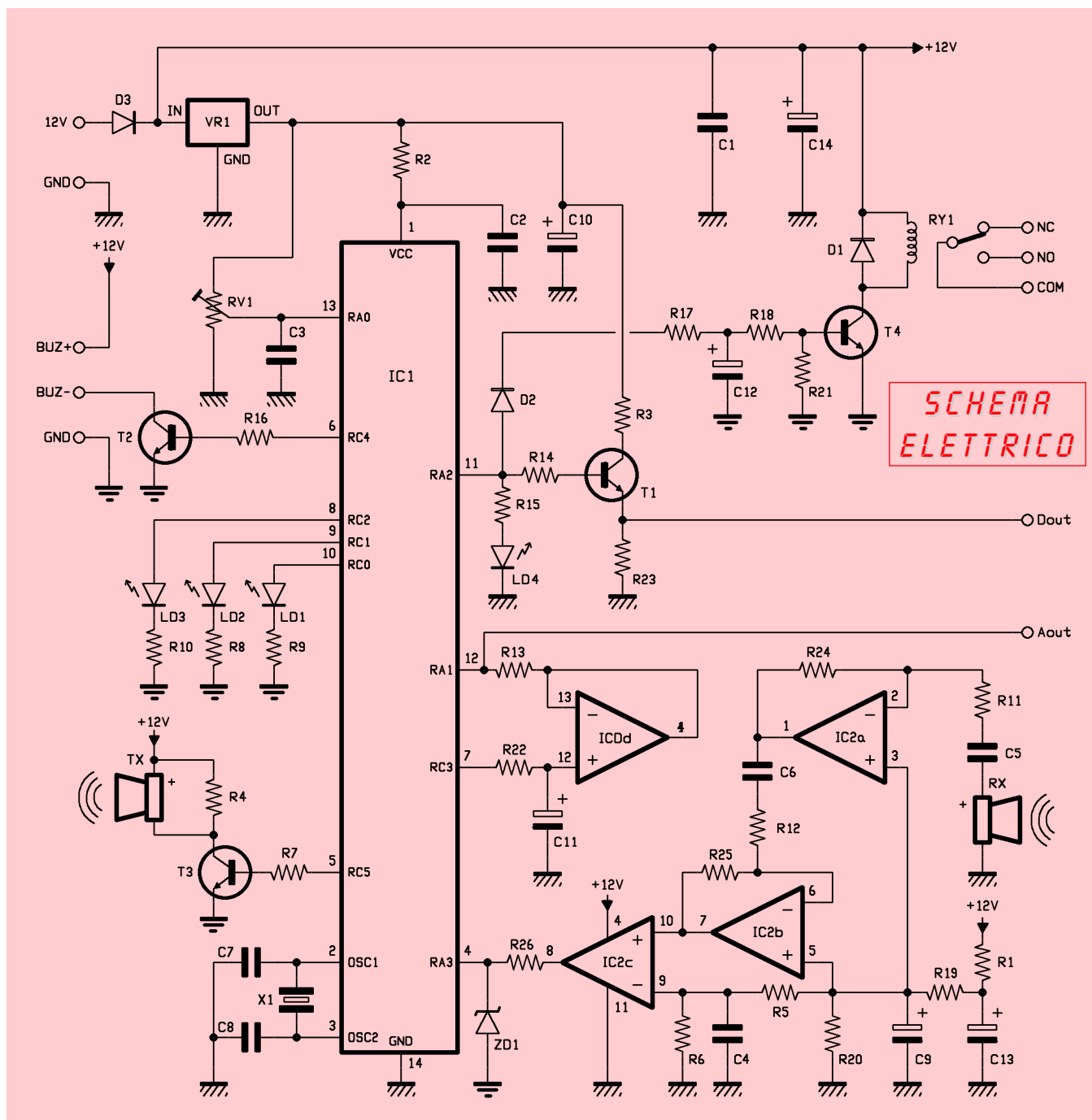
di *Gabriele Daghetta*



Unità di misura in grado di rilevare la presenza di persone e cose fino alla distanza di un paio di metri. Utilizzabile sia come radar di retromarcia per veicoli che per realizzare apparecchiature per l'automazione industriale, piccoli robot, ecc. Dispone di una barra a led per l'indicazione visiva della distanza e di un cicalino di allarme.

Il sensore ad ultrasuoni è una sorta di radar realizzato facendo emettere a una capsula ceramica una vibrazione sopra il campo dell'udibile (solitamente a 40 kHz) e rilevando con un trasduttore accordato alla medesima frequenza le onde riflesse dall'ambiente circostante; tale sistema viene utilizzato per differenti applicazioni in quanto è in grado sia di rilevare se un oggetto o una persona è entrata nel campo di applicazione del sistema (il segnale captato dal ricevitore subisce una brusca variazione di livello), sia - con apparecchiature più sofisticate - la distanza tra la sorgente e

l'oggetto riflettente in quanto l'ampiezza del segnale fornito dalla capsula ricevente è proporzionale alla distanza tra i due elementi. In queste pagine vogliamo proporvi la realizzazione di un progetto che sfrutta questo principio di funzionamento per realizzare più funzioni: ad esempio un sensore da collocare sul paraurti posteriore dell'auto per evitare di urtare quando si parcheggia in spazi ristretti oppure un misuratore di distanza (metro a ultrasuoni) stand-alone o collegato a un circuito di misura con A/D converter, ma anche un rilevatore di prossimità che permetta a un robot casalingo di ➤



vedere ed evitare gli ostacoli. Il circuito è sostanzialmente un radar a ultrasuoni assistito da microcontrollore che dispone di attuatori ed uscite con le quali consente la realizzazione delle applicazioni appena descritte; in particolare, quando rileva l'approssimarsi di un corpo fisso o in movimento scatta un relé, un transistor montato in configurazione open-collector e pilotato da un segnale rettangolare permette di far suonare un cicalino senza elet-

tronica o un piccolo altoparlante, e un led si illumina. Inoltre sono state previste un'uscita digitale TTL-compatible ed una analogica: la prima presenta una tensione continua, sempre quando il radar a ultrasuoni rileva l'approssimarsi di qualcuno o qualcosa; la seconda fornisce un potenziale strettamente correlato con la distanza del corpo rilevato. Completa il set di segnalazioni, una barra di tre diodi luminosi che indica la distanza stimata tra il

radar e l'oggetto rilevato. Le uscite funzionano come abbiamo appena spiegato, ed hanno ciascuna particolarità che vogliamo descrivere più dettagliatamente; riteniamo però sia il caso di vedere prima come funziona il sistema di rilevamento, quindi lasciamo da parte per un istante questo discorso e analizziamo il programma del microcontrollore e la routine che riguarda strettamente il radar a ultrasuoni. Il metodo usato dal nostro circuito

consiste nell'irradiare nell'ambiente una vibrazione a 40 kHz mediante un'apposita capsula ceramica accordata a tale frequenza, quindi nel rilevare le onde riflesse dagli oggetti circostanti; il rilevamento si effettua con un secondo trasduttore che, mentre il primo funziona da altoparlante, possiamo dire faccia da microfono. La sua membrana ceramica è sottoposta alla pressione (diciamo acustica, anche se i 40 kHz sono ben oltre il campo dell'udibile) dell'aria prodotta dagli ultrasuoni riflessi, che la raggiungono con intensità variabili in base al percorso compiuto e alla porosità e allo smorzamento dei materiali su cui l'onda emessa dal TX (trasmittente) è rimbalzata; ai capi del trasduttore ricevente (RX) rileviamo una tensione elettrica variabile prodotta dalla pressione ultrasonica sul

senso che la sua ampiezza e la sua frequenza sono e restano sempre le stesse mentre quando un oggetto si introduce nel campo di funzionamento ($20\text{ cm} \div 2,5\text{ m}$), la tensione varia. Se la si va a leggere, ad esempio raddrizzandola e rilevando la componente continua da essa ottenuta, è facile discriminare la condizione di riposo dall'intromissione di un corpo nel campo di copertura: infatti ai capi del raddrizzatore si verifica una variazione della tensione continua ottenuta.

Dopo l'inizializzazione delle linee di I/O, il programma che gira nel microcontrollore avvia una routine che simula il funzionamento del radar a ultrasuoni, ovvero che svolge quanto appena descritto; più esattamente, mediante un timer interno il PIC ricava una componente a 40 kHz che invia, mediante

zione è amplificare la tensione analogica ottenuta dalle onde rimbalzate nell'ambiente, quindi filtrarla e raddrizzarla per ottenere una componente continua. Più precisamente, il segnale elettrico prodotto dalla RX viene applicato (mediante il condensatore C5 e la resistenza R11) al piedino 2 dell'IC2a, un operazionale che lavora nella configurazione da amplificatore invertente e che guadagna in tensione $R24/R11$ volte; la componente amplificata viene nuovamente invertita ed elevata di livello dall'IC2b, il cui guadagno dipende dal rapporto $R25/R12$. Ne risulta una tensione che viene confrontata con un riferimento costante nell'IC2c, un terzo operazionale che nel nostro circuito funziona da comparatore non-invertente: ogni volta che il potenziale derivante dal segnale ricevuto supera la soglia determinata dal potenziale applicato al piedino 9, l'8 passa da zero a circa 12 volt.

Fermiamoci un istante per vedere la rete di polarizzazione degli operazionali IC2a, IC2b, IC2c, progettata per fornire a ciascuno un preciso riferimento; i primi due hanno l'input non-invertente polarizzato con poco meno di 6 volt (ricavati dal partitore R19/R20, alimentato a valle del filtro R/C R1/C13) ed è giusto così perché, amplificando segnali analogici, è conveniente portare, a riposo, la loro uscita a metà della tensione che li alimenta, così da garantire eguale escursione tanto nelle semionde positive, quanto in quelle negative. Dai 6 volt usati per IC2a e IC2b, il partitore R5/R6 ricava il potenziale che fa da riferimento per il comparatore; si tratta di una tensione poco più bassa (circa 5,8 V) di quella presente a riposo all'uscita dell'IC2b, cosa voluta per far sì che il comparatore commuti quando il picco del segnale risultante dalla capsula ricevente a ultrasuoni supera i 200 ➤

Specifiche tecniche

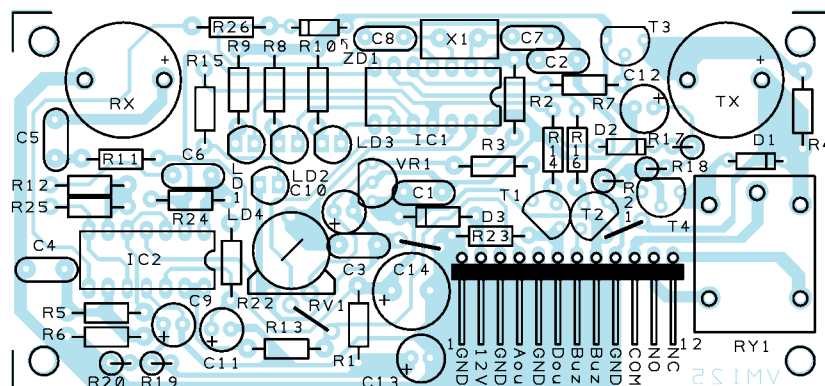
- Funzionamento a ultrasuoni a 40 kHz;
- Misura della distanza;
- Uscita a relé 1 A, 250 V;
- Uscita digitale 0/4 V;
- Uscita analogica $0 \div 5\text{ V}$;
- Rilevamento di oggetti tra 0,2 e 2,5 m;
- Barra a led per indicazione visiva della distanza;
- Cicalino per indicazione sonora della distanza.

materiale ceramico (per il fenomeno piezoelettrico il trasduttore della capsula genera una tensione proporzionale alla deformazione meccanica cui è sottoposto dalla pressione dell'aria) tensione la cui ampiezza e frequenza dipendono dalla quantità, intensità e tempo di arrivo delle varie componenti riflesse. In condizioni di riposo, ossia quando il radar è fermo in un ambiente in cui nulla si muove, la tensione si mantiene costante, nel

la linea RC5 (inizializzata come uscita) al transistor T3, un NPN che la amplifica in corrente per pilotare con essa la capsula piezo trasmittente. Contemporaneamente si predispone per il controllo ciclico della RA3 (impostata come ingresso) dalla quale legge le variazioni di tensione; notate che la capsula ricevente (RX) non viene interfacciata direttamente con il micro ma, piuttosto, il segnale da essa ricavato passa attraverso una rete la cui fun-

ELENCO COMPONENTI:

R1+R3: 47 ohm
 R4: 220 ohm
 R5: 10 kohm
 R6: 270 kohm
 R7+R18: 1 kohm
 R19+R23: 15 kohm
 R24+R26: 22 kohm
 RV1: Trimmer MO 10 kohm
 C1+C4: 100 nF multistrato
 C5, C6: 10000 pF ceramico
 C7, C8: 18 pF ceramico
 C9+C11: 10 µF 35 VL elettrolitico
 C12, C13: 100 µF 25 VL elettrolitico
 C14: 470 µF 25 VL elettrolitico
 D1, D2: 1N4148
 D3: 1N4007
 ZD1: Zener 5,1V 400mW
 IC1: PIC16F630
 IC2: TLV274
 LD1+LD4: led 3 mm rosso
 T1+T4: BC547
 VR1: 78L05
 X1: quarzo 8 MHz
 RY1: Relé 12Vdc/10A singolo scambio
 TX: capsula trasmittente ad ultrasuoni
 RX: capsula e ricevente ad ultrasuoni
 Varie:
 - Zoccolo 7+7 (2 pz.)
 - Strip maschio orizzontale 12 pin
 - Circuito stampato



Le due capsule ad ultrasuoni possono essere montate sul circuito stampato, sia in piedi che orizzontalmente (in tal caso vanno stagnate su due spezzoni di filo in rame rigido saldati a loro volta nelle rispettive piazzole) oppure al di fuori, collegate ciascuna con del cavetto schermato coassiale.

millivolt positivi. Ogni volta che il segnale in questione supera la soglia, il comparatore fornisce un impulso positivo, mentre quando scende al disotto dello zero fittizio costituito dai 6 volt che polarizzano IC2a e IC2b, il comparatore mantiene la propria uscita a livello basso (circa 0 V). Da tale modo di funzionamento possiamo dedurre che IC2c è essenzialmente un raddrizzatore a singola semionda, o, se preferite, un rivelatore: il suo scopo è rendere unidirezionale la tensione

variabile ricavata dalla capsula RX e ricavare impulsi rettangolari da far leggere al microcontrollore. Siccome l'ampiezza di tali impulsi è circa 12 volt e il PIC16F630 accetta sulle proprie linee di input non più di 5,5 volt, è stato necessario interporre il diodo ZD1, che (con l'aiuto della resistenza di limitazione della corrente R26) limita a 5,1 il potenziale applicato ad RA3. Dunque, siamo tornati a questa linea del PIC, che il software legge ciclicamente per verificare su di

essa la presenza degli impulsi dovuti al ritorno dell'onda a 40 kHz generata dall'apposita routine. Rilevato l'arrivo degli impulsi, una routine elabora i dati corrispondenti misurando il valor medio della tensione che essi compongono; più esattamente, il micro verifica la larghezza e la distanza degli impulsi, per determinare l'intensità del segnale riflesso verso la capsula RX. La misura coinvolge l'A/D converter interno al PIC16F630, che in fase di inizializzazione degli

I/O assegnamo alla linea RA3; il convertitore ha una risoluzione di 10 bit, può essere abbinato a un massimo di 8 I/O (letti in multiplexing) e consente di definire, mediante un potenziale applicato alla RA0, la tensione di riferimento del campionamento. Nel nostro caso, con il trimmer RV1 definiamo il range di tensione che l'A/D deve andare a convertire, quindi scegliamo la sensibilità della conversione: più si sposta il cursore verso il positivo dei 5 volt, meno sensibile è il circuito e viceversa. L'ampiezza della scala di riferimento del converter è direttamente proporzionale alla sensibilità, ovvero alla distanza di rilevamento del radar, quindi con il trimmer RV1 possiamo definire la distanza coperta dal sensore, scegliendo tra un minimo di 20 cm e un massimo di 2 metri e mezzo.

La conversione A/D determina dei dati numerici che vengono letti dal programma principale al fine di valutare la distanza e comandare di conseguenza le uscite di cui il circuito dispone; vediamo una per una come vengono gestite, iniziando dalla linea RA2, che viene forzata a livello logico alto quando l'oggetto rilevato si trova tra la minima e la massima distanza percettibile. In caso contrario (oggetto fuori portata o troppo vicino) RA2 si pone a zero logico. Quindi il led LD4 si illumina quando il radar rileva la prossimità di un oggetto entro il campo compreso tra 20 cm e una distanza che dipende dalla sensibilità decisa mediante il trimmer RV1. Quanto al relé, scatta poco dopo l'accensione dell'LD4 e ricade con leggero ritardo rispetto al suo spegnimento. Notate che il software ritiene superata la soglia minima di distanza quando rileva che il segnale captato dalla capsula RX e amplificato da IC2a e IC2b è appena al disotto del massimo livello; considera l'oggetto oltre la massima distanza quando, invece, in base all'impostazione dell'RV1 il segnale giunge sulla RA0 con un'ampiezza inferiore a quella della soglia minima stabilita. La linea RA2, responsabile del controllo di LD4 e RY1, pilota anche il transistor T4, un NPN usato come buffer per controllare l'output digitale: la linea Dout fornisce un livello logico che ne segue le vicende, ovvero lo zero quando RA2 si trova a livello basso e l'1 (circa 4 volt) quando invece assume lo stato alto.

Oltre all'uscita digitale, il software ne prevede una analogica, che fornisce un potenziale la cui ampiezza è direttamente proporzionale alla distanza alla quale si trova l'oggetto rilevato, ovviamente entro i soliti margini; Aout (questo è il nome dell'output analogico) determina una tensione ricavata da un'apposita routine che genera un'onda PWM il cui duty-cycle è direttamente proporzionale alla distanza rilevata, o, se preferite, inversamente proporzionale all'ampiezza della componente letta dall'A/D converter del micro. Gli impulsi uscenti da RC3 vengono filtrati dalla cella passa-basso ➤

27^a



MUNICIPIO
SCANDIANO
2006

MOSTRA REGIONALE DELL'ELETTRONICA

18-19 FEBBRAIO 2006

SCANDIANO (RE)



con il patrocinio di A.R.I. sez. Reggio Emilia

comune di scandiano

componentistica / computer / hi-fi car
telefonia / radiantismo CB e OM
videoregistrazione

mercato delle pulci radioamatoriali

CENTRO FIERISTICO SCANDIANO

Tel. 0522.857436 www.fierascandiano.it

orari: sabato 18 dalle 9.00 alle 18.30
domenica 19 dalle 9.00 alle 18.00

composta da R22 e C11; ai capi di quest'ultimo troviamo perciò una tensione continua e ben livellata, la cui ampiezza segue il duty-cycle dell'onda PWM e quindi è tanto maggiore quanto più debole (a causa della maggior distanza) è l'intensità del segnale letto dalla capsula ricevente e viceversa. Il potenziale viene applicato all'ingresso di un operazionale (IC2d) funzionante da buffer non-invertente, che lo restituisce con la medesima ampiezza dal proprio piedino 13, dal quale, mediante R13, raggiunge Aout e viene anche letto dalla RA1. Scopo del buffer è permettere di pilotare con l'Aout dispositivi che assorbono anche alcune decine di milliampere senza caricare l'RC3, linea che non potrebbe dare più di qualche mA.

E veniamo, infine, all'uscita prevista per il comando del buzzer: fa capo al transistor T2, un NPN la cui base viene pilotata dal microcontrollore mediante la propria linea RC4; è stata prevista essenzialmente per quando si usa il circuito da radar di parcheggio, dato che comanda un cicalino facendolo suonare in maniera diversa a seconda della distanza rilevata. Ecco come funziona: se il radar non rileva nulla o l'oggetto si trova oltre la distanza coperta (più lontano di 2,5 metri) la linea RC4 assume lo zero logico, il transistor è interdetto e il buzzer resta spento; quando invece tra le capsule del radar ed un even-

tuale oggetto vi sono meno di 2,5 metri, il microcontrollore genera la sua prima segnalazione di allarme facendo commutare la condizione logica della linea RC4 (tipicamente la cadenza è 0,5 secondi a livello alto ed altrettanti a zero volt) e mandando alternativamente in conduzione e in saturazione T2, il che determina l'emissione di un suono intermittente da parte del cicalino collegato ai punti BUZ+ e BUZ-. Infine, se il radar e l'oggetto rilevato si avvicinano troppo, scendendo a 20 cm o meno, RC4 viene posta fissa ad 1 logico, il transistor va costantemente in saturazione e il cicalino suona continuamente.

Oltre alle uscite succitate, il microcontrollore può comandare una scala di led che esprimono a loro modo la distanza rilevata; essi vengono comandati mediante i convertitori interni, ciascuno dei quali ha impostata una soglia differente da quella degli altri.

In parole povere, LD3 si accende quando l'oggetto rilevato è più lontano di 20 cm ma si trova a breve distanza, LD2 si illumina insieme ad LD3 quando la distanza cresce, tipicamente oltre il metro, mentre alla massima distanza coperta sono accesi tutti e tre i diodi LD1, LD2, LD3.

Sotto i 20 cm i tre led sono tutti spenti. Durante il funzionamento può capitare che uno o più led non assumano una condizione stabile: accade tipicamente quando il corpo

rilevato è in movimento o si trova a una distanza che non corrisponde ad alcuna delle soglie impostate.

Bene, ora che conoscete dettagliatamente il funzionamento di tutte le uscite potete sapere quali e quante utilizzarne per ognuna delle applicazioni cui intendete destinare il circuito. Volendo valutare l'interazione tra le varie forme di segnalazione, a titolo d'esempio sappiate che con il cursore del trimmer RV1 a metà corsa, Aout fornisce 0,8 V quando è acceso solo LD3, ovvero se il circuito rileva la prossimità di un oggetto collocato a una distanza di poco inferiore ai 50 centimetri.

Concludiamo la descrizione dello schema dicendo che l'intero circuito funziona a tensione continua di valore compreso tra 12 e 15 volt, applicata tra i punti 12 V e GND; il diodo D3 protegge dall'inversione di polarità ed assicura il passaggio di corrente solo dall'input di alimentazione al resto del circuito. Il regolatore VR1 è un 78L05 che ricava 5 volt esatti destinati a far funzionare il microcontrollore e ad alimentare il trimmer di regolazione della sensibilità e il transistor che fa da buffer dell'uscita digitale.

Realizzazione pratica

Detto ciò vediamo come costruire il modulo: per prima cosa occorre preparare la basetta stampata, ricorrendo alla fotoincisione e ricavando la pellicola dalla traccia scaricabile dal nostro sito Internet www.elettro-

Per il MATERIALE

Il progetto descritto in queste pagine è disponibile in scatola di montaggio (cod. VM125) al prezzo di 39,00 Euro. Il kit comprende tutti i componenti, la basetta forata e serigrafata, le minuterie, le capsule ad ultrasuoni ed il microcontrollore già programmato. Il sensore a ultrasuoni è anche disponibile già montato e collaudato allo stesso prezzo del kit. 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>**

nicain.it. Ottenuto il c.s., seguendo il disegno di montaggio inserite e saldate i componenti a partire dalle resistenze da disporre orizzontalmente e dai diodi (verificate la polarità) proseguendo con quelle da mettere in verticale R17÷R21, gli zoccoli per il micro e il TLV274, il trimmer e i condensatori (attenzione alla polarità degli elettrolitici) il quarzo, i transistor, il 78L05, i quattro led ed il relé.

Completate il montaggio inserendo le due capsule ad ultrasuoni: distinguete la trasmittente dalla ricevente perché la prima ha nella sigla o stampata sull'involucro le lettere S o T, mentre l'altra riporta una R; altri tipi di capsule si possono distinguere perché la trasmittente dietro (cioè da dove spuntano i terminali) non è schermata, mentre la ricevente (esposta alle interferenze) è tutta racchiusa in un involucro metallico. Per completare il circuito occorre realizzare tre piccoli ponticelli, usando degli spezzoni di terminali tagliati dalle resistenze: il primo prende posto fra T2 e T4, il secondo (J2) tra C14 e D3, mentre J3 si trova tra R13 ed RV1.

Le due capsule ad ultrasuoni possono essere montate sul circuito stampato, sia in piedi che orizzontalmente (in tal caso vanno stagnate su due spezzoni di filo in rame rigido saldati a loro volta nelle rispettive piazzole) oppure al di fuori, collegate ciascuna con del cavetto schermato; ricordate che, se sul suo corpo ha indicata una polarità, la trasmittente deve avere il polo positivo sulla piazzola siglata +.

