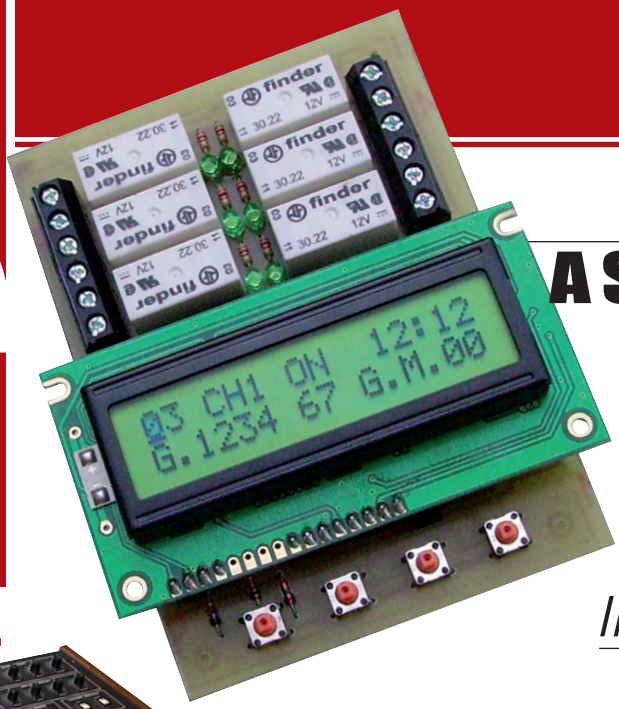
fare elettronica

N° 268 OTTOBRE - Anno 23 - € 6,00

www.farelettronica.com



timer

A SEI CANALI CON LCD

IN PRATICA

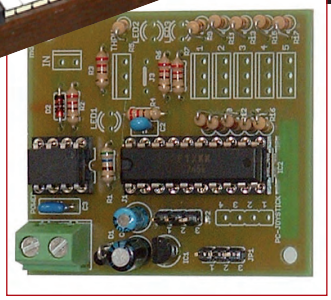
Una lampadina allo stato solido





MIDI

Una scheda di interfaccia PER PC



PIC MICRO

Un nuovo modo per
GESTIRE UNA TASTIERA

TRASDUTTORI DI POSIZIONE

Encoder e commutatori
CODIFICATI

LCD Grafici

Come realizzare
UN ALBUM FOTOGRAFICO

PROGETTARE

UN EQUALIZZATORE
Ambientale

AUTOMOTIVE

I sistemi elettronici x-by-wire



DISPLAY 7 SEGMENTI

Come pilotarli con CUBLOC

ISSN 1591-2272

9 771591 227008 70268

MENSILE Poste Italiane Spa - Spedizione in abbonamento Postale - D.L. 353/2003 (conv. In L. 27/02/2004 n. 46) art. 1, comma1, DCB Milano. In caso di mancato recapito, restituire all'editore che si impegna a pagare la relativa tassa presso il CPM di Roserio - Milano

www.farelettronica.com

fare elettronica

servizio MIP

More Info Please!

Richiedi maggiori informazioni sui contenuti di Fare Elettronica, visita il sito:

www.farelettronica.com/mip

Oppure

compila questo modulo ed invialo via fax al numero 02 66508225

Numero della rivista 268

Codici MIP*
(da compilare)

DATI DEL LETTORE

NOME

COGNOME

AZIENDA

INDIRIZZO AZIENDA

CAP CITTÀ

PROV. NAZIONE

TEL.

FAX

E-MAIL

Privacy. Ai sensi del Decr. Lgs. 196/2003 la informiamo che i dati trasmessi verranno impiegati coi principali scopi di indagini di mercato e nelle modalità previste dallo stesso, prevalentemente con mezzi informatici. Il conferimento, di norma facoltativo, è obbligatorio per permettere il rapporto commerciale. È in ogni caso fatto diritto dell'interessato esercitare i propri diritti, nei modi previsti dal "Titolo II art. 7" della legge sopra citata, scrivendo a Inware Edizioni Via Cadorna 27 – 20032 Cormano o trmite email a info@inwaredizioni.it

* Utilizza il numero MIP che compare alla fine di ogni articolo o all'interno delle pagine di tuo interesse

Guida al numero 268

Informati!

Richiedi maggiori informazioni sui contenuti di Fare Elettronica, visita il sito: www.farelettronica.com/mip

pag. 3

Rispondi&Vinci!

Divertiti e metti alla prova le tue conoscenze con

ElettroQuiz

e vinci ogni mese fantastici premi!



pag. 64

Risparmia!

Abbonati o Rinnova oggi il tuo abbonamento a Fare Elettronica



1 ANNO × 11 RIVISTE = €49,50

CON UN RISPARMIO DEL 25%

pag. 112

Pratica

18 Interfaccia MIDI

Nel presente articolo viene descritta una particolare interfaccia MIDI per PC che ha una particolarità in più che senz'altro verrà apprezzata dai musicisti più esigenti. Proseguite nella lettura e scoprirete di cosa si tratta.

34 Lampdina allo stato solido

Ecco come realizzare una lampadina allo stato solido, ossia formata da diversi LED ad alta luminosità. Non richiede alimentatore in quanto funziona direttamente a 220V.

52 Programmatore orario

Avete mai avuto la necessità di controllare un carico elettrico in maniera automatica senza la necessità della vostra presenza? Con questo circuito ne potrete controllare addirittura 6!