Lo stesso dicasi per la ricevente,

nella quale normalmente il polo positivo si riconosce andando per esclusione: il negativo è quello elettricamente connesso all'involucro.

Terminate le saldature, si possono inserire gli integrati (il microcontrollore deve essere già programmato con l'apposito firmware).

Per agevolare le connessioni è stato previsto un connettore maschio a passo 2,54 mm, da 12 poli con terminali a 90°; potete decidere se montarlo, oppure collegare le uscite e l'alimentazione direttamente sulle piazzole.

A proposito di alimentazione, per il collaudo e il normale utilizzo ve ne serve uno che fornisca 12÷15 Vcc ed una corrente di almeno 100 milliampere; per l'uso in automobile (radar di parcheggio) vi basta collegare il punto +12 V ed il GND (se avete acquistato il kit di montaggio,

corrispondono, rispettivamente, ai fili rosso e bianco del connettore femmina volante) al positivo dell'impianto elettrico ed alla massa della vettura.

L'alimentazione va prelevata (interponendo un fusibile ritardato da 500 mA) dalla chiave del quadro, così da alimentare l'unità solo quando si usa l'auto: ciò evita che il suo assorbimento possa, in caso di sosta per diversi giorni, scaricare la batteria.

Alimentato il modulo, portate il cursore dell'RV1 a metà corsa e portatevi davanti alle capsule, verificando che, oltre i 20 cm e fino a qualche metro il sensore vi rilevi: ve ne accorgete perché il relé scatterà e insieme ad esso si illumineranno LD4 e, in base a dove vi troverete, uno, due o tutti quanti i led LD1, LD2, LD3.

USCITA	FUNZIONE	UTILIZZABILE...
Relé	Normalmente in conduzione tra COM ed NC, si chiude tra COM ed NO ogni volta che <i>Dout</i> si porta a livello logico alto e torna a riposo con un leggero ritardo rispetto al ritorno a zero volt della stessa <i>Dout</i> .	...in abbinamento o in alternativa alla <i>Dout</i> , come attuatore anti-intrusione negli impianti d'allarme o per avviare la riproduzione di un messaggio o l'apertura di un tornello all'approssimarsi di una persona o di un'automobile.
<i>Dout</i>	Normalmente a livello logico basso, assume lo stato alto (4 V) quando viene rilevata la prossimità di un corpo, entro la distanza coperta, ossia tra 0,2 e 2,5 metri.	...in abbinamento o in alternativa all'uscita a relé, la quale segue praticamente il suo andamento.
LD4	Segue le vicende della <i>Dout</i> : è acceso quando il radar rileva un corpo a una distanza di 0,2÷2,5 m; è spento in assenza di rilevamento o se l'oggetto si trova più vicino di 0,2 o più lontano di 2,5 metri.	...come segnalazione luminosa che ci permette di capire quando un oggetto è entrato nel raggio d'azione del radar; nell'uso da sensore per antifurto è un utile ripetitore dello stato dell'uscita.
<i>Aout</i>	Fornisce una tensione di valore direttamente proporzionale alla distanza dell'oggetto rilevato, che varia da 0 volt quando l'oggetto è distante non più di 20 cm a 4 V quando, invece, si trova ad almeno 2,5 metri.	...per pilotare un microamperometro a lancetta la cui scala può essere graduata in decimetri, oppure un voltmetro digitale o un circuito che possa visualizzare la tensione; lo scopo è realizzare un metro a ultrasuoni.
Cicalino (BUZ +/-)	Comanda un cicalino facendolo suonare in maniera pulsante se viene rilevato un oggetto a meno di 2,5 metri e continuamente se tra le capsule e l'oggetto stesso ci sono meno di 20 cm. Se la distanza supera i 2,5 metri il cicalino tace.	...quando il circuito viene montato su autoveicoli per coadiuvare il parcheggio: il suono pulsante avverte che ci si sta avvicinando a un muro o ad un altro veicolo, mentre quando si sente la nota fissa significa che la distanza è minima e conviene fermarsi.
LD1, LD2, LD3	Formano una barra che indica la distanza: la distanza per la quale si accendono dipende dalla regolazione del trimmer; LD3 indica la distanza minore, LD2, acceso insieme a LD3, quella intermedia, mentre quando il rilevamento avviene alla massima distanza o oltre tutti i tre diodi sono accesi. I diodi sono spenti sotto i 20 cm.	...come monitor della distanza; dà un'indicazione approssimativa che dipende dalla taratura, però in alcuni casi può essere utile: ad esempio, usando il circuito da radar per parcheggio offre una segnalazione visiva di massima della distanza, da accompagnare o usare in alternativa a quella del cicalino.

PS1503SB
**Alimentatore
0-15Vdc / 0-3A**

Uscita stabilizzata singola 0 - 15Vdc con corrente massima di 3A. Limitazione di corrente da 0 a 3A impostabile con continuità. Due display LCD con retroilluminazione indicano la tensione e la corrente erogata dall'alimentatore. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio; peso: 3,5 Kg.

PS1503SB € 62,00

PS3010
**Alimentatore
0-30Vdc/0-10A**

Alimentatore stabilizzato con uscita singola di 0 - 30Vdc e corrente massima di 10A. Limitazione di corrente da 0 a 10A impostabile con continuità. Due display indicano la tensione e la corrente erogata dall'alimentatore. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio; peso: 12 Kg.

PS3010 € 216,00

PS3020
**Alimentatore
0-30Vdc/0-20A**

Alimentatore stabilizzato con uscita singola di 0-30Vdc e corrente massima di 20A. Limitazione di corrente da 0 a 20A impostabile con continuità. Due display indicano la tensione e la corrente erogata dall'alimentatore. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio; peso: 17 Kg.

PS3020 € 330,00

PS230210
**Alimentatore
con uscita duale**

Alimentatore stabilizzato con uscita duale di 0-30Vdc per ramo con corrente massima di 10A. Ulteriore uscita stabilizzata a 5Vdc. Quattro display LCD indicano contemporaneamente la tensione e la corrente erogata da ciascuna sezione; possibilità di collegare in parallelo o in serie le due sezioni. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio; peso: 20 Kg.

PS230210 € 616,00

con tecnologia
SWITCHING

LA TECNOLOGIA SWITCHING
CONSENTE DI OTTENERE UNA
NOTEVOLE RIDUZIONE DEL
PESO ED UN ELEVATISSIMO
RENDIMENTO ENERGETICO
DELL'APPARECCHIATURA.

Alimentatore stabilizzato da laboratorio in tecnologia switching con indicazione delle funzioni mediante display multilinea. Tensione di uscita regolabile tra 0 e 20Vdc con corrente di uscita massima di 10A. Soglia di corrente regolabile tra 0 e 10A. Il grande display multifunzione consente di tenere sotto controllo contemporaneamente tutti i parametri operativi.

Caratteristiche: Tensione di uscita: 0-20Vdc; limitazione di corrente: 0-10A; ripple con carico nominale: inferiore a 15mV (rms); display: LCD multilinea con retroilluminazione; dimensioni: 275 x 135 x 300 mm; peso: 3 Kg.

PSS2010 € 265,00

PSS2010
**Alimentatore Switching
0-20Vdc/0-10A**

Alimentatori da Laboratorio

Alimentatore stabilizzato con uscita duale di 0-30Vdc per ramo con corrente massima di 3A. Ulteriore uscita stabilizzata a 5Vdc con corrente massima di 3A. Quattro display LCD indicano contemporaneamente la tensione e la corrente erogata da ciascuna sezione; limitazione di corrente 0÷3A impostabile indipendentemente per ciascuna uscita. Possibilità di collegare in parallelo o in serie le due sezioni. Peso: 11,6 Kg.

PS23023 € 252,00

PS23023
**Alimentatore
2x0-30V/0-3A 1x5V/3A**

Alimentatore stabilizzato con uscita singola di 0-30Vdc e corrente massima di 3A. Limitazione di corrente da 0 a 3A impostabile con continuità. Due display LCD indicano la tensione e la corrente erogata dall'alimentatore. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio. Peso: 4,9 Kg.

PS3003 € 125,00

PS3003
**Alimentatore
0-30Vdc/0-3A**

Alimentatore stabilizzato con uscita singola di 0-50Vdc e corrente massima di 5A. Limitazione di corrente da 0 a 5A impostabile con continuità. Due display indicano la tensione e la corrente erogata dall'alimentatore. Contenitore in acciaio, pannello frontale in plastica. Colore: bianco/grigio. Peso: 9,5 Kg.

PS5005 € 225,00

PS5005
**Alimentatore
0-50Vdc/0-5A**

Alimentatore da banco stabilizzato con tensione di uscita selezionabile a 3 - 4,5 - 6 - 7,5 - 9 - 12Vdc e selettore on/off. Bassissimo livello di ripple con LED di indicazione stato. Protezione contro corto circuiti e sovraccarichi. Peso: 1,35 Kg.

PS2122LE € 18,00

PS2122LE
**Alimentatore
da banco 1,5A**

Alimentatori a tensione fissa

PS1303
**Alimentatore
13,8Vdc/3A**

Alimentatore stabilizzato con uscita singola di 13,8 Vdc in grado di erogare una corrente massima di 3A (5A di picco). Il circuito di alimentazione a 220 Vac è protetto tramite fusibile mentre l'uscita dispone di protezione da cortocircuiti. Contenitore in acciaio. Colore: bianco/grigio; peso: 1,7 Kg.

PS1303 € 26,00

PS1310
**Alimentatore
13,8Vdc/10A**

Alimentatore stabilizzato con uscita singola di 13,8 Vdc in grado di erogare una corrente massima di 10A (12A di picco). Il circuito di alimentazione a 220 Vac è protetto tramite fusibile mentre l'uscita dispone di protezione da cortocircuiti. Contenitore in acciaio. Colore: bianco/grigio; peso: 4 Kg.

PS1310 € 43,00

PS1320
**Alimentatore
13,8Vdc/20A**

Alimentatore stabilizzato con uscita singola di 13,8 Vdc in grado di erogare una corrente massima di 20A (22A di picco). Il circuito di alimentazione a 220 Vac è protetto tramite fusibile mentre l'uscita dispone di protezione da cortocircuiti. Contenitore in acciaio. Colore: bianco/grigio; peso: 6,7 Kg.

PS1320 € 95,00

PS1330
**Alimentatore
13,8Vdc/30A**

Alimentatore stabilizzato con uscita singola di 13,8 Vdc in grado di erogare una corrente massima di 30A (32A di picco). Il circuito di alimentazione a 220 Vac è protetto tramite fusibile mentre l'uscita dispone di protezione da cortocircuiti. Contenitore in acciaio. Colore: bianco/grigio; peso: 9,3 Kg.

PS1330 € 140,00

PSS4005
**Alimentatore Switching
0-40Vdc/0-5A**

Alimentatore stabilizzato da laboratorio in tecnologia switching con indicazione delle funzioni mediante display multilinea. Tensione di uscita regolabile tra 0 e 40Vdc con corrente di uscita massima di 5A. Soglia di corrente regolabile tra 0 e 5A.

Caratteristiche: tensione di uscita: 0-40Vdc; limitazione di corrente: 0-5A; ripple con carico nominale: inferiore a 15 mV (rms); display: LCD multilinea con retroilluminazione; dimensioni: 275 x 135 x 300 mm; peso: 3 Kg.

PSS4005 € 265,00

Tutti i prezzi si intendono
IVA inclusa.

Il compilatore C18

a cura di Carlo Tauraso

3



Muoviamo i primi passi alla scoperta del C18 Microchip, ideale per la programmazione di sistemi complessi quali il CAN-Bus. In questa terza puntata analizziamo strutture, funzioni, parametri, puntatori ed altro ancora.

I tipi base del C18 sono sufficienti a descrivere buona parte delle strutture dati necessarie allo sviluppo firmware; tuttavia, a partire dalle definizioni viste nella scorsa puntata è possibile costruire qualcosa di più sofisticato. Attraverso la parola chiave *typedef* possiamo creare dei nuovi nomi di tipi dati; ad esempio è possibile creare una variabile booleana chiamata “risultato” che assume due soli valori: vero o falso. Associamo alla condizione logica “falso” il valore 0.

```
typedef enum _BOOL { FALSO = 0, VERO } BOOL;
risultato BOOL;
```

Attraverso la predetta *typedef* possiamo abbinare alle variabili dei nomi mnemonici, che sono i tipi più utilizzati perché permettono uno sviluppo firmware molto

più leggibile e intuitivo. L'adozione di codici mnemonici (cosiddetti perché ricordabili più facilmente di quelli binari) risale ai primi microprocessori ed è un'alternativa alle istruzioni in linguaggio macchina.

```
typedef unsigned char BYTE;           // 8-bit
typedef unsigned short int WORD;      // 16-bit
typedef unsigned long DWORD;         // 32-bit
```

Il vero potere della parola chiave risiede però nell'uso con strutture ed unioni.

Strutture e Unioni

Una struttura si può definire come un insieme di una o più variabili, eventualmente di tipo diverso, raggruppa- ➤

te sotto un unico nome; in questo modo possiamo organizzare dati complessi. Nella realtà, infatti, molto spesso le variabili dichiarate sono legate da relazioni piuttosto strette, pertanto è necessario e conveniente trattarle come fossero una cosa unica.

Analogamente, attraverso le unioni possiamo definire una variabile che può contenere, in istanti differenti, oggetti di tipo e dimensioni diverse. Supponiamo di dover gestire, ad esempio, un nodo di rete; in generale esso viene identificato da due valori fondamentali, che sono l'indirizzo *IP* e l'indirizzo *MAC*. Il primo è costituito da quattro byte, mentre il secondo ha una lunghezza di sei byte. Vediamo come possiamo definire una struttura del genere in maniera efficiente:

```
typedef struct _MAC_ADDR
{
    BYTE v[6];
} MAC_ADDR;

typedef union _IP_ADDR
{
    BYTE    v[4];
    DWORD   Val;
} IP_ADDR;

typedef struct _NODE_INFO
{
    MAC_ADDR    MACAddr;
    IP_ADDR     IPAddr;
} NODE_INFO;
```

Si noti che nell'unione si definisce la possibilità di accedere all'indirizzo *IP* sia attraverso i singoli campi da otto bit, che attraverso il valore complessivo a 32 bit. A questo punto una variabile contenente tutte le informazioni necessarie e sufficienti per identificare un'interfaccia di rete si può dichiarare attraverso la seguente istruzione:

```
NODE_INFO Server;
```

Se vogliamo accedere ai singoli membri della struttura si utilizza l'operatore "." quindi:

```
Server.IPAddr.v[0] = 192;
Server.IPAddr.v[1] = 168;
Server.IPAddr.v[2] = 0;
Server.IPAddr.v[3] = 1;
```

Con questa serie di istruzioni abbiamo valorizzato i membri della struttura relativi all'indirizzo *IP*. Naturalmente attraverso la notazione *#define* è possibile definire anche delle costanti simboliche; in tal caso il compilatore rimpiazzerà con il nome corrispondente tutte le occorrenze del nome assegnato. Ad esempio definiamo il carattere di controllo di una EEPROM (tipo 24LC256); ne risulta una situazione del tipo:

```
#define EEPROM_CONTROL (0xa0)
```

Questa parola chiave ha dei risvolti piuttosto interessanti; ad esempio rende possibile inserire parti di programma a seconda delle definizioni che si trovano in esso. Lo stesso firmware, quindi, viene modificato a seconda dell'utilizzo che se ne fa. Per fare un esempio, diciamo che nello *stack TCP/IP* di Microchip è possibile includere o escludere alcune funzioni inserendo o meno la stringa *#define STACK_CLIENT_MODE* come si vede dal listato seguente:

```
#ifdef STACK_CLIENT_MODE
    Cache.MACAddr.v[0] = remoteNode.MACAddr.v[0];
    Cache.MACAddr.v[1] = remoteNode.MACAddr.v[1];
    Cache.MACAddr.v[2] = remoteNode.MACAddr.v[2];
    Cache.MACAddr.v[3] = remoteNode.MACAddr.v[3];
    Cache.MACAddr.v[4] = remoteNode.MACAddr.v[4];
    Cache.MACAddr.v[5] = remoteNode.MACAddr.v[5];
#endif
```

In questo caso, in pratica, l'indirizzo *MAC* estratto da un pacchetto *ARP* (*Address Resolution Protocol*) viene salvato in cache. Se avessimo compilato il programma omettendo la definizione *STACK CLIENT MODE* tali istruzioni non sarebbero state incluse nel listato. Usando lo stack come server, infatti, non è necessario gestire una cache di indirizzi, visto che non si effettuano delle richieste *ARP*. A beneficio dei neofiti, precisiamo che una richiesta *ARP* è un'interrogazione che un dispositivo di rete invia ad un host per conoscerne l'indirizzo *MAC*; può essere inviata solo ad host dei quali si conosce l'indirizzo *IP*, dato che il pacchetto di dati contenente il comando deve includere l'address *IP* del dispositivo cui è destinato.

Puntatori

Parlando di un linguaggio ANSI C non ci si può scorciare dei puntatori, anche perchè sono essenziali nelle funzioni. Un puntatore si può definire come una variabile che contiene l'indirizzo di un'altra variabile. Ad esempio, se definiamo una variabile intera *A* possiamo assegnare il suo indirizzo ad un puntatore ed utilizzare quest'ultimo per accedere al suo valore. Si veda il seguente codice:

```
int A;
int *punt;

punt = &A;
*punt = 10;
```

In pratica definiamo un puntatore ad un intero e lo chiamiamo *punt*. Attraverso la notazione *&A* assegniamo

l'indirizzo della variabile *A* al puntatore; possiamo quindi assegnare il valore *10* alla variabile *puntata* da *punt*. La variabile *A* assume il valore *10*. Naturalmente i puntatori si possono utilizzare anche per le strutture viste nel paragrafo precedente. Se scriviamo:

```
NODE_INFO PCremoto;
```

possiamo accedere ai singoli membri della struttura con le seguenti istruzioni:

```
PCremoto->MACAddr.v[0] = 0x03;
(*PCremoto).MACAddr.v[0] = 0x03;
```

La prima notazione è quella più utilizzata quando si ha a disposizione il puntatore ad una struttura. Dopo questa piccola parentesi, veniamo all'argomento fondamentale di questa puntata: le funzioni.

Funzioni e Parametri

L'utilizzo di linguaggio di alto livello nello sviluppo firmware, oltre a rendere più semplice la stesura di un programma permette di dividere la risoluzione di problemi complessi in diversi sotto-problemi più semplici. In particolare, attraverso l'uso di funzioni è possibile relegare una serie di istruzioni in una sorta di scatola nera che può essere successivamente utilizzata senza preoccuparsi del suo contenuto. In questo modo è pos-

senza alla nascita del *Framework USB* o delle routine incluse nelle *Application Note* pubblicate sul sito ufficiale Microchip. In generale, nello sviluppo strutturato si usa raccogliere le dichiarazioni di variabili e le definizioni di funzioni in un unico file con estensione ".h". Vediamo nel concreto una funzione usata nello stack TCP/IP ed inclusa nel file *arp.h*.

```
void ARPput (NODE_INFO *remote, BYTE opCode);
```

Si tratta di una routine che invia ad un nodo della rete identificato dalla struttura *NODE_INFO* un pacchetto ARP con un codice operativo definito da un valore che chiamiamo *opCode*; la funzione in questione non ritorna alcun parametro.

La dichiarazione di una funzione in C18 si può sintetizzare con la seguente notazione:

```
tipoparaOUT nomefunzione (tipopara1 nomepara1, tipopara2
nomepara2 ...);
```

tipoparaOUT: tipo del risultato ritornato dalla funzione; nel caso in cui non ci sia alcun parametro in uscita, si utilizza la parola chiave *void*;

nomefunzione: è la denominazione della funzione; tale stringa viene utilizzata nel momento in cui si richiama la funzione all'interno del programma;

tipoparaN: è il tipo di parametro che viene passato alla funzione.

Tabella 1

Directory	Descrizione
\delays	Funzioni per la gestione di ritardi sincronizzati con il clock di sistema
\stdlib	Funzioni standard per la manipolazione di stringhe, funzioni matematiche, classificazione di caratteri (stabilire se un carattere è una cifra o una lettera), conversioni tra diversi tipi
\pmc\ADC	Funzioni per il modulo A/D del PIC
\pmc\I2C	Funzioni per gestire la comunicazione con dispositivi basati su I2C come le EEPROM 24LCXX
\pmc\SPI	Funzioni per gestire la comunicazione con dispositivi basati su SPI come le card SD e MMC
\pmc\PWM	Funzioni per generare impulsi modulati in ampiezza attraverso l'apposito modulo del PIC
\pmc\USART	Funzioni per gestire comunicazioni seriali
\pmc\XLCD	Funzioni per gestire LCD basati su controller Hitachi 44780

sibile inserire nel nostro firmware delle parti di codice che abbiamo sviluppato in altri progetti, aumentando la nostra produttività in maniera notevole. Analogamente possiamo utilizzare delle librerie messe a disposizione da altri, come, ad esempio, quelle di Microchip. Esse sono raggruppate sotto la directory **mcc18\src\pmc** e consentono di gestire buona parte delle funzionalità offerte dalla classe PIC18, come si vede nella Tabella 1. Naturalmente l'elenco non è esaustivo, visto che lo sviluppo di librerie specifiche è in continua evoluzione: si

nomeparaN: è il nome del parametro passato alla funzione. Esso verrà utilizzato all'interno del codice della stessa per manipolarne il valore.

È importante tenere presente che il passaggio di parametri nelle funzioni C18 avviene normalmente per valore; ciò significa che le istruzioni all'interno delle funzioni non possono agire direttamente sulle variabili passate come parametro, ma solo sul valore di una loro copia privata. Per ovviare a questa situazione, che potrebbe sembrare una limitazione, si utilizzano i pun-

tatori. Vediamo il caso della *ARPPut*. Per il codice operativo ci interessa soltanto acquisire il relativo valore; infatti dobbiamo gestire soltanto due tipi di codici: REQUEST nel caso di interrogazione e REPLY nel caso di risposta ad un host remoto. Quindi, per questo parametro è sufficiente accedere al relativo valore, visto che dobbiamo soltanto inizializzare il campo del pacchetto ARP prima di inviarlo. Le seguenti costanti simboliche permettono di schematizzare i due possibili valori:

```
#define ARP_REPLY      (0x00)
#define ARP_REQUEST    (0x01)
```

Per l'altro parametro, invece, il discorso è differente. Come abbiamo visto nei paragrafi precedenti, infatti, la struttura *NODE_INFO* contiene l'indirizzo *MAC* e *IP* del nodo remoto con il quale vogliamo comunicare. È necessario, quindi, poter accedere alle relative variabili in maniera da poterle modificare. In particolare, sarà necessario estrarre l'indirizzo *MAC* dal pacchetto in risposta e inserirlo nella relativa struttura per poterlo utilizzare nelle connessioni future. Focalizziamo la nostra attenzione sulla seguente sequenza di istruzioni:

```
if (((packet.SenderIPAddr.v[0] ^ remote->IPAddr.v[0]) &
    MY_MASK_BYTE1)
||
    ((packet.SenderIPAddr.v[1] ^ remote->IPAddr.v[1])
    &
    MY_MASK_BYTE2)
||
    ((packet.SenderIPAddr.v[2] ^ remote->IPAddr.v[2])
    &
    MY_MASK_BYTE3)
||
    ((packet.SenderIPAddr.v[3] ^ remote->IPAddr.v[3])
    &
    MY_MASK_BYTE4)
)
{
    packet.TargetIPAddr.v[0] = MY_GATE_BYTE1;
    packet.TargetIPAddr.v[1] = MY_GATE_BYTE2;
    packet.TargetIPAddr.v[2] = MY_GATE_BYTE3;
    packet.TargetIPAddr.v[3] = MY_GATE_BYTE4;
}
else
    packet.TargetIPAddr      = remote->IPAddr;
```

In questo caso dobbiamo verificare se l'host remoto a cui indirizziamo il pacchetto *ARP* si trova sulla nostra stessa sottorete; allo scopo eseguiamo un *AND* logico tra l'indirizzo IP remoto e la nostra maschera di sottorete. Viene poi fatto l'*OR* esclusivo con il nostro indirizzo IP. Nel caso in cui almeno uno dei risultati sia pari a 1, significa che l'host remoto si trova su un'altra sottorete, quindi per raggiungerlo è necessario che il pacchetto transiti attraverso il nostro gateway. Nel caso in cui, invece, l'IP remoto si trovi nella nostra stessa sot-

torete, possiamo assegnare l'indirizzo remoto a quello del destinatario del pacchetto. Si noti che per accedere al valore dell'indirizzo passato come parametro si usa l'operatore "->".

Naturalmente, per chiamare questa funzione siamo costretti ad utilizzare la seguente notazione:

```
ARPPut(&PCRemoto, ARP_REQUEST);
```

In questo modo passiamo correttamente l'indirizzo della variabile *PCRemoto* al relativo puntatore. Si faccia attenzione che è necessario utilizzare l'operatore "&". In altri casi l'omissione di tale operatore può comportare l'accesso e la scrittura di aree di memoria riservate ad altre variabili comportando degli errori difficili da gestire. Al termine dell'elaborazione una funzione passa all'esterno il risultato attraverso la parola chiave *return*. Ad esempio, nello *stack TCP/IP* tra le funzioni di supporto troviamo una piccola routine che effettua lo swap tra i due byte di un valore a 16 bit e che trasmette in uscita il valore così modificato. Ecco come si presenta la sequenza di istruzioni:

```
typedef union _WORD_VAL
{
    WORD Val;
    BYTE v[2];
} WORD_VAL;

WORD swaps(WORD v)
{
    WORD_VAL t;
    BYTE b;

    t.Val = v;
    b = t.v[1];
    t.v[1] = t.v[0];
    t.v[0] = b;

    return t.Val;
}
```

In pratica, attraverso la variabile di comodo *b*, i due byte vengono scambiati sfruttando il vettore *v* definito nell'unione. Ogni programma C18 ha una funzione fondamentale chiamata *main* che contiene la sequenza di istruzioni principale. L'esecuzione del codice avviene a partire dalla *main* e prosegue attraverso le chiamate alle altre funzioni. Chiaramente, per poter effettuare correttamente la chiamata di una funzione esterna alla *main* dobbiamo includere la definizione della stessa attraverso la seguente notazione:

```
#include "nomefile.h"
```

In essa *nomefile* è il nome del file con estensione *.h* che contiene le definizioni relative alle funzioni che inten-

diamo richiamare; a riguardo dobbiamo fare una piccola precisazione sintattica: nel caso il file si trovi nella directory di sistema (tipicamente *mcc18h*) al posto delle virgolette si utilizzano come delimitatori i segni di maggiore e minore (< >). Supponiamo, ad esempio, di

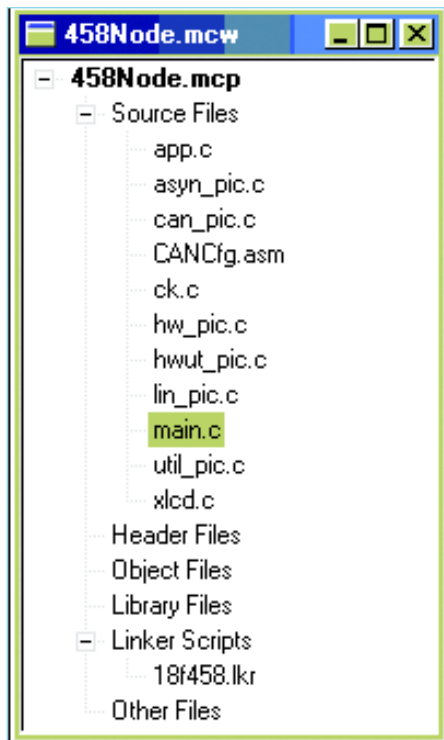


Fig. 1

corso di ricerca per gli header di sistema quello registrato nella variabile d'ambiente denominata *MCC_INCLUDE*. Nel caso, invece, il nome del file sia racchiuso tra virgolette, lo cercherà sia nel percorso locale dal quale avviamo il compilatore, sia nella directory di sistema (*mcc18h*).

Ma vediamo come si presenta un progetto completo di funzioni esterne all'interno del *MPLAB IDE* (Fig. 1).

Si tratta di un progetto che riguarda un nodo CAN sviluppato sul microcontrollore PIC18F458. Come si vede chiaramente, nella struttura si identifica il corpo del listato principale (*main.c*), dello script necessario al linker (*18f458.lkr*), del file di configurazione del PIC (*CANCfg.asm*) ed una serie di altri file con estensione *.c* ognuno dei quali contiene delle funzioni specifiche. Ciascuno di questi file ha il proprio header (estensione *.h*) contenente le dichiarazioni delle funzioni e delle necessarie variabili. Possiamo modificare tali funzioni facendo doppio clic sulla voce corrispondente; si apre in questo modo una finestra di editing che ci permette di vedere il relativo listato.

Se prendiamo ad esempio il file *asyn.c*, troviamo le funzioni che gestiscono la comunicazione seriale del nodo verso un PC (Fig. 2).

In questo caso ci troviamo posizionati sulla sequenza relativa all'invio di un carattere attraverso la seriale. Nel listato troviamo anche una *include* corrispondente

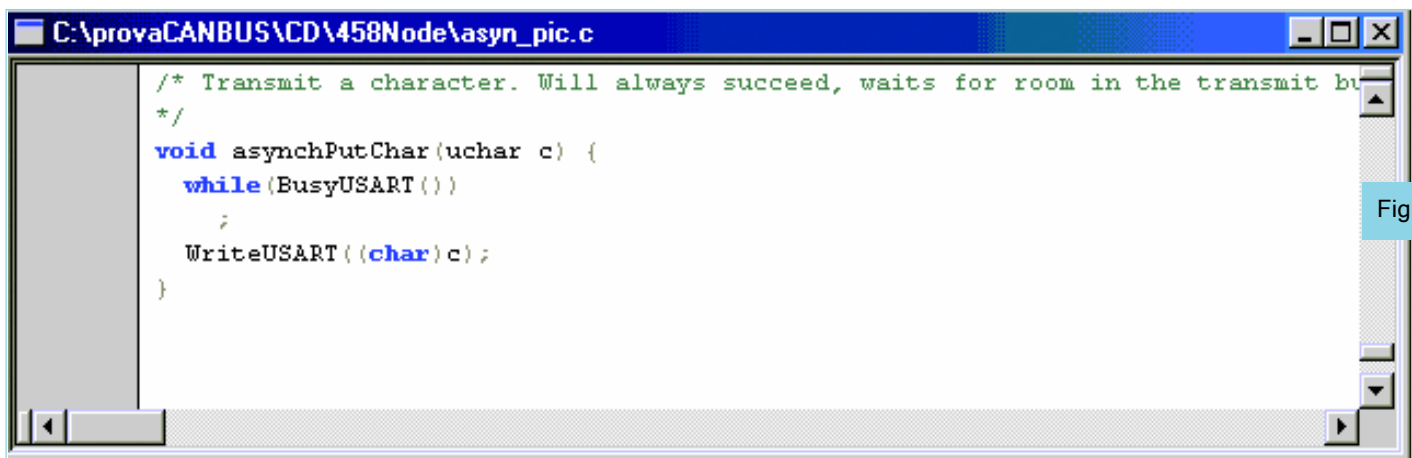


Fig. 2

voler utilizzare nel nostro firmware le funzioni matematiche della libreria standard; a tal fine dovremo inserire nel nostro listato l'istruzione seguente:

```
#include <math.h>
```

Se andiamo a verificare il file *math.h* nella directory *mcc18h*, vedremo una ventina di funzioni interessanti che permettono di calcolare dei valori anche piuttosto complessi. In generale il compilatore utilizza come per-

alla *USART.h*; in pratica, le due funzioni richiamate sono presenti nella libreria standard: la prima (*BusyUSART*) restituisce un valore logico vero finché la porta seriale non risulta libera, mentre la seconda (*WriteUSART*) invia tramite la stessa seriale il carattere *c*, passandolo come parametro.

In quest'ultimo caso si utilizza il *casting* del parametro, anteponendo tra parentesi il tipo di dato: ciò serve a fare in modo che il valore passato sia pienamente compatibile con quello necessario alla funzione chiamata. Ma ➤

Tabella 2

	ROM	RAM
far	un qualunque indirizzo nell'area di programma	un qualunque indirizzo nell'area dati
near	un indirizzo inferiore a 64K nell'area di programma	un indirizzo contenuto nell'area dati ad accesso diretto

come viene trattato il parametro che passiamo ad una funzione? Quali sono le regole che stabiliscono la visibilità delle variabili dichiarate e la loro allocazione? Nei prossimi paragrafi troverete le risposte a queste domande.

Regole di visibilità e qualificatori

Per indirizzare la memoria di programma, i microcontrollori della famiglia PIC18 utilizzano un bus differente da quello usato per indirizzare la memoria che contiene i dati. Pertanto, per regolare l'allocazione di una variabile nei due rispettivi spazi di memorizzazione, sono previsti due differenti qualificatori: si utilizza il qualificatore *rom* per allocare l'oggetto nella memoria di programma, mentre *ram* è usato per allocare lo stesso oggetto nella memoria riservata ai dati.

Si noti che i puntatori "puntano" in generale nell'area dati, a meno che non siano dichiarati con il qualificatore *rom*. In aggiunta si possono utilizzare i qualificatori *far* e *near*. Nella Tabella 2 sintetizziamo le diverse opzioni possibili.

Ricordiamo che l'area dati ad accesso diretto è costituita da quelle locazioni nei banchi 0 e 15, ai quali si può accedere indipendentemente dal contenuto del registro *BSR* (*Bank Select Register*). È chiaro che una variabile *ram* allocata con il qualificatore *far* comporta l'utilizzo del *BSR* per selezionare il banco nel quale essa viene conservata. Si faccia attenzione che un puntatore *rom far* occuperà un maggior spazio (24bit) rispetto ad un puntatore *rom near* (16 bit).

Le classi di memorizzazione utilizzate dal C18 sono le stesse dell'ANSI C. In generale, tutte le variabili dichiarate in una funzione sono di classe *auto*, pertanto

IPER
SICUREZZA.IT

www.ipersicurezza.it

ELECTRONIC SECURITY SYSTEM

*Microspie audio/video - Investigazione - Sicurezza casa -
Bonifiche ambientali - Criptaggio telefonico - Difesa personale -
Videosorveglianza - Modellismo - Kit montaggio - GPS*

esse vengono create nel momento in cui la funzione viene richiamata e “spariscono” al termine della stessa. Si tratta quindi di variabili locali e dinamiche, le quali tra una chiamata e l'altra non mantengono il proprio valore (ciò vale naturalmente anche per gli argomenti di funzione); in particolare, se non vengono opportunamente inizializzate conterranno, all'entrata della funzione, dei valori casuali.

Se vogliamo che il valore di una variabile venga preservato nelle successive chiamate della funzione nella quale viene dichiarata, dobbiamo utilizzare la classe *static*; si faccia attenzione che in questo modo si crea una locazione di memoria privata e permanente di una funzione. Per quanto riguarda la visibilità, sia le variabili *static* che quelle *auto* definite in una funzione sono accessibili solo all'interno di tale funzione. Analogamente, se la dichiarazione avviene in testa ad un file contenente più funzioni le variabili saranno accessibili solo all'interno di tale file e in nessun altro listato del nostro progetto.

Nel caso in cui una variabile locale non presenti alcun qualificatore, viene considerata parte della classe *auto*. Se vogliamo creare delle variabili globali accessibili anche da altre funzioni, dobbiamo usare un'ulteriore classe chiamata *extern*; in questo caso dobbiamo distinguere tra la dichiarazione e la definizione della variabile: ricordate che una dichiarazione stabilisce le sue caratteristiche (tipo, lunghezza) mentre una definizione alloca anche la relativa memoria.

Pertanto, se vogliamo far sì che una variabile sia accessibile da varie funzioni in più listati differenti, dovremo prima definirla, ad esempio, in un file, poi dichiararla attraverso il qualificatore *extern* nei listati contenenti le funzioni che devono accedervi. Chiaramente, l'estrazione della dichiarazione in un file esterno e l'utilizzo della *#include* in questo caso aiuta molto.

Un'altra classe di memorizzazione è qualificata attraverso la parola *register*. In questo caso si stabilisce che la relativa variabile verrà usata frequentemente, pertanto il compilatore tenterà di collocarla in uno dei registri del micro per aumentarne la velocità di accesso; naturalmente, considerando lo scarso numero di registri è bene utilizzare tale classe con molta parsimonia.

Esiste ancora un'ultima classe chiamata *overlay*; una variabile dichiarata appartenente a tale classe all'interno di una funzione (non è possibile usare tale classe per parametri, funzioni o variabili globali) viene allocata staticamente, il che rende l'accesso più rapido e allo stesso tempo consente un notevole risparmio di spazio. Infatti, se due variabili *overlay* vengono dichiarate in due funzioni separate e non in relazione tra di loro (nel senso che all'interno di una non c'è la chiamata all'al-

tra) possono condividere la stessa locazione di memoria. In pratica, durante l'esecuzione della prima funzione conterrà i valori della prima variabile, mentre nell'esecuzione della seconda funzione conterrà i valori della seconda variabile. L'allocazione statica comporta solo il fatto che viene riservata una locazione a tale scopo; però, tra una chiamata e l'altra il valore in essa contenuto non viene conservato.

Si faccia attenzione al fatto che i parametri di una funzione possono essere *static* o *auto*. In generale, se non si mette alcun qualificatore il parametro viene considerato parte della classe *auto*.

Se, invece, definiamo il parametro esplicitamente come *static*, esso non viene inserito nello stack ma allocato globalmente, il che ne permette l'accesso diretto. Se diamo un'occhiata allo stack *TCP-IP* di Microchip, nel file *ARPTsk.c* troveremo un tipico utilizzo di una variabile *static* con questa sequenza di istruzioni:

```
#ifdef STACK_CLIENT_MODE
static NODE_INFO Cache;
#endif
```

Lo stack può essere utilizzato per lo sviluppo sia server che client; in quest'ultimo caso viene dichiarata una variabile *cache* che contiene le informazioni fondamentali di un dispositivo remoto “scoperto” attraverso, ad esempio, un pacchetto *ARP*. La variabile può contenere l'indirizzo *IP* e *MAC* di un server FTP o SMTP con il quale vogliamo far comunicare il nostro PIC in qualità di client. È chiaro che in questo caso la dichiarazione *static* è fondamentale per condividere tali informazioni tra tutte le funzioni che devono utilizzarle; infatti, una volta che esse vengono valorizzate, possono venire utilizzate anche dalle altre funzioni *ARP*: ad esempio per verificare se l'IP di un host è stato risolto o se si trova nella nostra sottorete.

Infine, è interessante vedere una dichiarazione *extern*, per quanto riguarda lo stato del modulo che si occupa del *DHCP* (*Dynamic Host Configuration Protocol*). La sequenza di istruzioni che fa parte del file header corrispondente (*dhcp.h*) è:

```
#if !defined(THIS_IS_DHCP)
extern SM_DHCP smDHCPState;
#endif
```

Ogni modulo che implementa uno dei livelli *ISO/OSI* dello stack o un particolare protocollo, inizia con la dichiarazione del tipo *THIS_IS_...* In questo caso vediamo che nel modulo principale (quello che, per intenderci, contiene la *THIS_IS_DHCP*) *dhcp.c* troviamo la ▶

definizione della variabile *smDHCPState*, con la sua inizializzazione:

```
SM_DHCP smDHCPState = SM_DHCP_INIT;
```

Attraverso la direttiva *#if !defined - endif* facciamo in modo che in tutti gli altri file contenenti la *#include dhcp.h*, eccezion fatta per quello principale, la variabile in questione sia accessibile. Questo fatto è molto importante perché, come vedremo in seguito, ciascun livello può assumere diversi stati, ognuno dei quali viene identificato da variabili che iniziano per *sm* (*State Machine*). Se queste variabili non fossero globali ed accessibili dalle funzioni che fanno uso del *DHCP* (tutte quelle che sono raggruppate in file contenenti la *#include dhcp.h*) si creerebbero non pochi problemi di funzionamento dei dispositivi.

Integrazione di un progetto

Molto spesso è necessario intervenire per modificare un progetto preesistente (magari derivante da una *application note*) o adattare a uno scopo specifico una struttura dimostrativa (ad esempio una demoboard) messa a disposizione dal produttore di un microcontrollore,

come nel caso dello *stack TCP/IP*. In entrambe le situazioni è necessario aggiungere delle funzionalità sotto forma di funzioni richiamabili all'occorrenza.

Per rendere più chiaro il concetto supponiamo di voler inserire in una scheda generica il supporto per le SD-Card: oltre ad effettuare le necessarie modifiche allo schema del circuito originario, svilupperemo un'apposita libreria contenente tutte le funzioni necessarie a gestire le varie fasi di utilizzo di una memoria SD.

Potremmo creare, ad esempio, una funzione *BOOL SdInit(void)* che gestisce l'inizializzazione della card e che ci restituisce un valore VERO o FALSO a seconda che essa sia andata a buon fine oppure no.

Il modo migliore per realizzare un'integrazione è quello di raggruppare le definizioni di ciascuna funzione e delle variabili globali necessarie in un file header (ad estensione *.h*) e sviluppare ciascuna funzione raggruppandola in un file denominato in maniera opportuna. Ad esempio, aggiungiamo i nostri due file (Fig. 3 e 4) al progetto del nodo CAN, attraverso la solita procedura vista nelle precedenti puntate.

Aggiungendo i relativi file *SD.c* ed *SD.h* (evidenziati, nella Fig 4, con il contorno rosso) avremo la possibilità di richiamare le funzioni di accesso per le SD-Card, all'interno sia del *main* che di altre routine inglobate negli altri file con estensione *.c*. Dovremo soltanto inse-

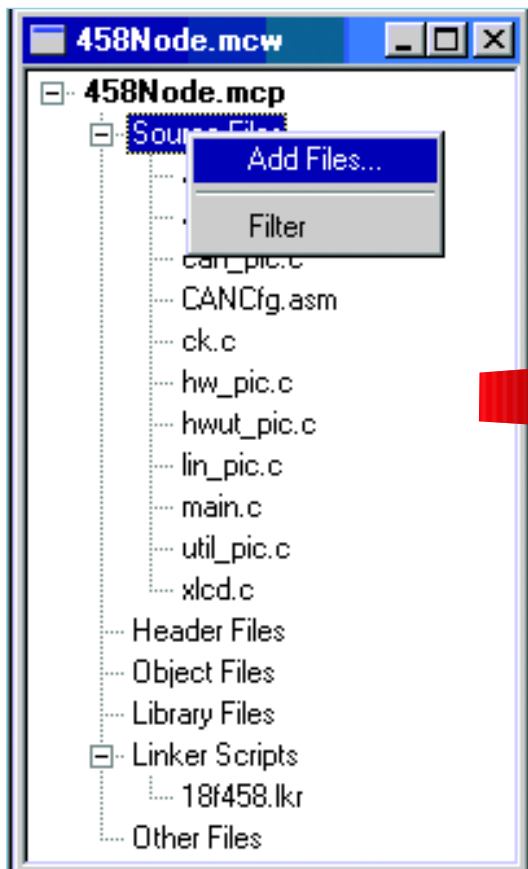


Fig. 3

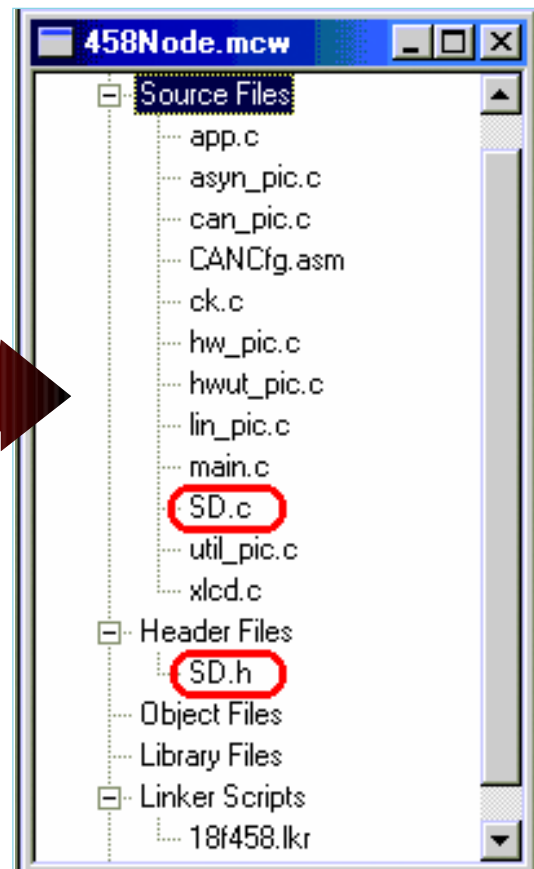


Fig. 4

rire la *include* corrispondente: `#include "SD.h"`. In particolare, osservate che l'aver aggiunto tale header all'elenco degli oggetti del progetto serve per renderne più comodo l'editing e quindi la modifica, nel momento in cui aggiungeremo ulteriori funzioni.

Per la corretta compilazione ciò non è necessario, visto che il compilatore cercherà direttamente il file nella directory locale del progetto.

Come si evince dal discorso appena fatto, l'aggiunta di funzionalità risulta molto chiara e permette in ogni momento di ritornare sui propri passi; possiamo, infatti, identificare facilmente le singole parti di cui si compone il nostro firmware, agendo di volta in volta sul file che ci interessa. Soltanto al termine, lavoreremo nel *main* per mettere insieme il tutto.

Questo genere di approccio ci permette di dividere problemi molto complessi in singoli blocchi più semplici e quindi più facilmente risolvibili.

Non solo, se a qualche mese di distanza dalla stesura del nostro progetto software dovremo mettervi mano modificando una determinata funzionalità, potremo riuscire agevolmente nell'intento entrando in *editing* nel file corrispondente, effettuando i cambiamenti necessari e ricompilando il tutto con molta più facilità, soprattutto se abbiamo usato un'efficiente metodologia di denominazione.

In un sistema dove abbiamo a disposizione un unico grande listato, magari anche con delle variabili globali riutilizzate per più scopi, una cosa del genere comporterebbe molto più lavoro, sia per identificare la parte di codice responsabile di una determinata funzionalità, sia per effettuare delle modifiche che non vadano ad intaccare altre fasi dell'elaborazione. Sotto questo punto di vista, l'utilizzo del C18, oltre a rappresentare una soluzione decisamente professionale, è anche un modo per aumentare notevolmente la produttività nello sviluppo di software.

Bene, dopo questo meritato elogio del C18, siamo giunti al termine della terza puntata di quello che, più che come un corso, si può ritenere una sorta di manuale operativo (*How-To*) sul C18.

I concetti che stiamo affrontando vogliono essere da un lato un utile ripasso per quanti conoscono il linguaggio C, mirato a far loro scoprire un'ulteriore applicazione, ma soprattutto un'esposizione chiara degli strumenti e delle modalità operative che sono necessari per comprendere lo sviluppo di alcuni progetti di sicuro interesse che intendiamo presentare sulla rivista.

Nella prossima puntata, con la quale intendiamo concludere il nostro breve percorso, parleremo di integrazione con moduli *assembly*, gestione di *interrupt*, stringhe e dettagli di *debug*.



Fiera
dell'informatica elettronica e radiantismo

Fiera dell'Informatica
dell'Elettronica e del RAdiantismo

25 e 26 febbraio 2006
apertura dalle ore 9,00 alle ore 18,00

Rovigo

Info: 0425 28360
www.arearebus.com/fiera/index.html
Cen. Ser. - Divisione Rovigo Fiere
Viale Porta Adige 45 - Rovigo

iniziative collaterali

SABATO 25 ore 10,00
conferenza
"LINUX e FORUM SOURCE"
il software libero
cos'è il software libero: la sua filosofia
Linux a casa e in azienda
installazione ed uso
un programma per ogni esigenza
il software didattico

ore 11,00
workshop
"COME INSTALLARE LINUX"
come installare programmi free
(open office - video - audio) in ambiente
Linux

DOVE SIAMO

provenendo da PADOVA:
nel casello di S. Maria
Piazzale, girare a destra in via
Rovigo e continuare in via
Porta Adige

provenendo da BOLOGNA:
nel casello di Rovigo,
girare a sinistra per via Veneto
e continuare in Viale Porta
Adige.

Flash upgrade per PicStart Plus

a cure della *Redazione*



Per il più diffuso programmatore dei suoi microcontrollori, Microchip rende oggi disponibile il modulo flash per aggiornare il firmware senza dover tutte le volte programmare un nuovo micro e sostituirlo a quello esistente. Si installa facilmente nello zoccolo del PIC17C44 e...