58 I sistemi robotici Lego Mindstorms

La programmazione in NQC (II parte)

Continuiamo il discorso iniziato nello scorso numero approfondendo la trattazione del linguaggio di programmazione NQC, che ci consentirà di sviluppare delle interessanti applicazioni per il nostro sistema robotico LEGO MINDSTORMS.

66 Il PLC per tutti

Utilizzare il modulo Led a 7 segmenti giganti

In questo numero saranno esaminati i moduli dei display a 7 segmenti, collegabili direttamente al Cubloc. Cablaggio e programmazione molto semplici, per un utilizzo alla portata di tutti.

74 LCD dalla A alla Z

Utilizzare il Demograph come album fotografico elettronico

In questo articolo verranno aggiunte altre funzionalità alla libreria ad alto livello, creata nella precedente puntata. Sarà così possibile disegnare linee e rettangoli ed evidenziare i testi con un semplice trucco. Infine, per mostrare le potenzialità della scheda di test realizzata, il Demograph verrà utilizzato come visualizzatore di immagini.

88 Videogames by Example

Esempi di programmazione in Spin

In questa puntata verrà mostrato come creare con Hydra immagini e grafica da visualizzare su TV utilizzando lo Spin. In particolare verrà spiegato come utilizzare i driver video e grafico per ottenere in maniera estremamente semplice immagini ed animazioni complesse.

100 Come gestire una tastiera utilizzando un solo ingresso del PIC

Proponiamo un sofisticato metodo per la gestione di una tastiera con un microcontrollore. A differenza della tradizionale e complessa connessione a matrice, questa tecnica richiede un solo ingresso analogico.

OTTOBRE 2007



Teoria

44 **Trasduttori di posizione (II parte)**
In questa puntata, ultima di teoria, continueremo la trattazione affrontando le varie tipologie di encoder e di commutatori codificati.

108 Imparare a progettare
L'equalizzatore ambientale
Gli equalizzatori ambientali sono preamplificatori in grado di apportare notevoli correzioni alla banda audio

MHz

40 Radioregolamenti, codici e abbreviazioni
Il codice Q (V)
Abbiamo già trattato una quarantina di gruppi di questo linguaggio internazionale che rimane un caposaldo nella preparazione degli operatori radio e basterebbero già da soli per allacciare un dialogo abbastanza variegato fra persone di differente lingua, ma continueremo la descrizione degli altri gruppi che riguardano specifici argomenti.

Speciale

26 **Sistemi elettronici a bordo auto: l'approccio x-by-wire**
Uno dei principali trend di sviluppo tecnologico nel campo dei veicoli civili e industriali è rappresentato dall'utilizzo di sistemi totalmente elettrici per la trasmissione e l'attuazione dei comandi di guida in sostituzione dei tradizionali apparati meccanici e/o idraulici. Questi sistemi sono chiamati x-by-wire e la loro introduzione nei veicoli permetterà di ottenere livelli di sicurezza di guida, efficienza di impiego e flessibilità di manovra mai raggiunti prima e renderà operanti funzionalità innovative di gestione e di manutenzione dei veicoli.

Aziende citate in questo numero

Atmel	8	Lego	58
Austriamicrosystems	8, 10	Mercedes-Benz	26
BMW	26	Microchip	100
DaimlerChrysler	30	NXP	30
Freescale	30, 33	Parallax	88
Fujitsu	9, 32	Renesas	10
General Motors	30	Robert Bosch	30
IDT	9	Sena Technologies	10

Risorse

8 Prima pagina

- *Austriamicrosystems presenta l'AS1524/25, un convertitore A/D micro-power da 12-bit e 150ksps*
- *Atmel presenta un sistema di adattamento dell'impedenza di antenne AM/FM*
- *Fujitsu rilascia un CODEC HDTV H.264 con memoria integrata e consumi ultraridotti*
- *IDT introduce i primi "mobile multimedia interconnect™" per dispositivi pda e per terminali mobili di fascia alta*
- *Server cost-effective per la connessione in rete di dispositivi seriali*
- *Si espande la famiglia di starter kit per i microcontrollori SuperH*
- *Austriamicrosystems presenta l'AS5046, un encoder rotante magnetico da 12-bit*

12 Gli eventi di Ottobre 2007

- *EXPO RADIO & INFORMATICA – Tito Scalo (PZ)*
- *EXPORADIO ELETTRONICA – Faenza (RA)*
- *3a Mostra regionale dell'elettronica Scandiano (RE)*
- *III° ELETTRO@BIT – Arezzo*
- *Radioelettra Informatica – Ferrara*
- *Computerfest&Radioamatore – Firenze*
- *Elettroexpo – Verona*
- *Fiera dell'informatica e dell'elettronica – Rovigo*

14 Idee di Progetto

- *95 - Alimentare un diodo LED a 230Vac*
- *96 - Semplicissimo ECG*
- *97 - Semplicissimo Elettrometro*
- *98 - Monitorare lo stato di una lampadina*
- *99 - Termometro con LM35*
- *100 - Timer per uovo sodo*

38 www.farelettronica.com Il portale della Rivista

www.farelettronica.com

DIRETTORE RESPONSABILE
Antonio Cirella

DIRETTORE ESECUTIVO
Maurizio Del Corso

COORDINAMENTO TECNICO
Tiziano Galizia

HANNO COLLABORATO IN QUESTO NUMERO
Giovanni Di Maria, Savino Giusto, Antonio Di Stefano, Nico Grilloni, Giuseppe Modugno, Grazia Ancona, Fabio Riscica, Franco Malenza, Federico Battaglin, Daniele Cappa, Silvano Breggion.