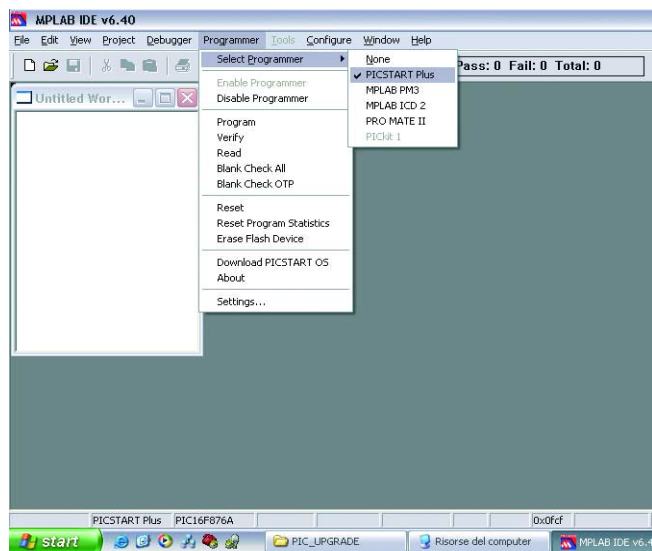
Il PicStart Plus è stato ed è il programmatore per microcontrollori Microchip di maggiore successo, sia per il prezzo d'acquisto decisamente abbordabile, che per la sua versatilità: collegato a un Personal Computer nel quale giri l'apposito software MPLAB, permette infatti di programmare la gran parte dei microcontrollori PIC. Per renderlo sempre attuale ed adattarlo ai nuovi micro immessi in commercio, Microchip rilascia periodicamente aggiornamenti del firmware da caricare nel PIC montato sulla scheda, che presiede il funzionamento dell'intero programmatore.

Purtroppo per aggiornare il firmware del dispositivo occorre, dopo aver scaricato dal sito Internet di Microchip o prelevato da un disco il relativo file .exe, trasferirlo in un nuovo micro OTP vergine, che non potrà più essere aggiornato; ciò significa che ogniqualvolta si desidera aggiornare il firmware del PicStart Plus bisogna acquistare un nuovo PIC17C44. Per ovviare a questo inconveniente e al fatto che presto il PIC17C44 sarà considerato obsoleto e diverrà quindi poco reperibile, Microchip ha realizzato il modulo PicStart Plus Flash Upgrade Module: si tratta di un pic-

colo circuito stampato sul quale è montato uno zoccolo da 20+20 pin con contatti a tulipano, dimensionato per essere inserito nello zoccolo del PIC che equipaggia il programmatore, al suo posto.

Contiene un recente microcontrollore PIC18F6720 provvisto di flash-EPROM, nel quale si può trasferire un nuovo file di aggiornamento del firmware quante volte si vuole, senza più dover riprogrammare un nuovo micro OTP, toglierlo dal texttool, aprire il contenitore, estrarre il vecchio OTP (ormai inutile) e mettervi quello appena

Fig. 2: dalla finestra principale del programma MPLAB, selezionate il programmatore PICSTART PLUS.



programmato. Ritenendo che il progetto possa interessare molti lettori, ci siamo procurati il kit di upgrade, l'abbiamo provato e vi spieghiamo come installarlo: dalle poche spiegazioni comprenderete come la cosa sia davvero alla portata di tutti. Partiamo dalla premessa che possono installare e gestire il modulo di aggiornamento tutti gli utenti che hanno installato nel loro computer il software MPLAB IDE versione 6.40 o successiva e che usano un programmatore PICSTART Plus Rev 5 o più recente; a riguardo, precisiamo che la revisio-

Fig. 1: la finestra per l'installazione del software MPLAB v. 7.20.

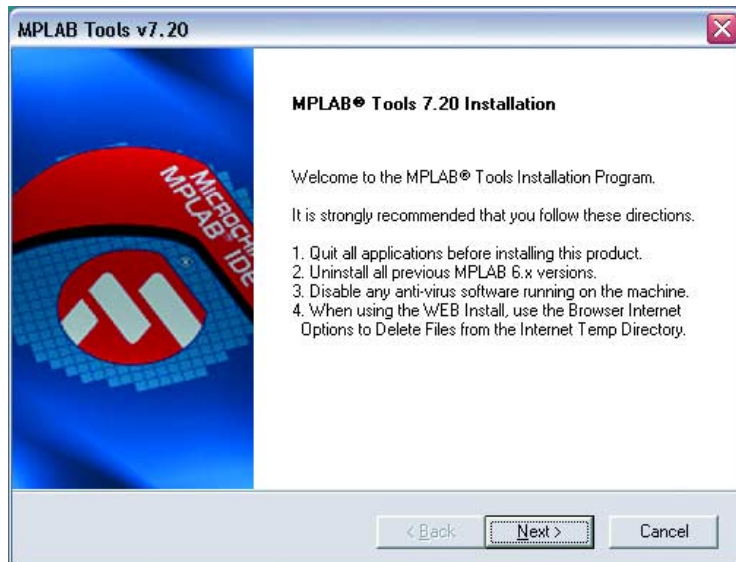
ne hardware del PICSTART Plus è l'ultimo numero immediatamente seguente la lettera -R del serial number riportato sul fondo del contenitore. Chi

avesse una release di MPLAB IDE più datata, potrà scaricare la 7.20 sia dal sito www.microchip.com, sia dal CD-ROM fornito in dotazione al kit di upgrade. Detto ciò, entriamo nel vivo e diciamo che il kit di aggiornamento comprende il modulo flash, un manuale di istruzioni e un CD-ROM contenente la release 7.20 dell'MPLAB e due versioni di firmware. Per

prima cosa bisogna aprire il programmatore e sostituire il PIC17C44 installato con il modulo, badando di orientarlo correttamente rispetto al bordo della scheda; non serve programmare alcunché, quindi niente operazioni preliminari con il text



La modifica consiste in una basetta contenente un PIC dotato di flash.



perché il modulo di upgrade viene fornito con un firmware molto recente. La release preinstallata (4.30.3) serve a far funzionare il programmatore ma costituisce anche la base per installare nuove release eventualmente disponibili in Internet. Assemblato nuovamente il programmatore, bisogna collegarlo alla porta seriale del computer ed alimentarlo; avviato il sistema, se già si è in possesso dell'MPLAB basta avviarlo, mentre, in caso contrario, inserito il CDROM nel lettore bisogna avviare l'installazione di MPLAB versione 7.20 (Fig. 1). Il software funziona in ambiente Microsoft Windows 98, 2000/XP.

Avviato MPLAB, dal menu *Programmer* impartite il comando *Select Programmer* e, nel sottomenu cui si accede, spuntate *PICSTART Plus*; ora dal solito menu impartite il comando *Enable Programmer*, senza il quale il programmatore resta a riposo e non può compiere operazioni. Sempre dallo stesso menu, impartite il comando *Download PICSTART OS*: si apre così la finestra di dialogo che mostra i file .exe disponibili per l'upgrade. A riguardo

va detto che se i file si trovano su un'unità a disco occorre sfogliare *Esplora Risorse* fino a trovare la directory in cui sono posizionati, ➤

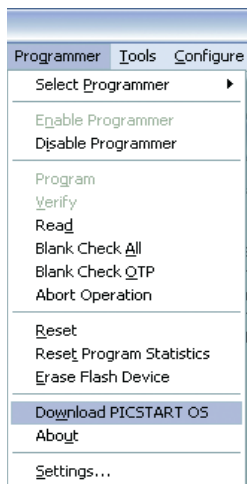


Fig. 3: per aggiornare il firmware, dal menu *Programmer* bisogna impartire il comando *Download PICSTART OS* e, quando richiesto dall'MPLAB, definire il nome del file .exe.

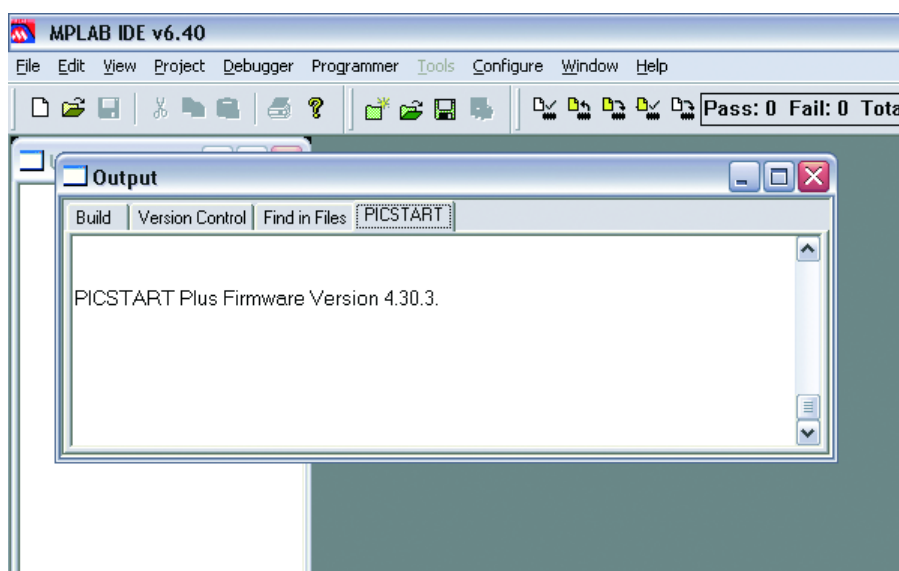
ovvero copiarli dall'unità in questione direttamente nella cartella *PicstartPlus*. Nel CD in dotazione al kit si trovano le versioni 4.30.3 e 4.30.4; installiamo quest'ultima, perché la prima è preinstallata: per iniziare il trasferimento nel programmatore del file con l'aggiornamento firmware, fate doppio clic sul nome del file corrispondente, ovvero selezionatelo e fate clic sul pulsante *Apri* della finestra di dialogo che appare. Inizia così l'upgrade, evidenziato dall'apertura della finestra di dialogo nella quale vediamo lo stato di avanzamento dell'operazione. Al completamento del download e dell'operazione di aggiornamento, il programma risponde con un'apposita finestra di dialogo che riapi-

loga l'operazione e visualizza l'esito, ovvero la versione corrente del firmware caricata nella flash.

Per futuri aggiornamenti, ricordate che in ogni momento potete interrogare il programmatore al fine di conoscere quale versione di firmware vi è installata: basta accedere al menu *Programmer* e impartire il comando *About*, quindi sfogliare la scheda *PICSTART* della finestra di dialogo

Bene, diciamo pure che la trasformazione del programmatore PicStart Plus originale in uno flash è conclusa; quando Microchip renderà disponibili nuove release di firmware vi accorgete di quanto sarà semplice aggiornarlo: basterà scaricare il file nel computer (dal sito Microchip o da un'unità a disco removibile) avviare il programma

Fig. 4: dalla finestra di dialogo *Output* si può conoscere in ogni momento la release di firmware presente nel PICSTART Plus.



(*Output*) che si apre. Ciò vi eviterà di andare a scrivere nel modulo di modifica una versione di firmware magari meno aggiornata di quella in uso.

MPLAB, abilitare il programmatore e, con l'apposito comando del menu *Programmer*, avviare la procedura indicando qual è il nuovo file .exe con cui aggiornare il dispositivo. A chi desidera maggiori informazioni riguardo all'upgrade, suggeriamo di visitare la sezione *Design* (a sinistra) della pagina

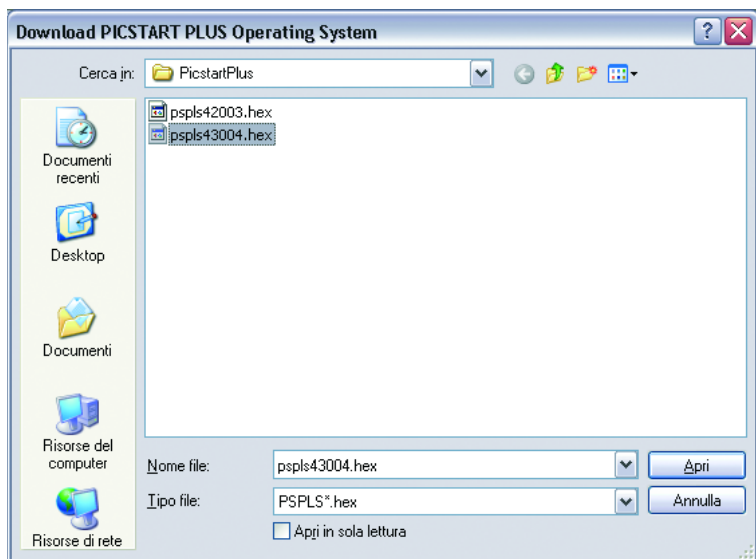
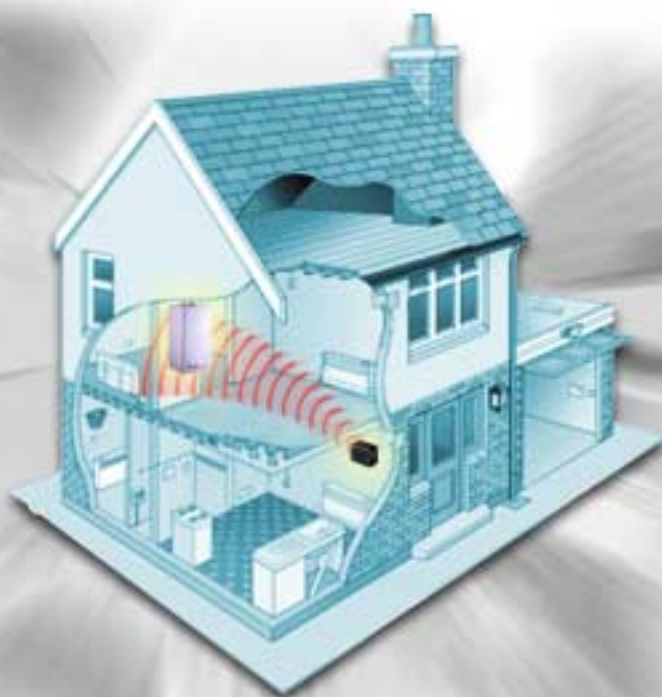


Fig. 5: la cartella *PicstartPlus* riassume i file disponibili per l'aggiornamento del firmware del programmatore; nei file illustrati il numero indica la release corrispondente.

Web www.microchip.com e fare clic sul link *Development Tools*, quindi, nella pagina cui si accede, su *PICSTART-plus*. L'upgrade (cod. PIC-FLASH-UPG) costa 56,00 Euro e può essere acquistato presso la ditta Futura Elettronica di Gallarate (www.futurashop.it).

Termostato con attivazione via radio

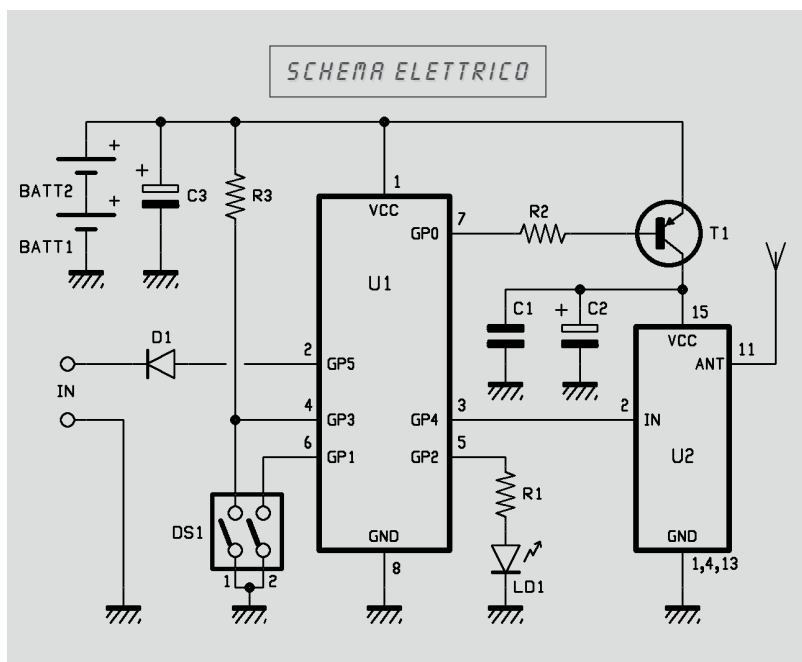
di *Davide Scullino*



Permette di azionare a distanza via radio una caldaia o un condizionatore d'aria usando un tradizionale termostato al quale è stato abbinato un sistema radio codificato. Il trasmettitore è collegato al termostato mentre il ricevitore è posto in prossimità della caldaia o del condizionatore. La portata del sistema è sufficiente per un appartamento o per una villetta.

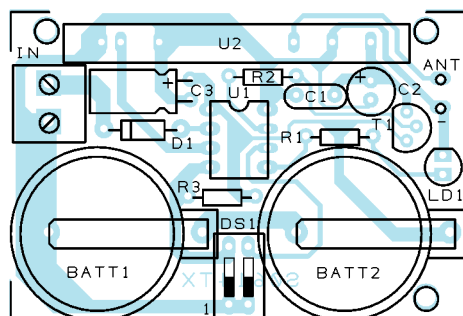
Quando bisogna installare qualcosa di nuovo che richieda l'alimentazione elettrica, ci si pone solitamente il problema di come collegarlo; se c'è vicino una presa di "forza motrice" basta un cavo quasi invisibile, ma quando il collegamento obbliga a compiere un percorso lungo e che passa da un locale all'altro o da un piano all'altro, bisogna prendere una decisione: far passare i cavi in canaline esterne e visibili, pregiudicando l'estetica dei locali, o metterli "sotto traccia" incaricando un muratore di scavare in pareti e soffitti per murare le guaine? Normalmente la maggior parte delle per-

sone opta per la seconda soluzione, sebbene comporti l'avere per casa polvere e frammenti di macerie fino a lavoro ultimato. Quando non è necessario portare tensione ma collegare a un apparato periferiche quali sonde, timer e termostati, si può aggirare l'ostacolo adottando connessioni senza fili: il successo degli impianti antifurto wireless e delle W-LAN (le reti locali di computer...) testimonia come ciò sia ormai di uso comune. Ad esempio, volendo comandare la caldaia murale del riscaldamento con un termostato che, per esigenze funzionali (è il luogo in cui la temperatura ➤



ELENCO COMPONENTI:

R1: 470 ohm	U1: PIC12F675	Varie:
R2, R3: 4,7 kohm	(MF614TX)	- Zoccolo 4+4
C1: 100 nF multistrato	U2: TX4MAVPF10	- Morsetti 2 poli
C2: 220 µF 16 VL	D1: 1N4007	- Portabatterie da CS
elettrolitico	T1: BC557	per CR2032 (2 pz.)
C3: 220 µF 16 VL	LD1: led 3 mm verde	- Circuito stampato
elettrolitico	DS1: Dip-switch 2 poli	codice S0614TX



cambia più frequentemente o quello più al riparo dal sole e da fonti di calore...) va collocato a una certa distanza o su un piano diverso da quello dove è installata, invece di posare decine di metri di canaline o "sventrare" le mura di casa per incassare le guaine corrugate dei cavi, si può pensare di trasmettere il comando di attivazione dal termostato a un relé che chiuda il contatto della caldaia. In questo articolo vi spieghiamo come farlo con la massima semplicità, versatilità e sicurezza.

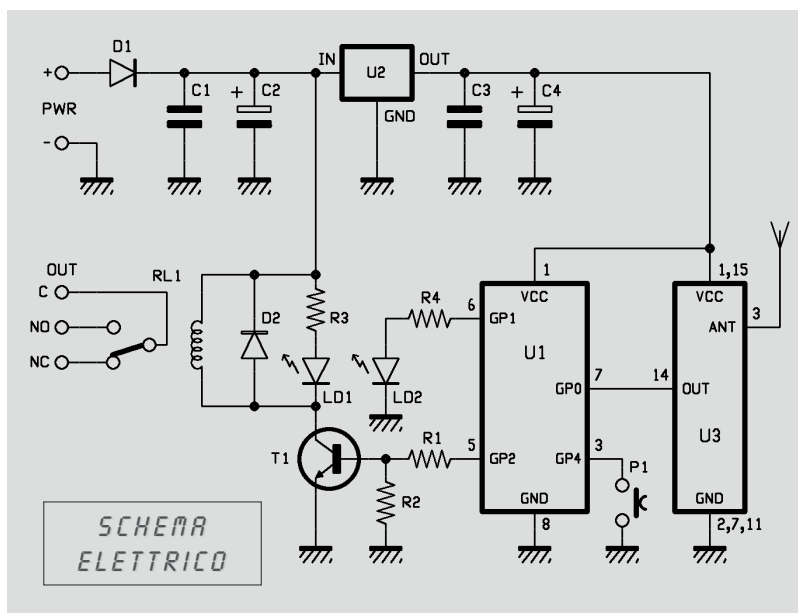
La nostra proposta riguarda un sistema di connessione via radio in UHF (alla frequenza di 433,92 MHz) da abbinare al normale termostato, composto da un trasmettitore e un ricevitore: il primo è l'elemento che si collega al contatto del termostato e da esso viene comandato quando si chiude; il secondo dispone di un relé il cui scambio si chiude quando viene chiuso il contatto d'ingresso del TX. Insomma, il nostro dispositivo può essere assimilato a un radiocomando, dal quale differisce per tutta una serie di accorgimenti necessari a garantire che compia la funzione richiesta; già, perché quando bisogna accendere una caldaia o una pompa di calore occorre mantenerne chiuso il comando di attivazione fin quando si vuole che riscaldi. Comunicare un comando del genere via radio implica il dover fare una scelta: trasmettere costantemente fin quando il termostato, rilevando il raggiungimento della temperatura voluta, non apre il proprio contatto; inviare i soli comandi di accensione e spegnimento. Nel primo caso si va incontro a due problemi sostanziali: si occupa continuamente il canale radio e il ricevitore può, non ricevendo il comando per la concomitante presenza di trasmissioni di altri apparati operanti sullo stesso canale, disattivare la propria uscita e con essa la caldaia; si spende

molta energia e, nel caso il trasmettitore funzioni a pile, si è costretti a frequenti sostituzioni. Il secondo propone una soluzione poco affidabile: infatti se, per qualche ragione, il ricevitore perde il comando di attivazione, la caldaia resta spenta; se poi non riceve quello di spegnimento, rimane sempre accesa, con le conseguenze del caso.

Tra le due soluzioni abbiamo scelto la prima, adottando però accorgimenti finalizzati ad ottimizzare le prestazioni e minimizzare le problematiche appena esposte: in altre parole trasmettiamo non continuamente ma periodicamente lo stato del termostato, cosicché se il ricevitore perde una trasmissione, recupera con quella successiva. L'invio periodico della condizione dell'ingresso del trasmettitore (contatto del termostato) rappresenta il miglior compromesso perché, giocando opportunamente sull'intervallo fra una trasmissione e la seguente, si riesce a trovare il miglior compromesso tra prontezza della risposta e consumo di energia. Vediamo dunque come funziona il nostro sistema, analizzando le due unità che lo compongono.