DIREZIONE • REDAZIONE • PUBBLICITÁ
INWARE Edizioni srl - Via Cadorna, 27/31 - 20032 Cormano (MI)
Tel. 02.66504755 - Fax 02.66508225
info@inwaredizioni.it - www.inwaredizioni.it
Redazione: fe@inwaredizioni.it

INTERNATIONAL ADVERTISEMENT
IEM - Wintergasse, 52- 3002 Purkersdorf Austria
Tel. +43 2231 68347 - Fax. +43 2231 68402
IEM@inwaredizioni.it

GRAFICA E IMPAGINAZIONE
Margherita Nardelli

STAMPA
ROTO 2000 - Via L. da Vinci, 18/20 - 20080, Casarile (MI)

DISTRIBUZIONE
Parrini & C. S.p.a. - Viale Forlanini, 23 - 20134, Milano

UFFICIO ABBONAMENTI
INWARE Edizioni srl - Via Cadorna, 27/31 - 20032 Cormano (MI)
Per informazioni, sottoscrizione o rinnovo dell'abbonamento:
abbonamenti@inwaredizioni.it
Tel. 02.66504755 - Fax. 02.66508225
L'ufficio abbonamenti è disponibile telefonicamente dal lunedì al venerdì dalle 14,30 alle 17,30
Tel. 02.66504755 - Fax. 02.66508225
Poste Italiane S.p.a. - Spedizione in abbonamento Postale D.L. 353/2003 (conv. In L. 27/02/2004 n. 46) art. 1, comma1, DCB Milano.
Abbonamento per l'Italia: € 49,50 - Abbonamento per l'estero: € 115,00
Gli arretrati potranno essere richiesti, per iscritto, a € 9,00 oltre le spese di spedizione

Autorizzazione alla pubblicazione del Tribunale di Milano n. 647 del 17/11/2003.

© Copyright - Tutti i diritti di riproduzione o di traduzione degli articoli pubblicati sono riservati. Manoscritti, disegni e fotografie sono di proprietà di Inware Edizioni srl.
È vietata la riproduzione anche parziale degli articoli salvo espressa autorizzazione scritta dell'editore. I contenuti pubblicitari sono riportati senza responsabilità, a puro titolo informativo.

Privacy - Nel caso la rivista sia pervenuta in abbonamento o in omaggio, si rende noto che i dati in nostro possesso sono impiegati nel pieno rispetto del D.Lgs. 196/2003. I dati trasmessi a mezzo cartoline o questionari presenti nella rivista, potranno venire utilizzati per indagini di mercato, proposte commerciali, o l'inoltro di altri prodotti editoriali a scopo di saggio. L'interessato potrà avvalersi dei diritti previsti dalla succitata legge. In conformità a quanto disposto dal Codice di deontologia relativo al Trattamento di dati personali art. 2, comma 2, si comunica che presso la nostra sede di Cormano Via Cadorna 27, esiste una banca dati di uso redazionale. Gli interessati potranno esercitare i diritti previsti dal D.Lgs. 196/2003 contattando il Responsabile del Trattamento Inware Edizioni Srl (info@inwaredizioni.it).

RICHIESTE DI ASSISTENZA
Per richiedere assistenza o chiarimenti sugli articoli pubblicati, vi preghiamo di utilizzzare il servizio MIP compilando l'apposito modulo on-line all'indirizzo www.farelettronica.com/mip.

COLLABORARE CON FARE ELETTRONICA
Le richieste di collaborazione vanno indirizzate all'attenzione di Maurizio Del Corso *(m.delcorso@inwaredizioni.it)* e accompagnate, se possibile, da una breve descrizione delle vostre competenze tecniche e/o editoriali, oltre che da un elenco degli argomenti e/o progetti che desiderate proporre.

ELENCO INSERZIONISTI

A.R.I. sezione Pescara **pag. 25**
Via delle Fornaci, 2 - 65125 Pescara (PE)
Tel. 085.643192 - www.aripescara.org

Area Rebus **pag. 33**
Via N. Sauro, 32 - 45100 Rovigo
Tel. 0425.27401 - www.arearebus.com

Artek Electronic Solutions **pag. 61**
P.zza Pirazzoli, 2 - 40020 Sasso Morelli (BO)
Tel. 0542.643192 - www.artek.it

Blu Nautilus **pag. 39, 77**
P.zza Tre Martiri, 24 - 47900 Rimini
Tel. 0541.439575 - www.blunautilus.it

Blu Press **pag. 29**
Via Cavour 65/67 - 05100 Terni (TR)
Tel. 0744.433606 - www.blupress.it

Comune di Scandiano **pag. 55**
Piazza Trampolini, 1 - 42019 Scandiano (RE)
Tel. 0522.764290 - www.fierascandiano.it

E.R.F. **pag. 111**
L.go Fiera della Pesca, 11 - 60100 Ancona
Tel. 0733.780815 - www.erf.it

Farnell Italia **pag. 17**
Corso Europa, 20-22 - 20020 Lainate (MI)
Tel. 02.939951 (401)

FIERA MILANO TECH **pag. 57**
via Gattamelata 34 - 20149 Milano (MI)
Tel. 02.3264.803 - www.fieramilanotech.it

FRAMOS Electronic Vertriebs GmbH **pag. 43**
Zugspitz str. 5 - Haus C - 82049 Pullach b.
(Monaco - Germania)
Tel. 0396899635 - www.framos.it

Futura Elettronica **pag. 83, 95, 105**
Via Adige, 11 - 21013 Gallarate (VA)
Tel 0331.792287 - www.futuranet.it

Istituto Internazionale di Ricerca **pag. 37, 91**
Via Forcella, 3 - 20144 Milano
Tel 02.83847272 - www.iir-italy.it

Microchip Technology **pag. 11, 13**
Via S. Quasimodo, 12 - 20025 Legnano (MI)
Tel. 0331.7426110 - www.microchip.com

MikroElektronika **pag. 107**
Admirala Geprata 1B - 11000 Belgrade
Tel. +381.11.30663787 - www.microe.com

Millenium Dataware **pag. 23**
Corso Repubblica, 48 - 15057 Tortona (AL)
Tel 0131.860254 - www.mdsrl.it

PCB-Pool **pag. 85**
Bay 98 IRL Shannon Freezone Shannon, Co. Clare (Ireland)
Tel. 02.64672645 - www.beta-layout.com

R.C.C. **pag. 99**
Via G. Di Vittorio 19 - 20097 San Donato Milanese (MI)
Tel. 02.51876194 - www.rccitaly.com

RS Components **IV cop**
Via M. V. De Vizzi, 93/95 - 20092 Cinisello Balsamo (MI)
Tel 02.660581 - rswww.it

Scuola Radio Elettra **III cop**
Via Biturgense, 104 - 00185 Cerbara di Città di Castello (PG)
Tel 075.862911 - www.scuolaradioelettra.it

Wireless **pag. 63**
Via Milanese, 20 - 20099 Sesto San Giovanni (MI)
Tel 02.48517925 - www.gowireless.it



Note dalla redazione

Maurizio Del Corso
m.delcorso@inwaredizioni.it

Nuove iniziative vi aspettano...

Le ferie sono finite da un pezzo, il transitorio post-ferie può considerarsi esaurito e siamo oramai al lavoro a pieno regime. Da questo numero ho l'onore e il privilegio di assolvere a un compito molto arduo passatomi dall'ottimo Tiziano Galizia: la guida della Rivista. Abbiamo la fortuna di partire da solide fondamenta su cui costruiremo un progetto editoriale ambizioso ma di sicuro interesse che coinvolgerà attivamente anche voi lettori. Seguite la rivista e scoprirete via via quello che abbiamo pensato per voi...

Tanto per iniziare il prossimo mese troverete in edicola, oltre al consueto numero di Fare Elettronica, anche un numero speciale dedicato al progetto e alla costruzione di una macchina a controllo numerico (CNC) con cui potrete realizzare i vostri circuiti stampati. Per ora, se siete abbonati, potete prenotare la vostra copia sul sito della rivista (www.farelettronica.com). Come inizio, niente male!

Per quanto riguarda il numero che state sfogliando ecco quello che vi aspetta. Il progetto di una lampada allo stato solido realizzata con diodi LED ad alta luminosità e di un programmatore orario con un display LCD sono solo due delle realizzazioni presentate questo mese. Un interessantissimo speciale sui sistemi di controllo x-by-wire vi spiegherà come in una moderna autovettura le parti di controllo meccaniche sono state soppiantate dall'elettronica. Continua la trattazione dei trasduttori di posizione, la progettazione degli equalizzatori ambientali, l'uso dei display alfanumerici con Cubloc e molti altri articoli non spettano altro che essere divorati!

Prima di lasciarvi vi rinnovo l'invito ad usare il form dei contatti sul sito della Rivista per inviarci i vostri commenti, i vostri suggerimenti, le vostre proposte e le vostre critiche. È una attività fondamentale, perché solo conoscendovi possiamo realizzare la rivista "su misura" per voi. Buona lettura!

abbonati
SUBITO!!!
e risparmi
fino al
40%



1 semplice
2 comodo
3 immediato

FALLO SU
www.farelettronica.com/abbonamento

INOLTRE SCELTI PER VOI







Rubrica di notizie e novità dal mondo dell'elettronica.

News

Austriamicrosystems presenta l'AS1524/25, un convertitore A/D micro-power da 12-bit e 150ksps



Austriamicrosystems ha allargato il suo portafoglio di convertitori A/D con l'aggiunta del nuovo AS1524, un convertitore da 12-bit, ultra low-power con ingresso singolo differenziale, e l'AS1525 a doppio ingresso single-ended. Questi dispositivi che al bassissimo consumo uniscono una velocità di campionamento di 150ksps e un contenitore TDFN a 8pin di solo 3x3mm, rappresentano la scelta ideale per l'impiego in apparecchiature alimentate a batteria e sistemi di acquisizione portatili.

Gli AS1524/25, consumano solo 350µA (3V) alla massima velocità di campionamento di 150ksps. Un sistema di shutdown avanzato provvede a porre il dispositivo in modalità sleep tra le conversioni, consentendo una significativa riduzione del consumo a velocità di campionamento ridotte. A 1ksps il consumo scende a 2,5 µA, mentre è di 200na in shutdown.

L'interfaccia è compatibile SPI, QSPI e Microwire, e consente un accesso veloce unitamente ad una ridotta occupazione dello spazio sulla scheda. Entrambi i dispositivi dispongono di un clock interno ma anche di un ingresso di clock esterno per garantire la massima flessibilità, ed operano con una tensione di alimentazione compresa tra 2,7V e 5,25V su un campo di temperatura che si estende tra -40°C e +85°C.