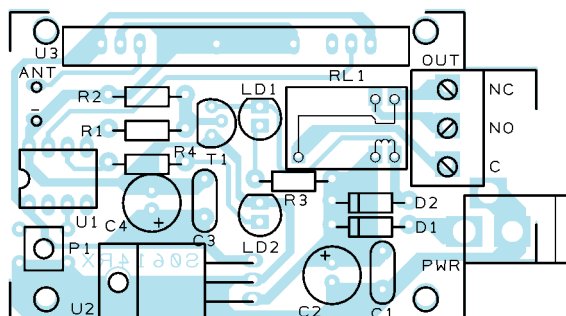
Il trasmettitore

La prima è la trasmittente; in queste pagine potete vederne lo schema elettrico, che ne illustra la struttura: il tutto si riduce a un microcontrollore e a un modulo TX UHF dell'Aurel. Dopo l'inizializzazione degli I/O il programma del micro testa continuamente la linea GP5 per verificarne il livello logico: quando i punti IN sono isolati, il resistore di pull-up settato internamente dal software determina lo stato alto, mentre se il termostato collegato al circuito chiude i predetti ingressi, la GP5 vede lo zero logico. In entrambi i casi il microcontrollore prevede la trasmissione periodica dei dati inerenti alla condizione dell'ingresso e quindi del ➤



ELENCO COMPONENTI:

R1: 4,7 kohm	U1: PIC12F675	Varie:
R2: 10 kohm	(MF614RX)	- Zoccolo 4+4
R3: 1 kohm	U2: 7805	- Morsettiera 3 poli
R4: 330 ohm	U3: RX4M50FM60	- Plug alimentazione
C1: 100 nF multistrato	D1, D2: 1N4007	- Vite 8 mm 3 MA
C2: 220 µF 25 VL	T1: BC547	- Dado 3 MA
C3: 100 nF multistrato	LD1: led 5 mm rosso	- Circuito stampato
C4: 220 µF 16 VL	LD2: led 5 mm verde	codice S0614RX
	P1: Microswitch	
	RL1: Relè 12V	

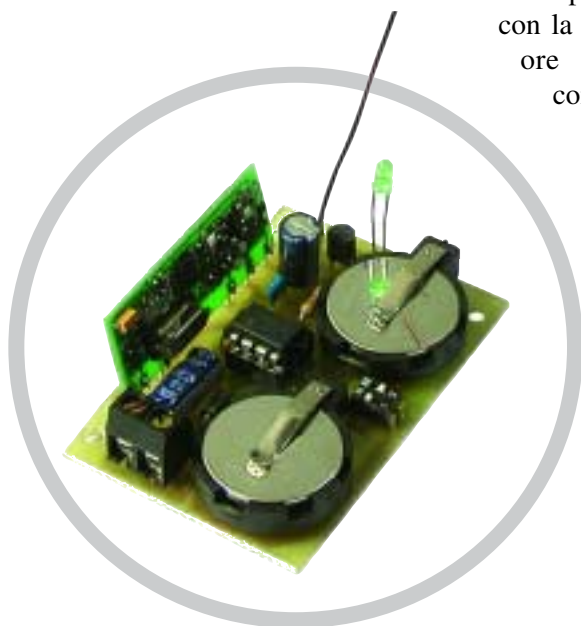




Il radiotermostato consta di due unità: una trasmittente, controllata dal contatto del termostato vero e proprio, e l'altra ricevente; quest'ultima è provvista di un relé del quale è reso disponibile l'intero scambio, così da comandare sia caldaie che condizionatori d'aria.

termostato, verso il ricevitore; le modalità vengono definite dalle impostazioni dei dip-switch, ovvero dalle condizioni logiche delle linee GP1 e GP3. Di queste ultime note un dettaglio: mentre per GP3 è possibile impostare il resistore di pull-up interno, per GP1 il PIC12F675 non lo permette; è stato quindi necessario collegare la R3

L'unità trasmittente del termostato via radio è alimentata con due pile a bottone del tipo CR2032.



fra il piedino 4 dell'U1 e il positivo di alimentazione (batterie). Le modalità di trasmissione dello stato del termostato (e quindi della caldaia) sono quattro: la prima prevede l'invio del comando ad ogni variazione della condizione logica dell'ingresso e si ottiene con i dip entrambi aperti (vedi tabella); il ricevitore esegue immediatamente il comando ricevuto.

La seconda (dip 1 aperto e dip 2 chiuso) combina la trasmissione in corrispondenza della variazione con la ripetizione periodica ogni 3 ore della condizione dell'IN;

come nel caso precedente il ricevitore non fa che seguire i comandi in arrivo. La terza modalità (dip 1 chiuso e dip 2 aperto) è come la precedente ma accorcia l'intervallo di ripetizione riducendolo ad 1 ora; nel ricevitore è però previsto un meccanismo di sicurezza che consente di aprire lo scambio del relé e disattivare la caldaia nel caso non venga rinnovata entro un'ora e un quarto

l'ultima condizione trasmessa. La quarta ed ultima modalità (entrambi i dip chiusi) prevede la solita trasmissione in concomitanza con la variazione e il rinnovo periodico ogni 3 ore, ovviamente in mancanza di variazioni nel frattempo; il ricevitore esegue i comandi di variazione e, sempre per il solito meccanismo di sicurezza, se dall'ultima variazione o trasmissione del TX non riceve conferme entro 3 h+15" disattiva il relé.

Come tutti i sistemi di comando a distanza, sebbene sia stato dimensionato per operare nell'ambito di poche decine di metri, il nostro prevede una codifica che ha lo scopo di impedire al trasmettitore di disturbare altri sistemi operanti sulla stessa frequenza ed al ricevitore di venire attivato da apparati estranei; inoltre, per evitare che due coppie TX/RX installate vicine tra loro (ad esempio in due villette a schiera attigue) interferiscano l'una con l'altra, è stato previsto di poter definire più codici, scegliendo tra cinque combinazioni, che sembrano poche ma in realtà sono adatte all'impiego: è infatti difficile che nello stesso raggio d'azione vengano installati più di due o tre termostati UHF.

Chi ha dato uno sguardo allo schema elettrico ha certo notato che non vi sono a disposizione abbastanza elementi per l'impostazione della codifica e delle modalità di connessione; dunque, qual è il segreto? Ebbene, l'impostazione si fa nell'istante in cui si alimenta il circuito: dopo la sequenza di rapidi lampeggi, il led LD1 lampeggia lentamente per circa 5 secondi al ritmo di 2 secondi acceso e 1 spento; per avviare la programmazione dell'indirizzo identificativo del sistema bisogna cambiare lo stato del dip 1 durante il lampeggio lento. A questo punto il led compie una nuova sequenza di lampeggi rapidi alternati allo spegnimento; il tutto per

cinque volte. Per definire l'address bisogna riportare dip 1 nella posizione originaria quando il led ha compiuto un numero di periodi di rapido lampeggio e spegnimento uguale al numero voluto: ad esempio, volendo assegnare l'indirizzo tre bisogna attendere che per tre volte LD1 lampeggi rapidamente e si spenga. Riportato dip 1 nella condizione in cui si trovava al momento dell'accensione del circuito, il software conferma l'impostazione mediante una sequenza di lampeggio (del solito LD1) lenta.

Notate che ad ogni accensione, se non si accede alla procedura di impostazione, dopo i rapidi lampeggi iniziali e il lampeggio 2/1 secondi, il circuito mostra l'attuale indirizzo: per vederlo bisogna lasciare i dip come sono, senza toccarne alcuno, attendendo il termine del lampeggio lento 2/1 secondi; a un certo punto LD1 pulsa per il numero di volte (max 5) che corrisponde all'address impostato. Vi accorgete di ciò nel momento in cui cambierete le pile del trasmettitore.

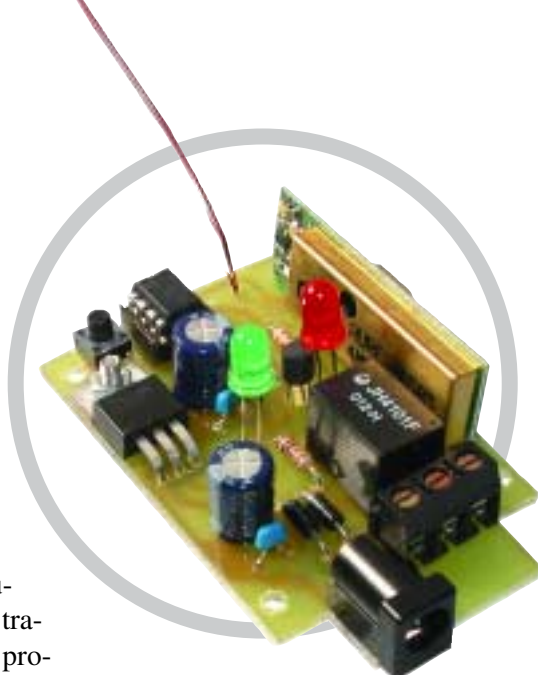
Bene, questo è quel che riguarda il funzionamento del microcontrollore. Dobbiamo aggiungere che ad ogni trasmissione il TX emette, sotto forma di treno di impulsi 0/3 V, i dati corrispondenti al comando e all'indirizzo, accompagnati da informazioni necessarie a contraddistinguere la trasmissione. Ogni stringa emessa ha il seguente formato: inizia con header, composto da una serie di lettere U maiuscole espresse ciascuna dal corrispondente binario del relativo valore della tabella standard ASCII, cui segue */; poi viene il comando, scritto sotto forma di testo; anche in questo caso ciascuna lettera è espressa dal proprio valore ASCII (in formato binario). Segue la funzione, sotto forma di numero (1, 2, 3, 4) che corrisponde alla modalità di comando: a riguardo si veda la

tabella illustrata in queste pagine; dopo il numero di modalità viene messo il codice della coppia (1÷5) quindi la stringa termina con una serie di lettere A maiuscole, espresse, come al solito, dai valori binari che corrispondono ai rispettivi ASCII delle lettere A. Gli impulsi corrispondenti ai bit componenti ciascuna stringa sono emessi dal piedino 3 e da lì inviati al 2 del modulo ibrido U2: quest'ultimo è un trasmettitore in tecnologia SMD prodotto dall'Aurel, che l'ha progettato esplicitamente per funzionare in circuiti a bassa tensione, alimentati con 2,7÷5 V. Il componente dispone di un oscillatore controllato da un PLL e un quarzo, che gli permettono di emettere una portante radio a 433,92 MHz esatti e stabili; la potenza dell'oscillatore è di 10 dBm. L'onda UHF viene emessa dal piedino d'antenna (11) quando la linea di ingresso (2) viene sottoposta a un livello logico TTL alto; a zero l'oscillatore viene spento. Al pin d'antenna può essere collegato anche un semplice spezzone di filo in rame lungo 17 cm, che garantisce una copertura di 50÷60 metri. Comunque nulla vieta di connettere al punto ANT un'antenna da 50 ohm d'impedenza.

Il ricevitore

Descritta l'unità trasmittente passiamo alla ricevente e vediamo cosa accade in essa quando riceve una stringa di dati in arrivo dal TX; partiamo dallo schema elettrico, che ci mostra la struttura del circuito, composto essenzialmente da un modulo ibrido radoricevente e da un microcontrollore.

Dopo l'inizializzazione degli I/O il software che gira in quest'ultimo controlla periodicamente la condizione logica delle linee GP0 e GP4: con la prima rileva l'arrivo di dati



dal radoricevitore e con la seconda controlla lo stato del pulsante che l'utente usa per forzare la procedura di autoapprendimento. Andiamo con ordine partendo dall'antenna: collegata al piedino 3, porta il segnale RF captato all'input dell'U3, un RX-4M50FM60SF Aurel, provvisto di stadio ricevente supereterodina (a conversione di frequenza) ad alta sensibilità (-111 dBm) e selettività (600 kHz a -3 dB) e di squadratore del segnale demodulato, reso disponibile sul piedino 14.

Quando rileva una commutazione sulla linea GP0, il microcontrollore ritiene sia giunta una stringa inviata dal trasmettitore: la trasferisce in RAM e la analizza per verificare innanzitutto che abbia il formato adottato nel sistema; se è così procede ed estrae l'address per verificare che il comando provenga dal TX abbinato, altrimenti abortisce la procedura e torna in attesa di una nuova commutazione su GP0 o GP4. Nel caso la stringa risponda al protocollo adottato dal sistema e provenga dal trasmettitore abbinato, viene decifrato il comando e letta la modalità, costantemente trasmessa ed aggiornata.

Possono quindi verificarsi due situazioni: lo stato trasmesso è diverso da quello in cui attualmente ➤

Il protocollo di comunicazione

Ogni trasmissione del TX collegato al termostato invia una stringa di dati il cui formato è sempre lo stesso e viene interpretato dal microcontrollore del ricevitore che conosce l'esatta sintassi; il formato della stringa è il seguente:

header (ASCII) - */ - comando - funzione - indirizzo - terminatore

L'header è la testa, ossia la parte iniziale che contraddistingue la stringa; è costituita da una serie di lettere *U* maiuscole espresse ciascuna con il valore binario corrispondente al loro equivalente ASCII, ossia 85 (decimale). **/* è un parametro fisso, espresso, al solito, con numeri binari corrispondenti ai valori ASCII dei rispettivi simboli (42 per l'asterisco e 47 per la barra). *Comando* può essere *ACCESO* o *SPENTO* e viene inviato sotto forma di testo, composto con i caratteri espressi ciascuno dal proprio valore ASCII; *funzione* è un numero pari a 1, 2, 3, 4, a seconda dell'impostazione dei dip della trasmittente: corrisponde alla modalità scelta e viene trasmesso sempre, allo scopo di aggiornare le impostazioni della ricevente, consentendole di recuperare la condizione dell'uscita anche in caso di blackout. Il parametro *Indirizzo* è l'address dell'unità trasmittente e serve alla ricevente in apprendimento per abbinarsi al trasmettitore e, nel normale esercizio, per verificare se il comando arriva dal TX abbinatogli. *Terminatore* è la coda del messaggio ed è composto da una serie di lettere *A* maiuscole, espresse ciascuna dal corrispondente binario del valore corrispondente (65 decimale) nella tabella standard dei caratteri ASCII. Quando riceve una stringa, il microcontrollore posto nel ricevitore verifica innanzitutto che inizi con la serie di lettere *U* maiuscole, altrimenti non procede neppure e si ridispone ad attendere nuove stringhe dall'uscita del ricevitore ibrido.

Notate che in ogni trasmissione il TX invia sempre tutti i parametri significativi: questo modo di funzionamento è stato voluto per semplificare le procedure (la trasmissione serve infatti sia per comandare il ricevitore, che per aggiornare lo stato del relé o effettuare l'abbinamento TX/RX mediante la procedura di apprendimento).

si trova il relé; in questo caso la linea GP2 viene impostata al fine di porre il relé nella condizione richiesta. Se invece il comando mira a dare la stessa impostazione attuale, viene ignorato: l'uscita che comanda il relé rimane nella stessa condizione.

Lo scambio di RL1 agisce da contatto normalmente aperto di attivazione della caldaia: quando viene ricevuto il comando di attivazione (punti IN del trasmettitore chiusi) il micro porta GP2 a livello logico alto, manda in saturazione T1 e lo fa chiudere; invece, in corrispondenza del comando di disattivazione (apertura dei contatti IN del trasmettitore), GP2 viene posta a zero e lascia interdire T1, facendo ricadere l'equipaggio mobile del relé ed aprire OUT ed NO.

Il programma che gira nel PIC è fatto in modo da mantenere il

comando ricevuto fino alla ricezione di un comando opposto, ovvero, nelle modalità che lo prevedono, allo scadere del tempo limite entro il quale attendere dal trasmettitore la stringa di conferma.

Ciò significa che se il relé viene eccitato, resta attivo fin quando il TX non invia la stringa contenente il comando opposto (*SPENTO*) ovvero, nelle modalità 3 e 4, fino allo scadere del tempo previsto (1h+15" per la 3 e 3h+15" per la modalità 4) senza che sia giunta la stringa di conferma periodica.

Quanto detto riguarda GP0; se invece l'evento rilevato riguarda la commutazione di GP4, significa che l'utente ha premuto il pulsante col quale richiede l'avvio della procedura di abbinamento TX/RX: premendo il tasto, il PIC fa lampeggiare 5 volte LD2, che poi mantiene acceso a luce fissa fin quando dal

trasmettitore si invia una stringa, per forzare la trasmissione della quale basta far cambiare lo stato dell'ingresso IN (aprirlo se è chiuso o chiuderlo se è aperto). Quando arrivano i dati, il micro li trasferisce in RAM e li analizza per verificare innanzitutto che abbiano il formato adottato nel sistema; se è così procede ed estrae l'indirizzo, che memorizza in EEPROM per verificare le stringhe che giungeranno nel normale esercizio ed accertare che i comandi provengano dal TX abbinato, altrimenti abortisce la procedura e torna in attesa di una nuova commutazione su GP0 o GP4. Quando viene salvato l'address, LD2 prima pulsa altre cinque volte, ma rapidamente, poi si spegne; tale segnalazione conferma l'acquisizione del codice e l'abbinamento del ricevitore con il trasmettitore che ha appena effettuato la trasmissione.

Detto questo dobbiamo far notare che il circuito dispone di due led: LD1 si accende a luce fissa quando il relé (e con esso l'uscita di comando della caldaia) è attivo e si spegne quando è a riposo; invece LD2, oltre che nel modo descritto riguardo all'autoapprendimento del codice, nel normale esercizio pulsa all'arrivo di una stringa dal TX.

L'intera ricevente è alimentata con una tensione continua o alternata di 12 o 9 V, applicata ai punti + e -PWR: D1, che nel primo caso funziona da protezione contro l'inversione di polarità, nell'alimentazione in alternata raddrizza a singola semionda la sinusoide.

In tutti i casi, ai capi dei condensatori C1 e C2 abbiamo una tensione continua che alimenta la bobina del relé; tramite il regolatore U2, da tale tensione ricaviamo 5 volt stabilizzati che servono a far funzionare il PIC12C675 e il ricevitore Aurel.

Realizzazione pratica

Passiamo adesso alle note costruttive

ve, dicendo innanzitutto che bisogna preparare due circuiti stampati: uno per la trasmittente e l'altro per la ricevente; entrambi sono piccoli e di semplice realizzazione, visto che sono del tipo a singola ramatura. Le rispettive tracce potete scaricarle gratuitamente dal sito della rivista: www.elettronica.in.it.

Per il montaggio non vi sono che le solite raccomandazioni: procedere in ordine di altezza, riferirsi ai disegni di montaggio prima di inserire i componenti polarizzati, disporre gli integrati dip su zoccoli. L'unità trasmittente funziona a pile: più esattamente, due elementi a bottone CR2032. Montate quindi sullo stampato corrispondente due clip portatile adatte allo scopo.

Nel trasmettitore prevedete una morsettiera bipolare a passo 5 mm da usare per agevolare le connessioni con il contatto del termostato; quanto alla ricevente, una morsettiera tripolare a passo 5 mm agevo-

lerà i collegamenti con il contatto di attivazione della caldaia. A riguardo precisiamo che solitamente caldaie e pompe di calore si azionano chiudendo il loro contatto di attivazione, quindi dovete usare i morsetti corrispondenti ai punti C ed NO dello scambio di RL1; se invece dovete comandare un apparato che si attiva aprendo il rispettivo contatto, collegate C ed NC.

Questi ultimi vanno impiegati anche quando si intende usare un termostato a singolo contatto per gestire a distanza un condizionatore d'aria: infatti i termostati da riscaldamento aprono il contatto quando la temperatura, salendo, raggiunge quella impostata e lo chiudono al disotto di quest'ultima; dunque, il relé si chiuderà non appena l'ambiente diventerà abbastanza fresco, quindi usando lo scambio C/NC il condizionatore verrà disattivato proprio appena il termostato, rilevando che la temperatura dell'am-

biente si è abbassata al livello voluto, aprirà il proprio contatto.

Sempre riguardo al relé, dobbiamo rammentarvi che il suo scambio può sopportare una tensione massima di 250 Vac e una corrente dell'ordine di 1 ampere; usatelo quindi per controllare il contatto di attivazione della caldaia o del condizionatore d'aria. Sconsigliamo di usare RL1 per interrompere l'alimentazione di tali dispositivi, che, normalmente, assorbono anche più di quello che lo scambio può sopportare; in particolare, condizionatori d'aria e termoconvettori assorbono correnti dell'ordine dei 5÷10 ampere. Per rendere quanto più possibile autonomo il gruppo del termostato, la trasmittente è alimentata con due pile a bottone, che avrete cura di inserire nelle clip tenendone il polo positivo rivolto verso l'esterno. Il ricevitore, essendo collocato vicino alla caldaia e quindi in un punto nel quale sia

IDEA ELETTRONICA: accendiamo le tue idee!

ROBOSAPIEN 2



Il giocattolo si evolve: parla, cammina, balla, gioca, raccoglie e lancia oggetti e fa persino mosse di kung fu. Si chiama Robosapien V2, è rappresentativo della nuova generazione di robot per il divertimento domestico, presentati in anteprima al CES di Las Vegas. Dotato di videocamera e radar a infrarossi, il robotino alto 60 cm è in grado anche di riconoscere le persone, interagendo con loro.

€ 299,00

MINI ROBOSAPIEN - edizione limitata -



Il robot più piccolo dei Robosapien! Questo piccolo robot ha funzionalità limitate rispetto al grande Robosapien, ma è fantastico per gli appassionati di Robosapien. Misura appena 18 centimetri di altezza e può camminare. I suoi caratteristici occhi neri si illuminano mentre cammina. Le pinze di presa funzionano con i tasti meccanici ed i relativi bracci possono essere spostati su e giù usando gli scissor sulla parte posteriore delle sue spalle. Il mini Robosapien utilizza due batterie AAA.

BIANCO € 20,00 - ARGENTO/BLU/ORO € 23,00

ROBOSAPIEN COLORATI

- edizione limitata -



Robosapien è un mini Robot Android progettato da Mark Tilden, creatore dei Robot B-2 e degli esafacetti B.O. Bug. Robosapien compie Movimenti e gesti fluidi, cammina veloce e dinamica a due velocità; braccia completamente funzionali con due tipi di presa delle mani. 67 funzioni pre-programmate: presa, lancio, calcio, danza, kung-fu, aerobica, imitazione, rap e molte altre ancora. 4 modalità di programmazione. 3 modalità dimostrative.

€ 125,00

MINI ROBOT PICAXE



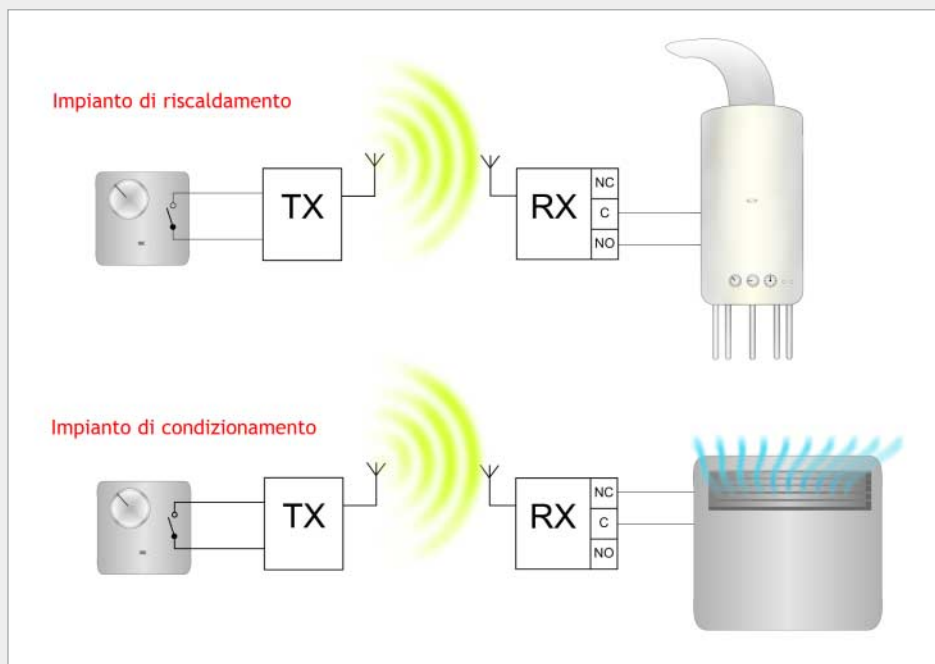
Il Microbot Picaxe consiste in una base in plastica di 120 X 80mm che include 2 motorini (rapporto 42:1) con relative ruote in gomma e ospita 4 batterie abbo, una scheda che è composta da un PICAXE 18 (espandibile con un PICAXE180) un chip che genera automaticamente gli impulsi PWM per comandare i motori e un chip di potenza che pilota i due motori (3.253C). La funzione PWM è molto interessante in quanto permette di modulare la velocità dei motori in modo molto graduale, senza impiegare il PICAXE, lasciando quindi spazio al programma di gestione dei sensori e della logica del Robot. Sono disponibili ben 5 input per collegare vari sensori e 4 Output. La scheda è, inoltre, provvista del comodo connettore da 3.5mm stereo, per la programmazione seriale, in questo modo tutte le operazioni di programmazione e debug avvengono in modo rapido e sicuro.

€ 84,00

Tutti i prezzi si intendono IVA compresa. Per ordini e informazioni:
IDEA ELETTRONICA - Via San Vittore n°24/A - 21040 Oggiona con S. Stefano - Varese - ITALY - Tel./Fax 0331/212723
Visitate il nostro sito: WWW.IDEAETTRONICA.IT

I contatti di uscita

L'unità ricevente del sistema di controllo a distanza è predisposta per comandare l'attivazione di caldaie a gas, pompe di calore e sistemi di condizionamento d'aria: dispone infatti di un relé del quale viene reso disponibile l'intero scambio, così da adattarsi a tutte le possibili situazioni; lo scambio va utilizzato in base al tipo di termostato che si vuole abbinare al trasmettitore. Vediamo prima il tipo a singolo contatto: siccome si chiude sotto la temperatura impostata e si apre al suo superamento, del relé sul ricevitore bisogna usare il contatto normalmente chiuso (C/NC) quando si intende gestire un condizionatore d'aria e il normalmente aperto se si vuole comandare una caldaia. Usando un termostato a doppio contatto, del relé del ricevitore si può invece usare il solo scambio normalmente aperto: basta connettere all'IN del trasmettitore il contatto del freddo (quello che si chiude quando la temperatura sale sotto il valore impostato) se si intende gestire un condizionatore d'aria o quello del caldo (il contatto che si chiude se la temperatura scende al di sotto del valore impostato) se occorre comandare una caldaia o pompa di calore.