Codice MIP 900338

Atmel presenta un sistema di adattamento dell'impedenza di antenne AM/FM

Atmel® Corporation annuncia la disponibilità dell'ATR4251, un amplificatore di antenna ad alta integrazione per sistemi AM/FM che consente una eccezionale qualità di ricezione se utilizzato con varie tipologie di antenne. L'amplificatore con adattatore di impedenza dell'ATR4251 consente di compensare la diminuzione del segnale causate dalle modeste dimensioni delle moderne antenne e dalla lunghezza del cavo di connessione.

L'ATR4251 consente una ricezione chiara e senza distorsione anche in condizioni estremamente difficili create dalla rapida variazione dell'intensità di campo che caratterizza i sistemi di ricezione mobili, grazie alla sua ampia dinamica. Questa elevata qualità di ricezione è dovuta all'equilibrio tra il guadagno in AM che può raggiungere i 35bB, e



l'ottimo comportamento in FM che vede una IP3 di 148dBuV, a cui si aggiunge un basso rumore che risulta ottimale in questa specifica applicazione.

Un amplificatore a basso rumore per l'FM e un amplificatore per l'AM vengono impiegati per gestire in modo ottimale le due diverse gamme di ricezione. L'amplificatore AM copre tutta la banda di frequenze compresa tra 150kHz e 30MHz che comprende i sistemi DRM, mentre l'LNA per la FM opera su una banda di frequenza compresa tra 70MHz e 166MHz che include le weather band e l'HD Radio.

Un controllo automatico del guadagno separato

consente la regolazione ottimale e previene il sovraccarico degli amplificatori in presenza di segnali di notevole ampiezza che potrebbero causare il blocco della ricezione.

Codice MIP 900339

Fujitsu rilascia un CODEC HDTV H.264 con memoria integrata e consumi ultraridotti



Fujitsu Microelectronics Europe ha annunciato l'introduzione di un nuovo chip capace di comprimere e decomprimere in tempo reale segnali video ad alta definizione H.264 e segnali audio Dolby Digital. Basato su un algoritmo H.264 sviluppato da Fujitsu Laboratories Limited, il chip MB86H51 integra le capacità di elaborazione audio-video ad alte prestazioni con una tecnologia di processo da 90nm, ed è il primo in assoluto a supportare lo standard H.264 High Profile, Level 4.0 utilizzato nelle apparecchiature consumer HD H.264 di nuova generazione.

Le applicazioni del nuovo chip comprendono registrazione audio in alta risoluzione, riproduzione e trasmissione di video in alta definizione su prodotti audio/video come camcorder portatili, registratori basati su hard disk e apparecchiature di rete residenziali. Il dispositivo è adatto anche ai PC multimediali e ai sistemi per la sicurezza e la videosorveglianza.

L'MB86H50/51 è in grado di comprimere e decomprimere in tempo reale segnali video ad alta definizione (rispettivamente fino a 1440 pixel x 1080 linee e fino a 1920 pixel x 1080 linee) in formato H.264 High Profile Level 4.0, e contemporaneamente comprimere e decomprimere audio in formato Dolby Digital.

Si tratta anche del primo CODEC HD H.264 a integrare due FCRAM da 256Mbit all'interno di un package fpBGA. Le ridotte dimensioni del package BGA e l'integrazione della memoria hanno permesso di raggiungere un fattore forma più piccolo con consumi elettrici estremamente limitati.

Codice MIP 900340

IDT introduce i primi "mobile multimedia interconnect™" per dispositivi pda e per terminali mobili di fascia alta

IDT introduce la prima soluzione sincrona Mobile Multimedia Interconnect™ (M2I) totalmente ottimizzata per le applicazioni legate a personal digital assistant e terminali mobili di fascia alta. Grazie al significativo progresso architetturale rispetto alle interfacce di generazione precedente, M2I opera fino a sei volte più velocemente e assicura consumi ridotti fino al novanta percento. Gli utilizzatori di terminali di fascia alta possono riprodurre i loro contenuti multimedia in modo affidabile e con livelli di qualità senza precedenti, a fronte di un impatto minimo sui "tempi di conversazione" e sulla vita della batteria.

La sofisticata interfaccia M2I garantisce la possibilità di supportare funzionalità specifiche aggiuntive quali Bluetooth, WiFi e GPS. M2I è stato progettato per lavorare con processori applicativi e processori bandabase dotati di interfaccia ADM (address-data mux).

L'interfaccia ADM vanta un numero di ingressi e di uscite più contenuto e una banda nettamente superiore rispetto ad altre soluzioni quali le RAM dual-port asincrone standard e le interfacce seriali embedded normalmente presenti nei terminali mobili di fascia alta, liberando risorse per supportare altre funzionalità.



Oltre a questo, M2I prevede otto I/O dinamicamente programmabili che il processore può utilizzare per controllare altri dispositivi.

Le elevate prestazioni e i bassi consumi sono legati a uno schema di temporizzazione sincrona che consente di ricorrere a un contatore interno che evita processi di indirizzamento multiplo. Grazie a questa soluzione, M2I è in grado di processare 64 Kbit di dati in soli 4.000 cicli, contro gli 8.000 cicli richiesti dai dispositivi di generazione precedente, assicurando un incremento globale di prestazioni pari a sei volte.

Codice MIP 900341

Server cost-effective per la connessione in rete di dispositivi seriali



L'LS110 è un server a porta singola e a basso costo che connette dispositivi seriali RS-232 come stampanti, bilance, apparecchiature medicali, macchine per linee di produzione, lettori di codici a barre, lettori di badge e carte di credito, terminali POS, e altri terminali per acquisizione dati, ad una rete Ethernet 10'Base-T. L'unità supporta funzioni avanzate come il protocollo per il controllo di porte telnet COM (rispondente alla normativa RFC2217), ed è protetta contro le sovratensioni. Sul lato seriale l'LS110 dispone di un connettore DB-9, che gestisce tensioni di ingresso comprese tra 7,5V e 15V, e supporta una velocità di trasferimento dati fino a 115kbps. Anche la porta seriale è protetta contro le sovratensioni che potrebbero causare l'interruzione del funzionamento del sistema.
Nella modalità TCP, l'LS110 supporta il protocollo per il controllo di porte telnet COM, utilizzato per accedere a dispositivi seriali connessi alla rete. Questa opzione abilita l'applicazione a settare i parametri delle porte seriali remote (baud rate, controllo di flusso, ecc.), a rilevare le variazioni dei segnali di linea, e a trasmettere e ricevere dati. L'LS110 è fornito con una copia del software COM Port Redirection Software supportata dalla relativa licenza, che consente di utilizzare il programma basato sulla comunicazione seriale pre-esistente.

Codice MIP 900329

Si espande la famiglia di starter kit per i microcontrollori SuperH

Renesas Technology Europe ha allargato la sua linea di Renesas Starter Kits (RSK) con l'introduzione di tre kit da utilizzare con il suo microcontrollore SuperH (SHC). Gli RSK di Renesas, oltre al basso costo presentano una notevole semplicità di impiego, e vengono forniti completi di scheda, display, debugger e software. I kit RSK7211 e RSK7086 e RSK7203plus sono già disponibili, e si affiancano agli esistenti strumenti di sviluppo dedicati all'SHC che comprendono l'RSK7124, l'RSK7201 e la scheda di comunicazione USB/Ethernet. Il Kit RSK7211 supporta il dispositivo Flash più veloce oggi disponibile, l'SH7211F, dotato di una MONOS Flash da 512Kbyte con tempo di accesso di 12,5ns, un ADC da 12 bit, e due sofisticate unita tiimer che presentano una risoluzione di 12,5ns. Il kit RSK7086 supporta l'SH7086F che rappresenta la soluzione ideale per il controllo di motori grazie alle varie versioni disponibili che presentano un numero di pin compreso tra 48 e 176. Il kit RSK7203plus viene utilizzato con le famiglie di microcontrollori high-end e comprende il debugger E10A-Lite debugger che grazie al gran numero di funzioni disponibili, consente lo sviluppo di sistemi complessi in modo semplice e rapido. Il kit integra un drive TFT e una connettività Ethernet.

Codice MIP 900327

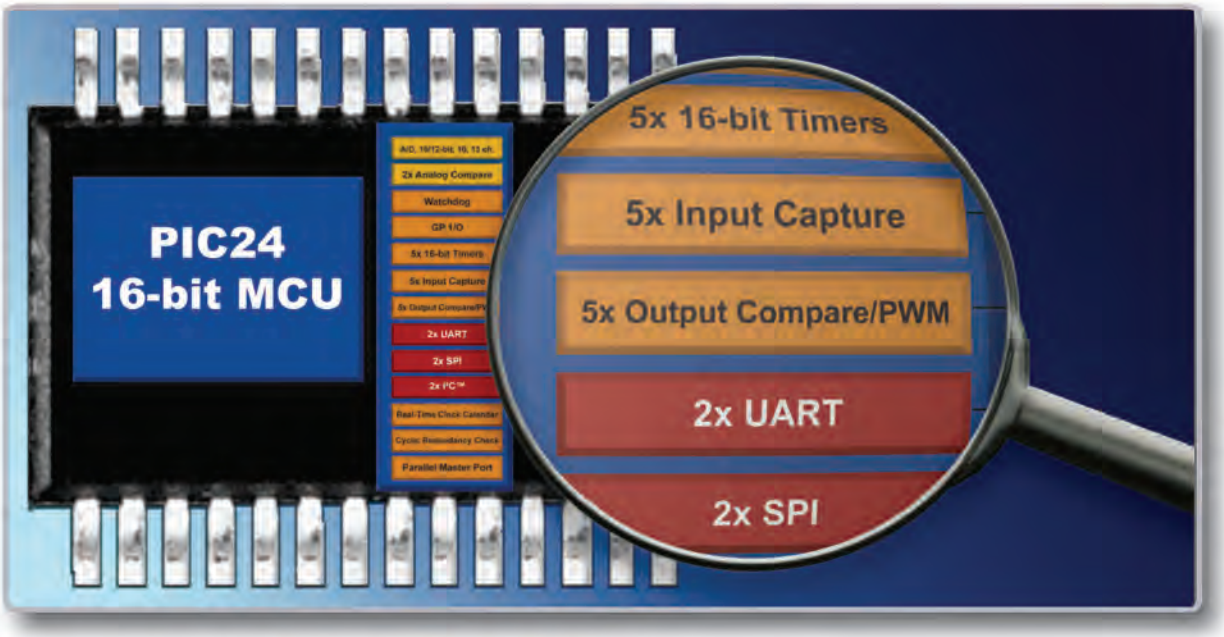
Austriamicrosystems presenta l'AS5046, un encoder rotante magnetico da 12-bit