Il relé da noi montato nell'unità ricevente è del tipo per alta tensione ma può commutare correnti relativamente deboli: dell'ordine di 1 ampere. Ciò significa che può essere usato

per chiudere il contatto di attivazione (quello riservato al termostato elettromeccanico o elettronico) ma non la linea di alimentazione principale, dalla quale, specie il condizionatore, assorbe ben oltre quello che viene consentito dallo scambio del RL1. Per sua natura il nostro sistema di trasferimento a distanza del comando termostatico è adatto ad essere abbinato a tutti i termostati disponibili sul mercato, siano essi meccanici o elettronici, purché provvisti di contatto pulito.

accessibile l'alimentazione elettrica della rete a 220 V, va equipaggiato con un alimentatore capace di fornire 12÷15 V e una corrente continua di almeno 150 mA.

In commercio si trovano alimentatori universali "a cubo" con spina

integrata e cavetto terminante con il tradizionale plug: per questo il nostro ricevitore monta una presa del genere.

Utilizzate un alimentatore che abbia il plug del diametro (interno/esterno) corretto e polarità posi-

tiva verso l'interno. Alimentate le due unità, provvedete alla verifica del funzionamento del sistema forzando la trasmissione da parte del TX (basta variare la condizione dell'ingresso) dopo aver premuto il P1 del ricevitore.

Per il

MATERIALE

Tutti i componenti relativi a questo progetto sono facilmente reperibili in commercio. I master dei due stampati ed il firmware dei due microcontrollori sono scaricabili gratuitamente dal sito della rivista (www.elettronica.in.it). Il modulo trasmettente TX4MAVPF10 costa 13,00 Euro mentre quello ricevente RX4M50FM60 costa 24,00 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>**



CENTRO FIERA DEL GARDA
MONTICHIARI (BS)
PROVINCIA DI BRESCIA

11 - 12 MARZO 2006

26^a MOSTRA NAZIONALE MERCATO RADIANTISTICO

- | | |
|--|---|
| ☐ Elettronica | ☐ Hi-Fi |
| ☐ Video | ☐ Computer |
| ☐ Strumentazione | ☐ Esposiz. Radio d'Epoca |
| ☐ Componentistica | ☐ Filatelia |

12^o RADIOMERCATINO
di PORTOBELLO

ORARIO CONTINUATO:

SABATO 9,00 - 18,30 - DOMENICA 9,00 - 17,30

Multimetri e strumenti di misura

Multimetro da banco



Multimetro professionale da banco con alimentazione a batteria/rete, indicazione digitale e analogica con scala a 42 segmenti, altezza digit 18 mm, selezione automatica delle portate, retroilluminazione e possibilità di connessione ad un PC. Funzione memoria, precisione $\pm 0,3\%$.

DVM645 Euro 196,00

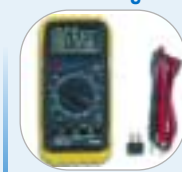
Multimetro digitale a 3 1/2 con LC



Apparecchio digitale a 3/2 cifre con eccezionale rapporto prezzo/prestazioni. 39 gamme di misurazione: tensione e corrente DC, tensione e corrente AC, resistenza, capacità, induttanza, frequenza, temperatura, tester TTL. Alimentazione con batteria a 9V.

DVM1090 Euro 64,00

Multimetro digitale RMS a 4 1/2 cifre



Strumento professionale con 10 differenti funzioni in 32 portate. Misurazione RMS delle componenti alternate. Ampio display a 4 1/2 cifre. È in grado di misurare tensioni continue e alternate, correnti AC e DC, resistenza, capacità, frequenza, continuità elettrica nonché effettuare test di diodi e transistor. Alimentazione con batteria a 9V. Completo di guscio di protezione.

DVM98 Euro 115,00

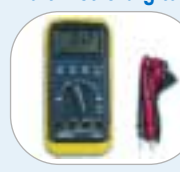
Multimetro digitale a 3 1/2 cifre con RS232



Multimetro digitale dalle caratteristiche professionali a 3/2 cifre con uscita RS232, memorizzazione dei dati e display retroilluminato. Misura tensioni in AC e DC, correnti in AC e DC, resistenza, capacità e temperatura. Alimentazione con batteria a 9V. Completo di guscio di protezione.

DVM345 Euro 72,00

Multimetro digitale a 3 3/4 cifre



Strumento professionale con display LCD da 3 3/4 cifre, indicazione automatica della polarità, bargraph, indicazione di batteria scarica, selezione automatica delle portate, memorizzazione dei dati e protezione contro i sovraccarichi. Misura tensioni/correnti alternate e continue, resistenza, capacità e frequenza. Alimentazione con batteria a 9V. Completo di guscio di protezione.

DVM68 Euro 47,00

Multimetro analogico



Multimetro analogico per misure di tensioni DC e AC fino a 1000V, correnti in continua da 50 μ A a 10A, portate resistenza ($\times 1$ a $\times 10K$), diodi e transistor (Ice0, hfe); scala in dB; selezione manuale delle portate; dimensioni: 148 x 100 x 35mm; alimentazione: 9V (batteria inclusa).

AVM360 Euro 14,00

LC meter digitale a 3 1/2 cifre



Strumento digitale in grado di misurare con estrema precisione induttanze e capacità. Display LCD con cifre alte 21mm, 6 gamme di misura per capacità, 4 per induttanza. Autocalibrazione, alimentazione con pila a 9V.

DVM6243 Euro 80,00

Multimetro con pinza amperometrica



Dispositivo digitale con pinza amperometrica. Display digitale a 3200 conteggi con scala analogica a 33 segmenti. Altezza digit 15 mm, funzione di memoria. È in grado di misurare correnti fino a 1.000 A. Massimo diametro cavo misurazione: $\varnothing$ 50 mm. Misura anche tensione, resistenza e frequenza. Funzione continuità e tester per diodi. Dotato di retroilluminazione. Alimentazione con batteria a 9V.

DCM268 Euro 118,00

Pinza amperometrica per multimetri digitali



Pinza amperometrica adatta a qualsiasi multimetro digitale. In grado di convertire la corrente da 0,1 a 300 A in una tensione di 1 mV ogni 0,1 A misurati. Adatto per conduttori di diametro massimo di 30mm. Dimensioni: 80 x 156 x 35mm; peso con batteria: $\pm 220g$.

AC97 Euro 25,00

Multimetro analogico



Multimetro analogico per misure di tensioni DC e AC fino a 1000V, correnti in continua da 50 μ A a 10A, portate resistenza ($\times 1$ a $\times 10K$), diodi e transistor (Ice0, hfe); scala in dB; selezione manuale delle portate; dimensioni: 148 x 100 x 35mm; alimentazione: 9V (batteria inclusa).

AVM460 Euro 11,00

Multimetro analogico con guscio giallo



Display con scale colorate. Per misure di tensioni DC e AC fino a 500V, corrente in continua fino a 250mA, e manopola di taratura per le misure di resistenza ($\times 1$ a $\times 10$). Selezione manuale delle portate; dimensioni: 120 x 60 x 30mm; alimentazione: 1,5V AA (batteria compresa). Completo di batteria e guscio di protezione giallo.

AVM460 Euro 11,00

Multimetro miniatura con pinza



Pinza amperometrica con multimetro digitale con display LCD retroilluminato da 3 2/3 cifre a 2400 conteggi. Memorizzazione dei dati, protezione contro i sovraccarichi, autospegnimento e indicatore di batteria scarica. Misura tensioni/correnti alternate e continue 0-200A e frequenza 40Hz-1kHz; apertura pinza: 18mm (0,7); torcia incorporata. Alimentazione con 2 batterie tipo AAA 1,5V. Viene fornito con custodia in plastica.

DCM269 Euro 86,00

Multimetro digitale a 3 1/2 cifre low cost



Multimetro digitale in grado di misurare correnti fino a 10A DC, tensioni continue e alternate fino a 750V, resistenza fino a 2 Mohm, diodi, transistor. Alimentazione con batteria a 9V (inclusa). Dimensioni: 70 x 126 x 26 mm.

DVM830L Euro 4,50



Luxmetro digitale

Strumento per la misura dell'illuminazione con indicazione digitale da 0,01lux a 5000lux tramite display a 3 1/2 cifre. Funzionamento a batterie, indicazione di batteria scarica, indicazione di fuoriscalda. Sonda con cavo della lunghezza di circa 1 metro. Alimentazione: 1 x 9V (batteria inclusa). Completo di custodia.

DVM1300 Euro 48,00

Rilevatore di temperatura a distanza -20/+270°C



Sistema ad infrarossi per la misura della temperatura a distanza. Possibilità di visualizzazione in gradi centigradi o in gradi Fahrenheit, display LCD con retroilluminazione, memorizzazione, spegnimento automatico. Puntatore laser incluso. Alimentazione: 9V (batteria inclusa).

DVM8810 Euro 98,00

Rilevatore di temperatura a distanza -20/+420°C



Sistema ad infrarossi per la misura della temperatura a distanza. Possibilità di visualizzazione in °C o °F. Puntatore laser incluso. Alimentazione: 9V.

DVM8869 Euro 178,00

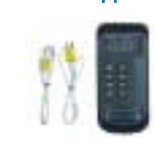
Termometro IR con lettura a distanza



Possibilità di visualizzazione in °C o °F, display LCD con retroilluminazione, memorizzazione, spegnimento automatico, puntatore a led. Gamma di temperatura da -20°C a +270°C. Rapporto distanza/spot: 6/1. Alimentazione: 2 x 1,5V (2 batterie minitilo AAA, comprese).

DVM77 Euro 56,00

Termometro con doppio ingresso e sensore a termocoppia



Strumento professionale a 3 1/2 cifre per la misura di temperature da -50°C a 1300°C munito di due distinti ingressi. Indicazione in °C o °F, memoria, memoria del valore massimo, funzionamento con termocoppia tipo K. Lo strumento viene fornito con due termocoppie. Alimentazione: 1 x 9V.

DVM1322 Euro 69,00

Termometro digitale da pannello



Termometro digitale da pannello con sensore via cavo lungo 1,5 metri. Facile da installare, con ampio display e completo di contenitore in ABS. Intervallo di misurazione della temperatura: -50°C ~ +70°C; tolleranza: 1°C; dimensione display: 12 x 6,5mm; lunghezza sensore via cavo: 1,5 metri; dimensioni: 47 x 26 x 13mm; alimentazione: 1 x LR44 (batteria a bottone inclusa).

PMTEMP Euro 14,00

Termometro digitale interno / esterno



Termometro digitale con indicazione contemporanea della temperatura interna e esterna in °C o °F. Ideale per controllare la temperatura di frigoriferi, freezer, ma anche per misurare la temperatura ambiente. Montaggio a muro o su supporto.

Doppio con sensore per temperatura esterna a tenuta stagna; display di facile lettura; allarme; memoria di minima e massima; gamma temperatura interna: -10°C / +50°C (+14°F / +122°F); gamma temperatura esterna: -50°C / +70°C (-58°F / +158°F); dimensioni termometro: 110 x 70 x 20mm; alimentazione: 1 x 1,5 V AAA (batteria compresa).

TA20 Euro 5,00

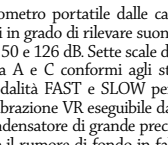
Termoigrometro digitale



Termoigrometro digitale per la misura del grado di umidità (da 0% al 100%) e della temperatura (da -20°C a +60°C) con memoria ed indicazione del valore minimo e massimo. Alimentazione 9V (a batteria).

DVM321 Euro 78,00

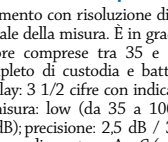
Fonometro analogico



Fonometro portatile dalle caratteristiche professionali in grado di rilevare suoni di intensità compresa tra 50 e 126 dB. Sette scale di misura, curve di pesatura A e C conformi agli standard internazionali, modalità FAST e SLOW per le costanti di tempo, calibrazione VR eseguibile dall'esterno, microfono a condensatore di grande precisione. Ideale per misurare il rumore di fondo in fabbriche, scuole e uffici, per testare l'acustica di studi di registrazione e teatri nonché per effettuare una corretta installazione di impianti HI-FI. L'apparecchio viene fornito con batteria alcalina.

FR255 Euro 26,00

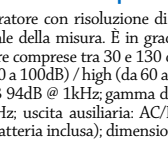
Fonometro professionale



Strumento con risoluzione di 0,1 dB ed indicazione digitale della misura. È in grado di rilevare intensità sonore comprese tra 35 e 130 dB in due scale. Completo di custodia e batteria di alimentazione. Display: 3 1/2 cifre con indicatore di funzione; scale di misura: low (da 35 a 100dB) / high (da 65 a 130dB); precisione: 2,5 dB / 3,5 dB; definizione: 0,1 dB; curve di pesatura: A e C (selezionabile); alimentazione: 9V (batteria inclusa).

DVM1326 Euro 122,00

Fonometro professionale



Misuratore con risoluzione di 0,1 dB ed indicazione digitale della misura. È in grado di rilevare intensità sonore comprese tra 30 e 130 dB. Scale di misura: low (da 30 a 100dB) / high (da 60 a 130dB); precisione: $\pm 1,5dB$ 94dB @ 1kHz; gamma di frequenza: da 31,5Hz a 8kHz; uscita ausiliaria: AC/DC; alimentazione: 1 x 9V (batteria inclusa); dimensioni: 210 x 55 x 32 mm.

DVM805 Euro 92,00



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 - Richiedi il Catalogo Generale!

Anemometro digitale



Dispositivo per la visualizzazione della velocità del vento su istogramma e scala di Beaufort completo di termometro. Visualizzazione della temperatura di raffreddamento (wind-chill factor). Display LCD con retroilluminazione. Strumento indispensabile per chi si occupa dell'installazione o manutenzione di sistemi di condizionamento e trattamento dell'aria, sia a livello civile che industriale. Indispensabile in campo nautico. Completo di inghietta. Alimentazione: 1x 3 V (CR2032, batteria inclusa).

WS9500 Euro 39,00

Multimetro digitale a 3 1/2 cifre

Multimetro digitale con display retroilluminato in grado di misurare correnti fino a 10A DC, tensioni continue e alternate fino a 600V, resistenza fino a 2 Mohm, diodi, transistor e continuità elettrica. Alimentazione con batteria a 9V (inclusa). Funzione memoria per mantenere visualizzata la lettura. Completo di guscio di protezione.

DVM850 Euro 12,00



a cura di Carlo Tauraso

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 terza puntata entriamo nel vivo dello sviluppo del modulo CAN e analizziamo l'hardware che useremo nei nostri esperimenti.



Nella precedente puntata del nostro corso vi abbiamo mostrato in anteprima lo schema elettrico della demoboard per le applicazioni CAN e annunciato l'imminente pubblicazione delle nozioni inerenti alla progettazione e alla messa in esercizio di nodi CAN. Eccoci dunque, puntuali, ad affrontare l'argomento annunciato il mese scorso.

Librerie CAN C18

Per sviluppare correttamente il firmware per i nostri nodi CAN dobbiamo innanzitutto considerare il fatto che Microchip rende disponibile un'interessante libreria C18, peraltro disponibile e scaricabile liberamente dal sito della

rivista. La presenza di una serie di funzioni che nascondono opportunamente i dettagli implementativi del protocollo CAN facilita notevolmente il lavoro del programmatore, che può concentrarsi maggiormente sulla logica della propria applicazione. Il pacchetto che prendiamo in considerazione è contenuto in un archivio compresso chiamato *ECAN.zip*. La libreria in questione permette la gestione di un nodo CAN nelle modalità sia standard che estesa, pertanto sarà possibile manipolare anche i messaggi con un *arbitration field* avente un identificatore lungo. Essa rappresenta ➤

Tabella 1

Nome File	Descrizione
ECAN.c	contiene l'implementazione in C18 delle funzioni CAN
ECAN.h	contiene le dichiarazioni delle funzioni CAN
ECAN.def	contiene una serie di definizioni che stabiliscono la fase di inizializzazione del modulo CAN e la sua modalità di funzionamento.

un ottimo punto di partenza per tutti coloro che vogliono avvicinarsi allo sviluppo firmware sul CAN-Bus.

Per realizzare il nostro primo progetto CAN vediamo come è strutturata questa libreria. Essa si compone di tre file (Tabella1).

In merito al file *ECAN.def* c'è da dire che può essere personalizzato a seconda delle proprie necessità. Ad esempio, tale libreria permette di sfruttare tre possibili modalità di funzionamento:

MOD00: compatibile con la versione standard del protocollo CAN;

tive alle operazioni di interfacciamento SPI. Qui possiamo creare il file *ECAN.def* e modificarne le definizioni utilizzando un'interfaccia grafica molto intuitiva.

Ricordiamo che per avere a disposizione tutti i listati necessari, dopo aver installato il pacchetto bisogna integrarlo con i moduli in versione 1.03, che si possono liberamente scaricare dal sito ufficiale Microchip (www.microchip.com). Avviamo l'applicativo e selezioniamo il modulo *ECAN* (*Polled*); trascinando quest'ultimo nell'elenco *Selected Model* vedremo apparire nel riquadro in

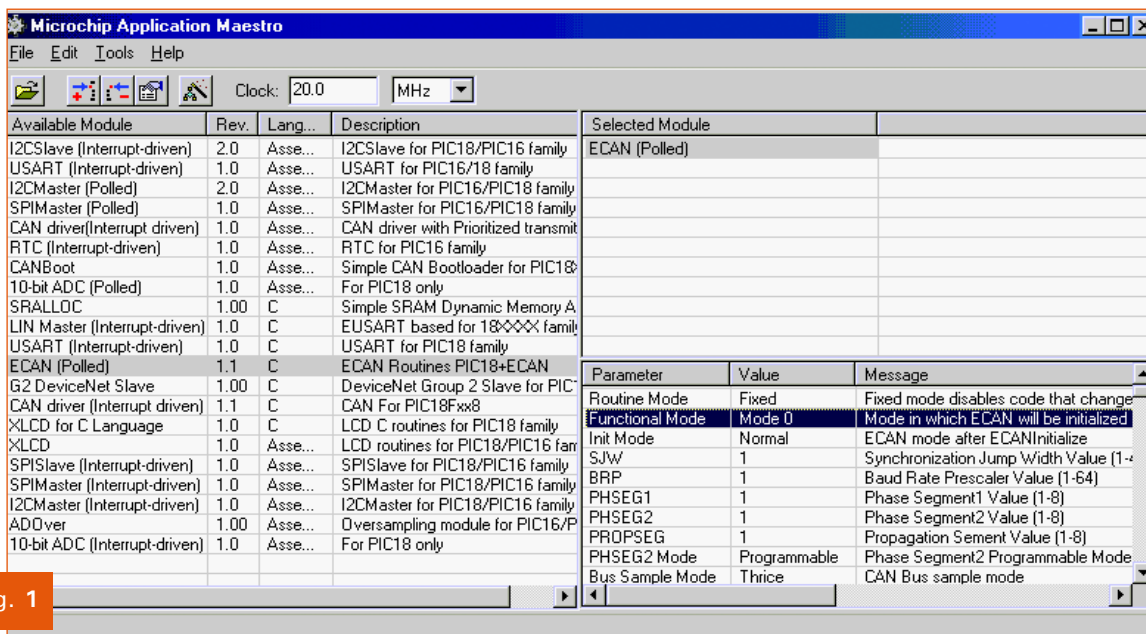


Fig. 1

MOD01: implementa le feature suppletive della versione estesa del protocollo CAN;

MOD02: compatibile con la versione estesa del protocollo CAN ed integrata dall'implementazione di un buffer FIFO in ricezione.

Nel file *.def* troviamo la seguente definizione:
#define *ECAN_FUNC_MODE_VAL*
ECAN_MODE_0. Associando uno dei tre possibili valori (*ECAN_MODE_0*, *ECAN_MODE_1*, *ECAN_MODE_2*) essa permette proprio di stabilire quale delle tre modalità utilizzare. Per una configurazione dettagliata è possibile utilizzare il modulo incluso nel pacchetto *Application Maestro* di Microchip.

Se ricordate avevamo incontrato questo sistema di sviluppo parlando delle memorie Flash; in quella occasione avevamo presentato la modalità di generazione delle funzioni rela-

basso a destra un elenco di parametri che sono proprio quelli definiti nel file *ECAN.def*. Ogni parametro è affiancato dal valore predefinito e da un messaggio che ne descrive il significato così come si vede in Fig. 1.

Se facciamo doppio clic su uno dei parametri abbiamo la possibilità di modificarne il valore. Supponiamo di voler cambiare la modalità di funzionamento del modulo CAN attraverso il campo chiamato *Functional Mode*.

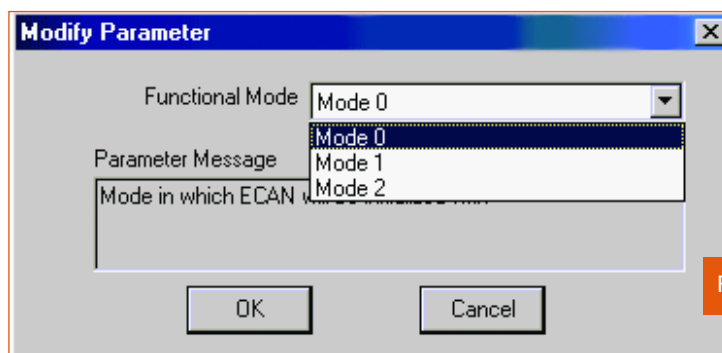


Fig. 2

Successivamente al doppio clic, apparirà la finestra di dialogo visibile in Fig. 2, nel cui menu a tendina selezioniamo, ad esempio, la modalità *Model*.

Generiamo il codice attraverso il pulsante comando evidenziato in rosso in Fig. 3 (si può impartire anche con la scorciatoia da tastiera *CTRL+G*).



Fig. 3

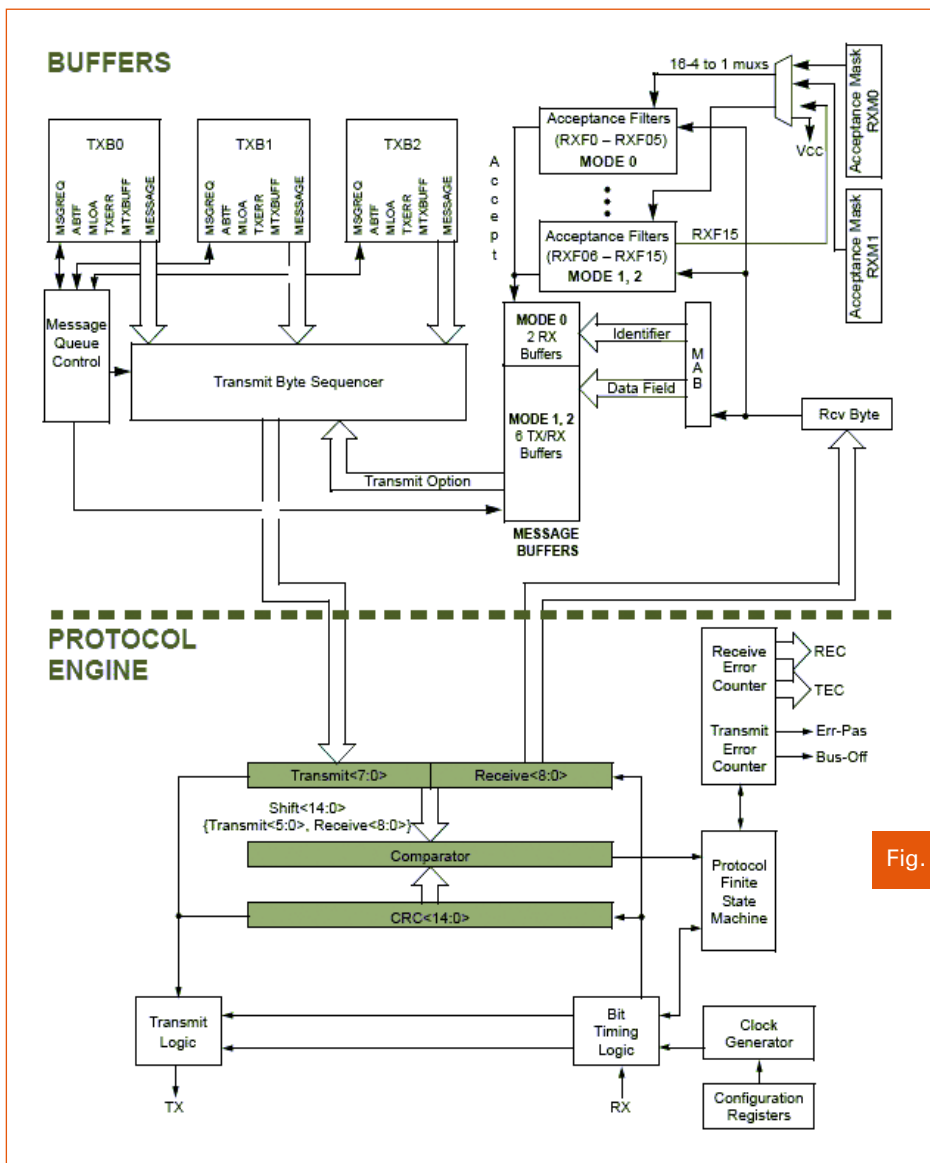


Fig. 4

questione, anche perché mette al sicuro il programmatore da eventuali errori di sintassi, soprattutto per quanto riguarda i possibili valori assegnabili ai singoli parametri. Naturalmente è possibile anche effettuare un editing diretto del file all'interno dell'ambiente MPLAB-IDE. In entrambi i casi bisogna conoscere il significato di ciascun parametro. Per farlo dobbiamo prima vedere in che modo un PIC implementa fisicamente un modulo CAN.

Hardware del Modulo CAN

Prendiamo come riferimento un PIC18F458. Il modulo CAN è

Nella directory selezionata troveremo un file chiamato *ECANPoll.def* che conterrà la definizione: `#define ECAN_FUNC_MODE_VAL ECAN_MODE_1`.

Questo è senz'altro il miglior metodo per effettuare una personalizzazione dettagliata del file in

composto da una serie di buffer, registri di controllo e da un motore che gestisce la ricezione e la trasmissione di messaggi. Osserviamo lo schema (incluso nella documentazione Microchip) di Fig. 4. Per quanto riguarda il motore si può identificare una serie di blocchi logici specializzati. Ciascuno di ➤

essi esegue una ben determinata funzione, secondo quanto esposto qui di seguito:

- 1) **Transmit Error Counter (TXERRCNT):** stabilisce la frequenza con cui avvengono gli errori di trasmissione; nel caso si verifichi un overflow il nodo viene posto in "bus-off", mentre se sul bus vengono trasmessi 128 pacchetti composti da 11 bit recessivi, il valore del contatore viene azzerato;
- 2) **Receive Error Counter (RXERRCNT):** stabilisce la frequenza con cui avvengono gli errori di ricezione; nel momento in cui il contatore assume un valore maggiore di 127 il nodo viene posto in "error-passive" (Err-Pass);

in uno stato di attesa; il relativo sotto-sistema è essenziale per la gestione dell'arbitraggio del canale;

- 8) **Transmit Logic:** effettua tutte le operazioni necessarie all'invio corretto di messaggi sul bus come ad esempio l'ascolto del canale per evitare collisioni.

La logica di gestione degli errori comporta una serie di passaggi tra diversi stati, come si vede dal diagramma di Fig. 5.

Molto importante è la struttura di buffer realizzata all'interno del PIC, perchè permette di effettuare una gestione efficiente dei messaggi rice-

vuti attraverso opportuni filtri, nonché di quelli in uscita, attraverso una coda basata su diversi livelli di priorità. Un PIC come il 18F458 contiene tre buffer in trasmissione (*TXB0*, *TXB1*, *TXB2*) e due in ricezione (*RXB0*, *RXB1*) due maschere di input (*RXM0*, *RXM1*) e sei filtri di input (*RXF0*, *RXF1*, *RXF2*, *RXF3*, *RXF4*, *RXF5*). Analizziamo prima la parte relativa alla trasmissione. Ciascun buffer di trasmissione è associato a un registro di controllo chiamato *TXBnCON*

(Transmit Buffer n Control Registers) composto dai seguenti bit:

- TXREQ (Transmit Request Status Bit):** quando è posto a 1 viene inoltrata la richiesta di invio di un messaggio; il suo valore è azzerato quando la trasmissione termina con successo;
- TXABT (Transmission Aborted Status Bit):** se è uguale a 1 la trasmissione è stata annullata;
- TXLARB (Transmission Lost Arbitration Bit):** se è uguale a 1 il nodo ha perso il controllo del canale durante la trasmissione;
- TXERR (Transmission Error Detected Status Bit):** se è uguale a 1 si è verificato un errore durante la trasmissione;
- TXPRI0-TXPRI1 (Transmit Priority bits):** è una coppia di bit che stabilisce la priorità con cui il messaggio in uscita verrà trattato dallo spooler che gestisce la coda di trasmissione; si va da un minimo di 0 ad un valore massimo di 3 (11 binario); chiaramente tale valore stabilisce l'ordine con cui i messaggi vengono trasmessi ma non influenza assolutamente il campo identificatore del messaggio stesso.

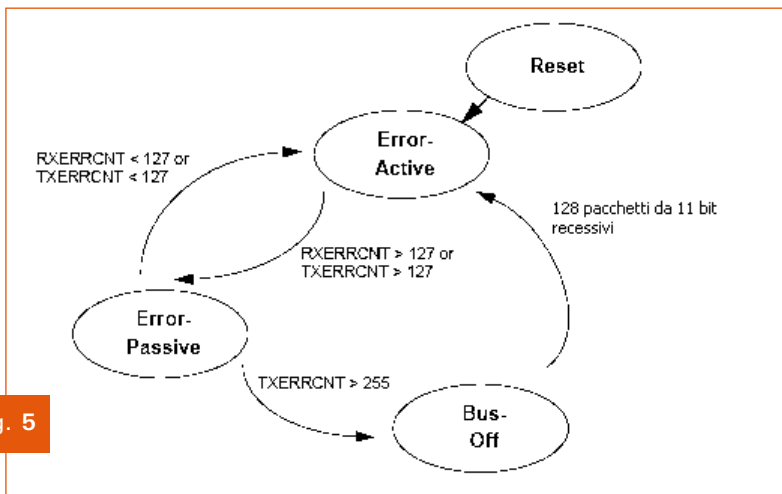


Fig. 5

- 3) **CRC Register:** Effettua il controllo del CRC per i messaggi in ricezione e lo ricalcola per quelli in trasmissione;
- 4) **Comparator:** permette di effettuare operazioni logiche di confronto tra i vari registri, tipicamente per il controllo del CRC;
- 5) **Bit Timing Generator/Logic:** si occupa di controllare la sincronizzazione delle sequenze di bit in ingresso ed uscita; ricordiamo, infatti, che ciascun nodo utilizza un bit-rate costante;
- 6) **Transmit/Receive Shift:** si tratta di due registri di shift che permettono di trasferire le sequenze di bit in ingresso ed uscita dal motore alla struttura di buffer relativa;
- 7) **Protocol FSM (Finite State Machine):** implementa i vari aspetti del protocollo CAN utilizzando un modello di macchina a stati finiti; in pratica la comunicazione avviene attraverso la transizione tra diversi stati legati ad una serie di eventi; ad esempio, nel caso in cui venga rilevata la presenza di una collisione durante uno stato di trasmissione si ha l'immediato blocco della trasmissione e la transizione

Accanto al registro di controllo TXBnCON troviamo una serie di registri operativi abbinati a ciascun buffer, che permettono il salvataggio dei vari campi di cui è composto il messaggio; i registri in questione sono:

TXBnSIDH (Transmit Buffer n Standard Identifier High Byte Registers): contiene il valore del byte più significativo dell'identificatore in formato standard relativo al messaggio contenuto nel buffer n ($0 \leq n < 3$);

TXBnSIDL (Transmit Buffer n Standard Identifier Low Byte Registers): contiene il valore del byte meno significativo dell'identificatore in formato standard nonché un bit di abilitazione del formato esteso e i due bit più significativi dell'identificatore in formato esteso relativo al messaggio contenuto nel buffer n ($0 \leq n < 3$);

TXBnEIDH (Transmit Buffer n Extended Identifier High Byte Registers): contiene il valore del byte più significativo dell'identificatore in formato esteso relativo al messaggio contenuto nel buffer n ($0 \leq n < 3$);

TXBnEIDL (Transmit Buffer n Extended Identifier Low Byte Registers): contiene il valore del byte meno significativo dell'identificatore in formato esteso relativo al messaggio contenuto nel buffer n ($0 \leq n < 3$);

TXBnDm (Transmit Buffer n Data Field Byte m Registers): contiene gli "m" byte dati ($0 \leq m < 8$) relativi al messaggio contenuto nel buffer n ($0 \leq n < 3$). Ricordiamo che il campo dati può essere lungo fino a 8 byte;

TXBnDLC (Transmit Buffer n Data Length Code Registers): contiene la lunghezza del campo dati che va da un minimo di 0 ad un massimo di 8 byte ed un bit chiamato *TXRTR* (Transmission Frame

Remote Transmission Request bit) che attiva o meno il flag RTR per cui un nodo anziché attendere passivamente delle informazioni da un altro dispositivo le richiede direttamente.

Per quanto riguarda la ricezione di messaggi dobbiamo considerare che la struttura si compone di due buffer di ricezione ognuno dei quali è in grado di contenere un intero messaggio. Esistono, quindi, dei registri esattamente complementari a quelli visti per la trasmissione che contengono i vari campi del messaggio (*RXBnSIDH*, *RXBnSIDL* ecc.). A supporto di tali registri c'è il cosiddetto *MAB* (Message Assembly Buffer) il quale non fa altro che acquisire il prossimo messaggio presente sul bus fornendo un terzo livello di bufferizzazione. Il messaggio in questione viene trasferito al relativo buffer *RXBn* soltanto se soddisfa i criteri stabiliti dai filtri di input. Dobbiamo fare una distinzione tra filtro e maschera di input. Una maschera stabilisce la posizione dei bit dell'identificatore da ➤

Tabella 2

Modalità	Valore REQOP (CANCON bit 7..5)	Descrizione
CONFIGURATION	1xx (i valori x sono di tipo "not-care" pertanto possono essere sia ad 1 che a 0)	In questa modalità è possibile accedere a tutti i registri di configurazione del nodo (cosa che non può avvenire nelle altre modalità). Il nodo non riceve né trasmette, i contatori per gli errori vengono azzerati, ed i flag relativi agli interrupt rimangono costanti. In questo modo si proteggono da un accesso non controllato i registri fondamentali relegando la loro valorizzazione ad una ben determinata fase di configurazione.
DISABLE	001	Il modulo CAN viene disattivato mantenendo soltanto la possibilità di intercettare un interrupt di Wake-Up basato sul traffico presente sul bus.
NORMAL	000	È la modalità di funzionamento standard. Il nodo opera nel pieno delle sue funzionalità ricevendo ed inviando messaggi.
LISTEN ONLY	011	In questa modalità il nodo riceve tutti i messaggi che transitano sul bus, anche quelli con errori. Può essere utilizzata per attività di monitoraggio del canale. Attraverso i registri di filtro è possibile selezionare i messaggi in ingresso da controllare. Il nodo opera in una modalità "silenziosa" nel senso che non trasmette nessun messaggio in uscita, inoltre i contatori degli errori vengono azzerati e disattivati.
LOOPBACK	010	Il modulo CAN viene configurato in maniera tale che i messaggi in uscita vengano caricati nel buffer di ingresso in maniera da essere interpretati come messaggi in entrata. Si tratta di una modalità molto interessante per attività di testing e di sviluppo visto che permette di verificare il comportamento del nodo alla ricezione di un ben determinato messaggio.

estrarre mentre il filtro stabilisce il valore che tali bit devono avere affinché il messaggio venga accettato dal nodo. Si faccia attenzione che il buffer *RXB0* si occupa dei messaggi ad alta priorità ed ha due filtri associati mentre *RXB1* elabora i messaggi a bassa priorità ed ha 4 filtri associati. Inoltre è possibile fare in modo che *RXB0* funzioni in una modalità “Double-Buffer”. In pratica, nel momento in cui *RXB0* contiene un messaggio valido, se ne riceve un altro non si verifica un errore di overflow ma il messaggio viene passato al buffer *RXB1* e viene trattato secondo i filtri di quest’ultimo.

Accanto ai registri associati alle operazioni di trasmissione e ricezione il PIC contiene diverse altre strutture di controllo, vediamo quelle che interessano direttamente le modifiche effettuate nei parametri del file *ECAN.def*.

CIOCAN (CAN I/O CONTROL REGISTER): questo registro contiene due bit. Il primo, chiamato *CANCAP*, permette di far sì che il modulo CAN generi un time-stamp per ogni messaggio ricevuto. Una volta attivo, il modulo CAN genera un segnale per il pin *CCP1*; quest’ultimo viene configurato in maniera tale che i quattro bit 3:0 di *CCP1CON* siano a 0011b (CCP special event trigger for CAN events). In questo modo alla ricezione di un messaggio viene estratto il valore del *Timer1* o del *Timer3* che viene utilizzato proprio come time-stamp. Il secondo bit, chiamato *ENDRHI*, stabilisce in che modo viene comandato il pin *CANTX1* in uno stato recessivo (assume la tensione *Vdd* oppure utilizza una modalità tipo “Tri-State”).

CANCON (CAN CONTROL REGISTER): con tre bit di questo registro è possibile stabilire la modalità operativa del nostro nodo. In pratica è possibile scegliere tra 5 opzioni: configuration, disable, normal, listen only, loopback. Ne riassumiamo le caratteristiche nella Tabella 2.

Nella realtà è possibile attivare una sesta modalità operativa chiamata “Error Recognition Mode”, ponendo a 11b (=3 in decimale) due bit (denominati *RXM*) del registro *RXBnCOM*. In tal modo tutti i messaggi in entrata, sia validi che invalidi, vengono accettati e trasferiti ai buffer di ricezione.

BRGCON1:BRGCON3: insieme di tre registri che permette di stabilire il bit-rate, le metodologie di sincronizzazione e il tipo di campionamento da usare. È importante considerare che

ogni nodo deve utilizzare il medesimo bit-rate nominale definito come numero di bit trasmessi al secondo in un trasmettitore ottimale. Il protocollo CAN utilizza una codifica di tipo NRZ (Non-Return-to-Zero) che non integra nello stream trasmesso il segnale di clock. Pertanto i nodi in ricezione devono recuperare il clock e sincronizzarsi con quelli in trasmissione. Per farlo correttamente la classe di PIC18FXX8 utilizza un DPLL (Digital Phase Lock Loop) che suddivide ciascun bit-time in più segmenti sulla base di un intervallo di tempo minimo chiamato “Time Quanta” (TQ). Siccome non è detto che tutti i nodi utilizzino la medesima frequenza di sistema (quella, per intenderci, derivante dall’oscillatore collegato ai pin OSC1 e OSC2), è necessario effettuare una precisa configurazione di ciascuno di essi determinando il valore del BRP (Baud Rate Prescaler intero compreso tra 0 e 63) ed il numero dei “Time Quanta” per ciascun segmento. Premetto che durante gli esperimenti del nostro corso saremo facilitati dal fatto che utilizzeremo per tutti i nodi la medesima configurazione circuitale e quindi la medesima frequenza di clock. Il bit-time nominale si può definire come il reciproco del bit-rate nominale ed è composto da 4 segmenti:

- *Synchronization Segment (Sync_Seg)*: permette la sincronizzazione tra i diversi nodi, durante questo intervallo di tempo che è lungo 1 TQ, il nodo si aspetta la transizione di livello logico del segnale in ingresso.
- *Propagation Time Segment (Prop_Seg)*: permette di compensare il ritardo derivante dal tempo di propagazione del segnale sul bus e quello derivante dal naturale ritardo interno dei nodi. La sua lunghezza può essere configurata tra un minimo di 1 ad un massimo di 8 TQ.
- *Phase Buffer Segment 1 (Phase_Seg1)* e *Phase Buffer Segment 2 (Phase_Seg2)*: permettono di ottimizzare la posizione del punto di campionamento (cioè del punto nel quale viene letto il livello logico presente sulla linea di ingresso e quindi il valore del bit ricevuto) all’interno del bit-time. La loro lunghezza può essere configurata da un minimo di 1 ad un massimo di 8 TQ.

La figura 6 permette di chiarire la successione dei segmenti (si consideri che graficamente ogni coppia di linee verticali corrisponde ad un TQ). Il bit time nominale va da un minimo di 8 TQ ad un massimo di 25 TQ. Si definisce Information

Processing Time (IPT) il tempo intercorrente tra il punto di campionamento e la determinazione del livello logico del bit successivo. Secondo le specifiche CAN questo intervallo deve essere inferiore o uguale a 2 TQ. Per il PIC18F458

esso è pari a 2, pertanto, durante la programmazione, la lunghezza del Phase Buffer Segment 2 deve essere maggiore o uguale a 2. Sulla base della frequenza del clock di sistema e quindi del quarzo utilizzato, possiamo stabilire il bit rate nominale attraverso un semplice calcolo; premesso che:

$$TQ (\mu s.) = [2x(BRP+1)]/Fosc (MHz) \text{ e}$$

$TBIT(\mu s.) = TQ (\mu s.) \times \text{numero di } TQ \text{ per bit time nominale e bit rate nominale (bit/s)} = 1/TBIT$, se poniamo $Fosc=20 \text{ MHz}$, $BRP=4$ e il bit time nominale pari a 16 TQ (il doppio di quanto visto nella Fig. 6) avremo:

$$TQ = [2x(4+1)]/20 = 0,5 \mu s.$$

e $TBIT = 16x0,5 = 8 \mu s.$, oltre a

$$\text{bit rate nominale} = 1/(8x10^{-6}) = 125.000 \text{ bit/s} = 125 \text{ kbps.}$$

Per effettuare una risincronizzazione, il PIC può operare sulla lunghezza dei segmenti *Phase 1* e *Phase 2*, fermo restando che la lunghezza del primo segmento può essere solo allungata, mentre quella del secondo può essere diminuita di un

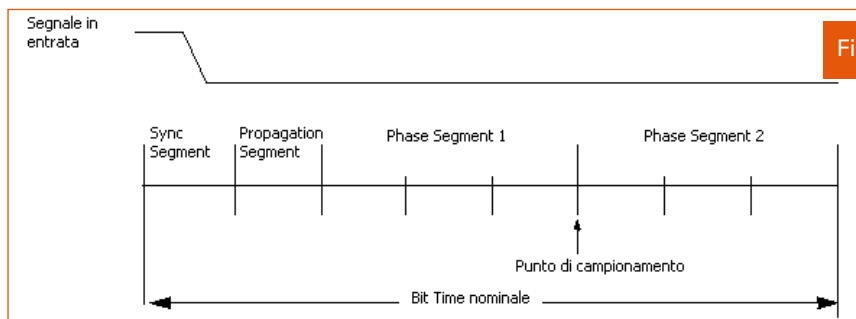


Fig. 6

intervallo che viene chiamato *Synchronization Jump Width (SJW)*. Il valore dell'intervallo va da un minimo di una ad un massimo di quattro volte TQ e può essere configurato tramite il registro *BRGCON1*. In linea generale, considerando la buona stabilità degli oscillatori ceramici, nella maggior parte dei casi è più che sufficiente un valore pari ad 1. Dopo aver visto la struttura hardware con cui un PIC implementa il modulo CAN, possiamo analizzare con maggior consapevolezza la tabella dei parametri precisabili all'interno del file *ECAN.def* (vedere la Tabella 3 a fondo pagina).

Finalmente abbiamo raggiunto il punto di contatto tra la parte teorica e quella pratica di questo corso. La presenza di una libreria C18 in grado di gestire sia la modalità standard che quella estesa del protocollo CAN, ha comportato la necessità di sottoporvi qualche paragrafo teorico in più, ma a questo punto abbiamo tutte le informazioni necessarie e sufficienti per iniziare il nostro primo sviluppo firmware per CAN bus. Utilizzeremo lo stesso approccio usato anche in ➤

Tabella 3

Parametro	Valori Possibili	Descrizione
ECAN_Bn_AUTORTR_MODE	ECAN_AUTORTR_MODE_DISABLE ECAN_AUTORTR_MODE_ENABLE	Configura il buffer di trasmissione n (con n compreso tra 0 e 5) nella modalità RTR (Remote Transmission Request) che permette di gestire automaticamente l'interrogazione di un nodo remoto. Può essere utilizzato nel modo 1 e 2 quindi con piena compatibilità con lo standard esteso.
ECAN_Bn_MODE_VAL	ECAN_RECEIVE_ALL_VALID ECAN_RECEIVE_STANDARD ECAN_RECEIVE_EXTENDED ECAN_RECEIVE_ALL	Configura il buffer di ricezione n (con n compreso tra 0 e 5) precisando le tipologie di messaggi che è abilitato a ricevere. Può essere utilizzato nel modo 1 e 2.
ECAN_Bn_TXRX_MODE_VAL	ECAN_BUFFER_TX ECAN_BUFFER_RX	Configura il buffer n (con n compreso tra 0 e 5) in trasmissione o ricezione. Può essere utilizzato nel modo 1 e 2.
ECAN_BRP_VAL	da 1 a 8	Stabilisce il valore da assegnare al BRP (Baud Rate Prescaler).
ECAN_BUS_SAMPLE_MODE_VAL	ECAN_BUS_SAMPLE_MODE_THRICE ECAN_BUS_SAMPLE_MODE_ONCE	Stabilisce se il campionamento della linea di ricezione necessario per stabilire il suo livello logico deve avvenire in una sola fase (al punto di campionamento) o in tre fasi (una precedente al punto di campionamento e due in corrispondenza dello stesso).
ECAN_CAPTURE_MODE_VAL	ECAN_CAPTURE_MODE_DISABLE ECAN_CAPTURE_MODE_ENABLE	Permette di attivare o disattivare la possibilità di generare un timestamp per ogni messaggio ricevuto attraverso il campionamento del segnale su CCP1. È necessario configurare il registro CCP1CON abilitando la modalità "CCP special event trigger for CAN events".