Austriamicrosystems, presenta l'AS5046, un nuovo encoder rotante magnetico con risoluzione di 12-bit che è in grado di rilevare la rotazione, l'ngolo e la distanza verticale di un magnete posto sopra il dispositivo. Il campo di misura della distanza verticale è di alcuni millimetri, e questo ne fa la soluzione ideale per la realizzazione di comandi contactless multiassiali. L'AS5046 è un completo system-on-a-chip, che combina gli elementi Hall e il processore del segnale in un solo chip. La possibilità di rilevare oltre alla posizione anche l'inclinazione del magnete, consente la realizzazione della funzionalità "pulsante", fattore che apre nuove possibilità nella realizzazione di interfacce come i joystick. L'AS 5046 è l'ultimo nato della famiglia di encoder magnetici rotanti contactless di austriamicrosystems, disponibile con risoluzioni di 8, 10 e 12-bit, con differenti interfacce, ed in grado di operare alla temperatura di 150°C. Il dispositivo comunica con il controllore esterno grazie ad una interfaccia seriale a due fili; per applicazioni standalone è disponibile una uscita raziometrica programmabile. L'AS5046 è disponibile in un contenitore SSOP16 lead-free, e può operare con tensioni di alimentazione di 3,3V o di 5,0V su un campo di temperatura ambiente compreso tra -40°C e +125°C.

Codice MIP 900342

Guardate quante periferiche potete mappare sui pin di I/O



Con la prestazione di Peripheral Pin Select Microchip potrete disporre della flessibilità necessaria per personalizzare il vostro microcontroller.

La famiglia di microcontroller a 16-bit PIC24 di Microchip include dei dispositivi miniaturizzati a 28-pin che offrono fino a 64 KB di Flash e a 8 KB di RAM, con un set di periferiche che può essere personalizzato dinamicamente in funzione della vostra applicazione!

3 SEMPLICI PASSI PER PARTIRE...

1. **Seminari web GRATUITI sui 16-bit**
2. **Campionature GRATUITE**
3. **Sconti esclusivi sui tool di sviluppo**

Low Pincount 16-Bit Microcontrollers

Device	Pins	RAM (KB)	Flash (KB)	Features Include
PIC24FJ16GA002	28	4	16	5x 16-bit Timers
PIC24FJ16GA004	44	4	16	5x Output Compare/PWM
PIC24FJ32GA002	28	8	32	5x Input Capture
PIC24FJ32GA004	44	8	32	Real Time Clock Calendar
PIC24FJ48GA002	28	8	48	2x UART
PIC24FJ48GA004	44	8	48	2x SPI
PIC24FJ64GA002	28	8	64	2x I²C™
PIC24FJ64GA004	44	8	64	Parallel Master Port
PIC24HJ12GP201	18	1	12	2x Analog Comparators
PIC24HJ12GP202	28	1	12	10/12-bit ADC

Acquistate e programmate i vostri dispositivi PIC24 a 16-bit e i relativi tool di sviluppo al link **www.microchipdirect.com**



Prezzi speciali e promozioni sul 16-bit 28-pin Development Kit e MPLAB® ICD 2 In-Circuit Debugger all'indirizzo.

Visitate subito il sito www.microchip.com/LPC

microchip DIRECT
www.microchipdirect.com

Now Pb-free!
RoHS Compliant

MICROCHIP
www.microchip.com/LPC

Codice MIP 268011

Il nome e il logo Microchip, il logo Microchip e MPLAB sono marchi registrati da Microchip Technology Incorporated negli USA e in altre nazioni. Tutti gli altri marchi qui menzionati appartengono ai rispettivi titolari. © 2007, Microchip Technology Inc. ME177/Ita/08.07

Rubrica di appuntamenti, manifestazioni ed eventi nel mondo dell'elettronica

EXPO

Gli eventi di Ottobre 2007

13-14 Ottobre 2007

EXPO RADIO & INFORMATICA

A Potenza, presso l'area industriale di Tito Scalo, si svolgerà il 13 e 14 Ottobre 2007 l'Expo Radio & Informatica, la mostra mercato dedicata all'elettronica e all'informatica, compagni ormai indispensabile nel lavoro e negli hobby, organizzata dall'EFAB Ente Fiera Autonomo di Basilicata. Ideata per consentire, agli operatori del settore, una migliore valutazione dell'offerta delle nuove tecnologie. Inoltre offre anche ad un pubblico più eterogeneo, in cerca di buone occasioni commerciali, tutta l'esperienza e la professionalità acquisita nel campo delle telecomunicazioni e dell'informatica. Saranno presenti espositori provenienti dall'intero territorio nazionale con: Radioricetrasmettitori per CB – OM, Computer, Telefonia, Antenne e Parabole per TV Sat, Hi-Fi Car, Strumenti di misura e utensili vari, Surplus, Radio d'epoca, Editoria Specializzata.

Informazioni utili

Ente Fiera Autonomo Basilicata
Tito Scalo (PZ)
Orario: dalle 09:00 alle 18:00
Organizzazione: EFAB
www.efab.it

Codice MIP 800018

27-28 OTTOBRE 2007

3.a MOSTRA REGIONALE DELL'ELETTRONICA

Mostra mercato dedicata all'elettronica, informatica, radiantistica, materiale di consumo, video giochi, telefonia ed editoria specializzata.