Continuazione Tabella 3

Parametro	Valori Possibili	Descrizione
ECAN_FILTER_MODE_VAL	ECAN_FILTER_MODE_ENABLE ECAN_FILTER_MODE_DISABLE	Attiva o disattiva l'uso di un filtro passa-basso per la rilevazione dell'attività del bus nella modalità di wake-up. In pratica configura il flag WAKFIL del registro BRGCON3.
ECAN_FUNC_MODE_VAL	ECAN_MODE_0 ECAN_MODE_1 ECAN_MODE_2	Definisce la modalità operativa del nodo secondo quanto visto nei precedenti paragrafi.
ECAN_INIT_MODE	ECAN_INIT_NORMAL ECAN_INIT_CONFIGURATION ECAN_INIT_LOOPBACK ECAN_INIT_DISABLE ECAN_INIT_LISTEN_ONLY	Definisce il modo di funzionamento iniziale secondo quanto visto nei precedenti paragrafi.
ECAN_LIB_MODE_VAL	ECAN_LIB_MODE_FIXED ECAN_LIB_MODE_RUNTIME	Definisce la possibilità di modificare le configurazioni del nodo a run-time (come ad esempio la modalità di funzionamento).
ECAN_PHSEG1_VAL	da 1 a 8	Stabilisce la lunghezza del segmento Phase 1.
ECAN_PHSEG2_MODE_VAL	ECAN_PHSEG2_MODE_PROGRAMMABLE ECAN_PHSEG2_MODE_AUTOMATIC	Permette di stabilire se il segmento Phase 2 deve essere gestito automaticamente dal modulo ECAN o può essere programmato liberamente.
ECAN_PHSEG2_VAL	da 1 a 8	Stabilisce la lunghezza del segmento Phase 2.
ECAN_PROPSEG_VAL	da 1 a 8	Stabilisce la lunghezza del segmento Propagation.
ECAN_RXB0_DBL_BUFFER_MODE_VAL	ECAN_DBL_BUFFER_MODE_DISABLE ECAN_DBL_BUFFER_MODE_ENABLE	Attiva o disattiva la modalità double-buffer per RXB0. È utilizzabile solo in modo 0.
ECAN_RXB0_MODE_VAL	ECAN_RECEIVE_ALL_VALID ECAN_RECEIVE_STANDARD ECAN_RECEIVE_EXTENDED ECAN_RECEIVE_ALL	Stabilisce la tipologia di messaggi in ingresso accettati da RXB0.
ECAN_RXB1_MODE_VAL	ECAN_RECEIVE_ALL_VALID ECAN_RECEIVE_STANDARD ECAN_RECEIVE_EXTENDED ECAN_RECEIVE_ALL	Stabilisce la tipologia di messaggi in ingresso accettati da RXB1.
ECAN_RXF0_MASK_VAL	ECAN_RXM0 ECAN_RXM1 ECAN_RXMF15	Collega RXF0 ad una ben determinata maschera. È utilizzabile nel modo 1 e 2. Si noti che è possibile configurare il filtro RXF15 come una maschera di input.
ECAN_RXFn_BUFFER_VAL	RXB0 RXB1 B0 B1 B2 B3 B4 B5	Collega un filtro ad uno specifico buffer in ricezione. È utilizzabile nel modo 1 e 2.
ECAN_RXFn_MODE_VAL	ECAN_RXFn_ENABLE ECAN_RXFn_DISABLE	Permette di abilitare o disabilitare i filtri in ricezione RXFn con n che va da 0 a 15. Può essere usato nella modo 1 e 2. Nel modo 0 n va da 0 a 5 e i filtri relativi sono tutti attivi.
ECAN_RXFn_MSG_TYPE_VAL	ECAN_MSG_STD ECAN_MSG_XTD	Permette di definire la modalità standard o estesa per i filtri RXFn. Nel modo 0 sono disponibili solo RXF0-RXF5.
ECAN_RXFn_VAL	un valore a 11 bit o a 29 bit	Assegna il valore a 11 bit o a 29 bit al filtro relativo RXFn.
ECAN_RXMn_MSG_TYPE	ECAN_MSG_STD ECAN_MSG_XTD	Definisce la modalità standard o estesa per le maschere di input RXM0-RXM1.
ECAN_SJW_VAL	da 1 a 4	Stabilisce la lunghezza del SJW (Synchronization Jump Width).
ECAN_TX2_MODE_VAL	ECAN_TX2_MODE_DISABLE ECAN_TX2_MODE_ENABLE	Abilita o disabilita il pin CANTX2 nei PIC che lo prevedono.
ECAN_TX2_SOURCE_VAL	ECAN_TX2_SOURCE_COMP ECAN_TX2_SOURCE_CLOCK	Definisce la sorgente di segnale per il pin CANTX2 che può essere il complemento del segnale presente su CANTX1 oppure il clock del modulo CAN.
ECAN_TXDRIVE_MODE_VAL	ECAN_TXDRIVE_MODE_TRISTATE ECAN_TXDRIVE_MODE_VDD	Definisce in che modo il pin CANTX1 viene comandato in uno stato recessivo.
ECAN_WAKEUP_MODE_VAL	ECAN_WAKEUP_MODE_ENABLE ECAN_WAKEUP_MODE_DISABLE	Attiva o Disattiva la modalità di Wake-up per il nodo.

precedenti corsi, secondo cui le singole istruzioni (in questo caso sarebbe meglio dire funzioni) verranno spiegate in dettaglio al momento del loro utilizzo all'interno del codice.

Il nostro primo esperimento ci permetterà di capire come un nodo possa acquisire dei dati dall'esterno (ad esempio da una sonda termometri-

ca) e renderli disponibili sul bus affinché possano essere utilizzati dagli altri nodi. È il primo passo verso la descrizione della cooperazione tra più dispositivi CAN, obiettivo del quale è arrivare a considerare un insieme di nodi CAN come un unico sistema in grado di svolgere più compiti, distribuendo le informazioni necessarie.

Una serie completa di scatole di montaggio hi-tech che sfruttano la rete GSM.

G
S
M

S
O
L
U
T
I
O
N
S

APRICANCELLO

Facilmente abbinabile a qualsiasi cancello automatico. Attiva un relè di uscita (da collegare all'impianto esistente) quando viene chiamato da un telefono fisso o mobile precedentemente abilitato. Programmazione remota mediante SMS con password di accesso. Completo di contenitore e antenna bibanda. Alimentatore non compreso.

FT503K Euro 240,00



TELECONTROLLO

Sistema di controllo remoto che consente di attivare, mediante normali SMS, più uscite, di verificare lo stato delle stesse, di leggere il valore logico assunto dagli ingressi nonché di impostare questi ultimi come input di allarme. Possibilità di espandere gli ingressi e le uscite digitali. Funziona anche come apricancello. Completo di contenitore.

FT512K Euro 255,00



TELEALLARME A DUE INGRESSI

Invia ad uno o più utenti un SMS di allarme quando almeno uno degli ingressi viene attivato con una tensione o con un contatto. Può essere facilmente collegato ad impianti di allarme fissi o mobili. Ingressi fotoaccoppiati, dimensioni ridotte, completamente programmabile a distanza.

FT518K Euro 215,00



CONTROLLO REMOTO

2 CANALI CON TONI DTMF

Telecontrollo DTMF funzionante con la rete GSM. Questa particolarità consente al nostro dispositivo di operare ovunque, anche dove non è presente una linea telefonica fissa. Può essere chiamato e controllato sia mediante un cellulare che tramite un telefono fisso. Il kit comprende il contenitore; non sono compresi l'antenna e l'alimentatore.

FT575K Euro 240,00



ASCOLTO AMBIENTALE

Sistema di ridotte dimensioni per l'ascolto ambientale. Può essere facilmente nascosto all'interno di una vettura o utilizzato in qualsiasi altro ambiente. Regolazione della sensibilità da remoto, chiamata di allarme mediante sensore di movimento, password di accesso. Viene fornito con l'antenna a stilo, mentre il sensore di movimento è disponibile separatamente.

FT507K Euro 280,00



MICROSPIA TELEFONICA

Collegata ad una linea telefonica fissa, consente di ascoltare da remoto tutte le telefonate effettuate da quella utenza. La ritrasmissione a distanza delle telefonate sfrutta la rete GSM. Microfono ambientale supplementare, I/O a relè. La scatola di montaggio non comprende il contenitore e l'antenna GSM.

FT556K Euro 245,00



COMMUTATORE TELEFONICO

Collegato al telefono di casa effettua automaticamente una connessione GSM tutte le volte che componiamo il numero di un telefonino. In questo modo possiamo limitare il costo della bolletta in quanto una chiamata cellulare-cellulare costa quasi la metà rispetto ad una chiamata cellulare-fisso. Il kit non comprende il contenitore e l'antenna GSM.

FT565K Euro 255,00



**FUTURA
ELETTRONICA**

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

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

Tutti i prezzi si intendono IVA inclusa.

con il patrocinio del
Ministero delle
Comunicazioni

Mostra mercato

Expo[®] 2006 Elettronica

ERBA 1 • 2 aprile

ore 9/18

 **LARIO FIERE**
(Como)
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 del
Gruppo Lecco Utenti Linux
www.lecco.linux.it
con il supporto tecnico di **yetopen.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.pcbshop.it>



Realizzare un prototipo o una serie in poco tempo è spesso determinante; ecco perché sono sempre più numerose le aziende operanti on-line che producono circuiti stampati dando la possibilità al cliente di inviare i master

per via telematica sotto forma di file. La nostra ricerca su Internet ci ha permesso di selezionare tre aziende con tali caratteristiche. PCB Shop produce c.s. dal singola faccia al multilayer a sei strati, stagnati, argentati o dorati. Accetta file in formato *Gerber* esteso e *NCDrill* (foratura). Offre il preventivo immediato on-line ed evade le campionature anche in soli 2 giorni.

<http://www.mdsrl.it>



MD S.r.l. è un'azienda specializzata nella preparazione di circuiti stampati per prototipi e serie, sia a singola faccia che a doppia faccia con fori metallizzati, con finitura a stagno chimico; il suo punto di forza è la rapidità di consegna: anche sole 24 ore (per la versione base, ossia solo rame e rivestimento con stagno delle piste, 48 se è richiesto il solder-resist) dall'invio del file (che deve essere nel formato *Gerber*, *Gerber* esteso, cioè *RS274-X*, *Cadsoft Eagle .brd*) per via telematica, eventualmente contenente le tracce dei lati rame, l'eventuale maschera del solder-resist e il piano di foratura. Nel sito è spiegato come preparare i file.

<http://www.ermes-technology.it>



Produttore di circuiti stampati civili e professionali dal semplice monofaccia fino al multilayer a 20 strati (con fori di interconnessione sia passanti che interrati) stagnati (con procedimento sia chimico che a fusione) passivati con rame, nichelati, grafitati, argentati e dorati sia chimicamente che con procedimento elettrolitico, rivestiti con solder-resist e serigrafati, ERMES si occupa essenzialmente della produzione di prototipi e piccole serie, partendo dal file *Gerber* inviato dal cliente (che può contare sul servizio di trasmissione FTP). La consegna dei prototipi può avvenire anche in soli tre giorni ed è possibile ottenere un preventivo on-line.

Vendo:

- Microfono spia direttivo a condensatore autoalimentato a euro 13,00;
- Mini microfono spia amplificato con ascolto negli auricolari in dotazione a euro 10,00;
- Microfono spia amplificato con ascolto nella cuffia in dotazione a euro 13,00;
- Mini parabola portatile con incorporati faretto, cannocchiale, e microfono spia amplificato con ascolto nella cuffia in dotazione a euro 34,00;
- Microspia ricevibile da ricevitori F.M. a 115MHz a euro 15,00;
- Cuffia con incorporato ricevitore stereo multibanda 80 a 115MHz, accessoriata per tutti gli usi domestici e di spionaggio con microspia, microfono a clip, trasmettitore stereo 83MHz ecc... a 40,00;
- Binocolo semiperiscopico tascabile per spionaggio con incorporato radio AM-FM separabile con altoparlante e auricolari a euro 25,00.

Tutti articoli nuovi! Spedisco.
Contattare Pietro al numero di telefono 0371-30418.

Vendo i seguenti starter kit:

- Kit completo per ST626X della SGS-THOMSON a euro 200,00;
- Sistema di sviluppo con emulatore per micro Zilog Z8 completo di Software e accessori a euro 100,00;
- Kit vari di Nuova Elettronica per lo studio, l'apprendimento, e la pratica dei microcontrollori ST6, il tutto per 120,00;
- Lavatrice ad ultrasuoni per piccoli oggetti, facile uso a euro 95,00.

Contattare Stefano al numero di cellulare 347-9019224.

Vendo:

- Cassetto per HP-141 RF 10-110 MHz 8553B con manuale a euro 200,00;
- Cassetto per HP-141 LF 20 Hz-300 KHz 8556A con manuale a euro 250,00;
- Mixer est. TEK 12-40 GHz (in 3 guide d'onda) a euro 300,00;
- Scheda SAIF-100 di acquisizione per HP-141 a euro 350,00;
- Vector Voltmeter HP-8405A a euro 450,00;
- HP-431C Power Meter senza sonda a euro 150,00;
- ICOM R71 - Ricevitore 0.1 - 30 Mhz con filtro SSB a euro 600,00;
- YAESU FT-23R;
- Microfono da tavolo Yaesu MD1 a euro 40,00;
- HP-215A Pulse Generator - Trigger 100 Hz - 1 MHz Pulse Width min. 10 nS a euro 100,00;
- Amplificatore RF 5.7 GHz con TWT RW-89 con alim. Siemens RWN-110 a euro 350,00;
- TWT RW-89 Siemens 15 W - 5.9-6.5 GHz a euro 100,00;
- TWT RW-85 Siemens 22 W - 6.4-7.1 GHz a euro 120,00;
- Transverter Microset 144-28 Mhz a euro 150,00. Contattare Davide al numero 335-6312494.

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.

Cerco in Milano:

Collaboratore per sviluppare apparecchi elettronici e un progetto da me inventato non perdetempo, massima serietà. Contattare Jerry al 328-1476157.

Sviluppo:

Firmware per microcontrollori STXX e PICXX e realizzo prototipi. Contattare Gianni al numero di 0376-396743.

Cerco:

Telefonino Nokia 3120 NON funzionante ma con display intatto in quanto debbo sostituirlo con quello rotto.

Contattare Luigi al numero di 338-2771113.

Vendo:

Modulo Telit GM862 GPRS completo di strip di collegamento PCB e antenna con cavo e connettore di collegamento al modulo.

Tutto nuovo mai montato. Prezzo di listino di euro 160,00 modulo, euro 8,00 connettore strip PCB, euro 55,00 antenna, euro 35,00 cavo diretto al modulo.

Vendo e realizzo a euro 200,00 tutto compreso.

Contattare Federico al numero di 3939469204 o info@fixfast.it (preferibile E-mail per l'invio di eventuali foto).

Vendo:

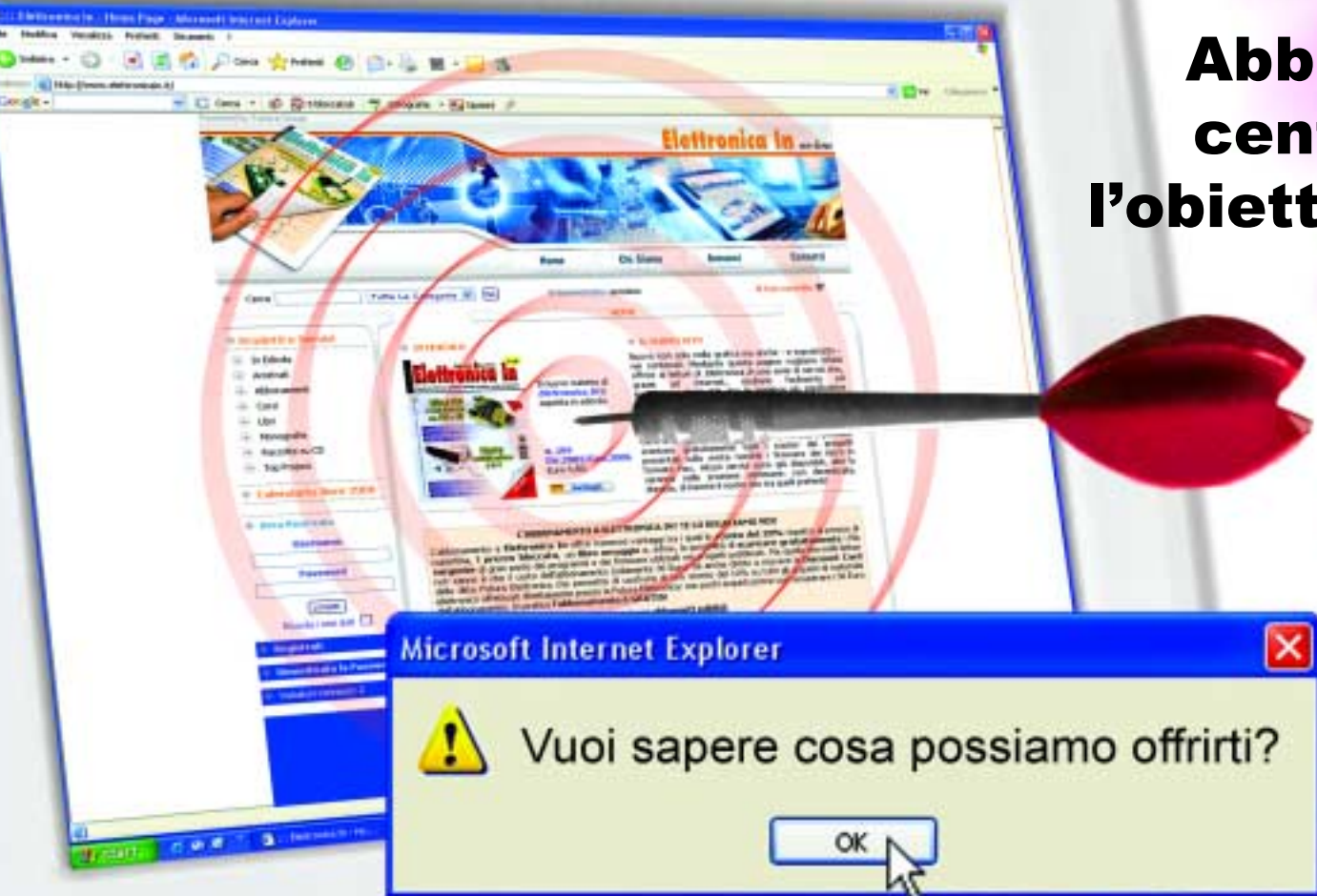
- Regolatore di potenza "Fiber" alim. 220V a euro 15,00;
- Temporizzatore multitutto con display HCA-A alim. 12/240V a euro 30,00;
- Omron Level Meter E4M-3AK ultrasuoni con uscita analogica 4-20mA e NO/NC alim. 220V a euro 100,00;
- Videocitofono B/N digitale a euro 100,00;
- Termostato elettronico "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.



Elettronica In *on-line*

**Abbiamo
centrato
l'obiettivo...**



**Per te ogni giorno un sito
ricco di novità da scoprire...**

Potrai:

- **Abbonarti o rinnovare l'abbonamento** alla rivista di Elettronica In;
- **Verificare** lo stato dell'abbonamento;
- **Scaricare:**
 - ▶ i **master** dei circuiti stampati, **software** e **firmware** relativi ai progetti pubblicati;
 - ▶ i **file sorgente** utilizzati in molti dei progetti presentati sulla rivista. *[solo per gli abbonati]*

Potrai inoltre acquistare, scaricandoli direttamente dal sito, tutti i nostri prodotti in formato digitale, dalle **riviste arretrate**, ai **corsi**, alle **raccolte tematiche**, ai **top projects**.

Troverai anche numerosi altri servizi:

- l'elenco di tutte le **manifestazioni fieristiche** di elettronica;
- le **news** dalle aziende;
- i **link più interessanti** e ... tanti altri servizi che stiamo implementando.

**...collegati a
www.elettronica.in.it**

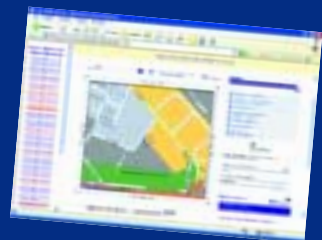
Sistemi professionali GPS/GSM

Localizzatore GPS/GSM portatile

FT596K (premontato) - Euro 395,00



Unità di localizzazione remota GPS/GSM di dimensioni particolarmente contenute ottenute grazie all'impiego di un modulo Wavecom Q2501 che integra sia la sezione GPS che quella GSM. L'apparecchio viene fornito premontato e comprende il localizzatore vero e proprio, l'antenna GPS, quella GSM ed i cavi adattatori d'antenna. La tensione di alimentazione nominale è di 3,6V, tuttavia è disponibile separatamente l'alimentatore switching in grado di funzionare con una tensione di ingresso compresa tra 5 e 30V (FT601M - Euro 25,00) che ne consente l'impiego anche in auto. I dati vengono inviati al cellulare dell'utente tramite SMS sotto forma di coordinate (latitudine+longitudine) o mediante posta elettronica (sempre sfruttando gli SMS). In quest'ultimo caso è possibile, con delle semplici applicazioni web personalizzate, sfruttare i siti Internet con cartografia per visualizzare in maniera gratuita e con una semplice connessione Internet (da qualsiasi parte del mondo) la posizione del target e lo spostamento dello stesso all'interno di una mappa. Sono disponibili per questo apparato sistemi autonomi di alimentazione (pacchi di batterie al litio) che consentono, unitamente a speciali magneti, di effettuare l'installazione in pochi secondi su qualsiasi veicolo. Ulteriori informazioni sui nostri siti www.futurashop.it e www.gpstracer.net.



SERVIZIO WEB GRATUITO

A quanti acquistano una nostra unità remota GPS/GSM diamo la possibilità di utilizzare gratuitamente il nostro servizio di localizzazione su web. Potrete così, mediante Internet, e senza alcun aggravio di spesa, visualizzare la posizione del vostro veicolo su una mappa dettagliata 24 ore su 24.

Produciamo e distribuiamo sistemi di controllo e sorveglianza remoti basati su reti GSM e GPS. Oltre ai prodotti standard illustrati in questa pagina, siamo in grado di progettare e produrre su specifiche del Cliente qualsiasi dispositivo che utilizzi queste tecnologie. Tutti i nostri prodotti rispondono alle normative CE e RTTE.

Localizzatore miniatura GPS/GSM con batteria inclusa

G19B - Euro 499,00



Dispositivo di localizzazione personale e veicolare di ridottissime dimensioni. Integra un modem cellulare GSM, un ricevitore GPS ad elevata sensibilità ed una fonte autonoma di alimentazione (batteria al litio). I dati relativi alla posizione vengono inviati tramite SMS ad intervalli programmabili a uno o più numeri di cellulare abilitati. Questi dati possono essere utilizzati anche da appositi programmi web che consentono, tramite Internet, di visualizzare la posizione del target su mappe dettagliate.

MODALITA' DI FUNZIONAMENTO

Invio di SMS ad intervalli predefiniti: l'unità invia ai numeri telefonici abilitati un messaggio con le coordinate ad intervalli di tempo predefiniti, impostabili tra 2 e 120 minuti. Gli SMS contengono l'identificativo dell'unità con i dati relativi alla posizione, velocità e direzione nel formato pre-selezionato.

Polling: l'unità può essere chiamata da un telefono il cui numero sia stato preventivamente memorizzato; al chiamante viene inviato un SMS con tutti i dati relativi alla posizione del dispositivo.

Polling SMS: Inviando un apposito SMS è possibile ottenere un messaggio di risposta contenente le informazioni relative alla cella GSM in cui l'unità remota è registrata. Questa funzione consente di sapere (in maniera molto più approssimativa) dove si trova il dispositivo anche quando non è disponibile il segnale della costellazione GPS.

Emergenza: Questa funzione fa capo al pulsante Panic dell'unità remota: premendo il pulsante viene inviato ad un massimo di tre numeri telefonici preprogrammati un SMS di richiesta di aiuto contenente anche i dati sulla posizione. L'attivazione di questo pulsante determina anche un allarme acustico.

Tutti i prezzi si intendono IVA inclusa.

Localizzatore GPS/GSM GPRS con batteria e microfono inclusi

WEBTRAC4S - Euro 645,00



Sistema di localizzazione personale e veicolare di ridottissime dimensioni. Si differenzia dal modello standard (G19B) per la possibilità di utilizzare connessioni GPRS (oltre alle normali GSM) e per la disponibilità di un microfono integrato ad elevata sensibilità. I dati relativi alla posizione vengono inviati tramite la rete GPRS o GSM mediante SMS o email. Funzione panico e parking. Possibilità di utilizzare servizi web per la localizzazione tramite pagine Internet.

MODALITA' DI FUNZIONAMENTO

Invio dei dati di localizzazione tramite rete GPRS e web server: l'unità remota è connessa costantemente alla rete GPRS ed invia in tempo reale i dati al web server; è così possibile conoscere istante dopo istante la posizione del veicolo e la sua direzione e velocità con un costo particolarmente contenuto dal momento che nella trasmissione a pacchetto (GPRS) vengono addebitati solamente i dati inviati ed in questo caso ciascun pacchetto che definisce la posizione è composto da pochi byte.

Ascolto ambientale tramite microfono incorporato: chiamando il numero dell'unità remota, dopo otto squilli, entrerà in funzione il microfono nascosto consentendo di ascoltare tutto quanto viene detto nell'ambiente in cui opera il dispositivo. Utilizzando un'apposita cuffia/microfono sarà possibile instaurare una conversazione voce bidirezionale con l'unità remota. La sensibilità del microfono è di -24dB.

Emergenza: Questa funzione fa capo al pulsante Panic dell'unità remota: premendo il pulsante viene inviato in continuazione al web server un messaggio di allarme con i dati della posizione ed a tutti i numeri telefonici memorizzati un SMS di allarme con le coordinate fornite dal GPS.

Park/Geofencing: tale modalità di funzionamento può essere attivata sia con l'apposito pulsante che mediante l'invio di un SMS. Questa funzione - attivata solitamente quando il veicolo viene posteggiato - determina l'interruzione dell'invio dei dati relativi alla posizione. Qualora il veicolo venga spostato e la velocità superi i 20 km/h, la trasmissione riprende automaticamente con una segnalazione d'allarme. Qualora la connessione GPRS non sia disponibile, vengono inviati SMS tramite la rete GSM.

FUTURA ELETTRONICA

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

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

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 apriancello. Tutte le impostazioni avvengono tramite SMS. Alimentazione compresa tra 5 e 32 Vdc, assorbimento massimo 500mA. Antenna GSM bbanda integrata. GSM: Dual Band EGS 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.

TDG33 - Euro 198,00