Informazioni utili

Centro Fieristico Scandiano
Scandiano (RE)
Orario: dalle 09:00 alle 18:00
Organizzazione: Comune di Scandiano
www.fierascandiano.it

Codice MIP 800020

20-21 Ottobre 2007

EXPORADIO ELETTRONICA



Da molti anni l'elettronica è entrata a far parte del nostro quotidiano, non solo in ambito professionale ma anche fra le mura di casa: dagli elettrodomestici ai giochi dei bambini, dagli antifurto alla Tv satellitare. Expo Elettronica a Faenza è uno degli appuntamenti più noti e frequentati sia per l'elettronica di consumo sia per prodotti rivolti ad un pubblico più esperto.

Informazioni utili

Faenza Fiere - Faenza
Orario: dalle 09:00 alle 18:00
Organizzazione: Blu Nautilus
www.blunautilus.it

Codice MIP 800019

6-7 OTTOBRE 2007

III° ELETTRO@BIT

Mostra mercato dedicata all'elettronica, informatica, radiantistica, materiale di consumo, video giochi, telefonia ed editoria specializzata.

Informazioni utili

Centro Affari - Arezzo
Orario: dalle 09:00 alle 18:30
Organizzazione: GEDIT

Codice MIP 800023

13-14 Ottobre 2007

Radioelettra Informatica

Radiantismo-C.B.-Computer-Hardware-Software-Elettronica Di Consumo-Componentistica-Satellitare-Telefonia-Surplus-Radio D'epoca-Editoria.

Informazioni utili

Ferrara Fiere Congressi - Ferrara
Orario: dalle 09:00 alle 18:00
Organizzazione: Ferrara Fiere Congressi
www.natalissimo.com

Codice MIP 800021

6-7 OTTOBRE 2007

COMPUTERFEST & RADIOAMATORE

Mostra mercato dedicata all'elettronica, informatica, radiantistica, materiale di consumo, video giochi, telefonia ed editoria specializzata. Dal sito Compendio è possibile scaricare le riduzioni per il biglietto d'ingresso.

Informazioni utili

Firenze – Stazione Leopolda
Orario: dalle 09:00 alle 18:00
Organizzazione: Compendio Fiere
www.compendiofiere.it

Codice MIP 800022

13-14 Ottobre 2007

ELETTROEXPO

L'Ente Autonomo Fiere Verona, presenta la **36^** edizione di **Elettroexpo**, **storica** manifestazione dedicata al mondo dell'elettronica e del radiantismo, che si terrà in data **12-13 maggio 2007** presso il quartiere fieristico di Verona. La manifestazione, per la quale si propone l'ormai consueta e collaudata formula "**esposizione-mercato**", raccoglie la migliore offerta dei settori dell'informatica, elettronica, componentistica e delle radio comunicazioni.

Informazioni utili

Verona fiere
Orario: dalle 09:00 alle 18:00
Organizzazione: Verona Fiere
www.veronafiere.it

Codice MIP 800024

27-28 Ottobre 2007

FIERA DELL'INFORMATICA E DELL'ELETTRONICA

Sabato 27 e domenica 28 Ottobre 2007 si svolgerà la F.I.E.Ra. - **F**iera dell'**I**nformatica, dell'**E**lettronica e del **Ra**diantismo. Grandi occasioni per tutti, a partire dai supporti per registrazioni audio/video per finire con l'informatica e il radiantismo.



Uno spazio considerevole è stato riservato all'hobbistica e all'elettronica - strumenti di misura e apparecchiature elettrico-elettroniche sono ben rappresentate da molti espositori specializzati.

Informazioni utili

Centro Servizi Rovigo Fiere - Rovigo
Orario: dalle 09:00 alle 19:00
Organizzazione: Area Rebus
www.arearebus.com

Codice MIP 800025

Conferenza MASTERs Europe 2007

6- 8 novembre 2007 – Monaco di Baviera

MASTERs Europe è una conferenza tecnica di approfondimento di due giorni in cui gli ingegneri possono acquisire una serie di conoscenze specifiche e condividere le proprie esperienze progettuali sia con gli esperti Microchip sia con altri ingegneri provenienti da tutto il mondo.

- Prima conferenza MASTERs europea
- Formazione tecnica per tutti i livelli di preparazione
- Eventi e attività pomeridiane informali
- Sconti fino al 50% sui tool di sviluppo
- 34 classi, strutturate su 5 livelli di preparazione
- Certificato MASTERs per i partecipanti titolati
- Classi di teoria e pratica

Per prenotare un posto, collegatevi al link: http://techtrain.microchip.com/EUMASTERs

MICROCHIP
www.microchip.com

Il nome e il logo Microchip sono marchi registrati da Microchip Technology Incorporated negli USA e in altre nazioni.
©2007 Microchip Technology Inc. Tutti i diritti riservati. ME181/Ita/09.07

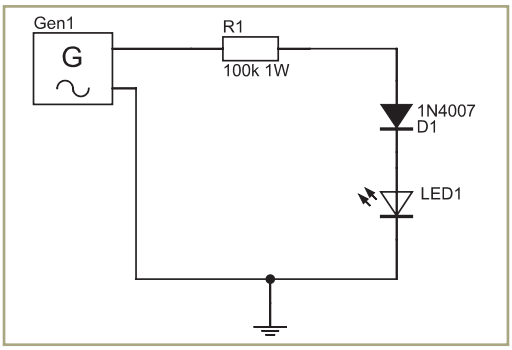
Codice MIP 800011

Dalla Redazione di Fare Elettronica una raccolta di idee ed applicazioni da tenere sempre a portata di mano.

Idee di progetto

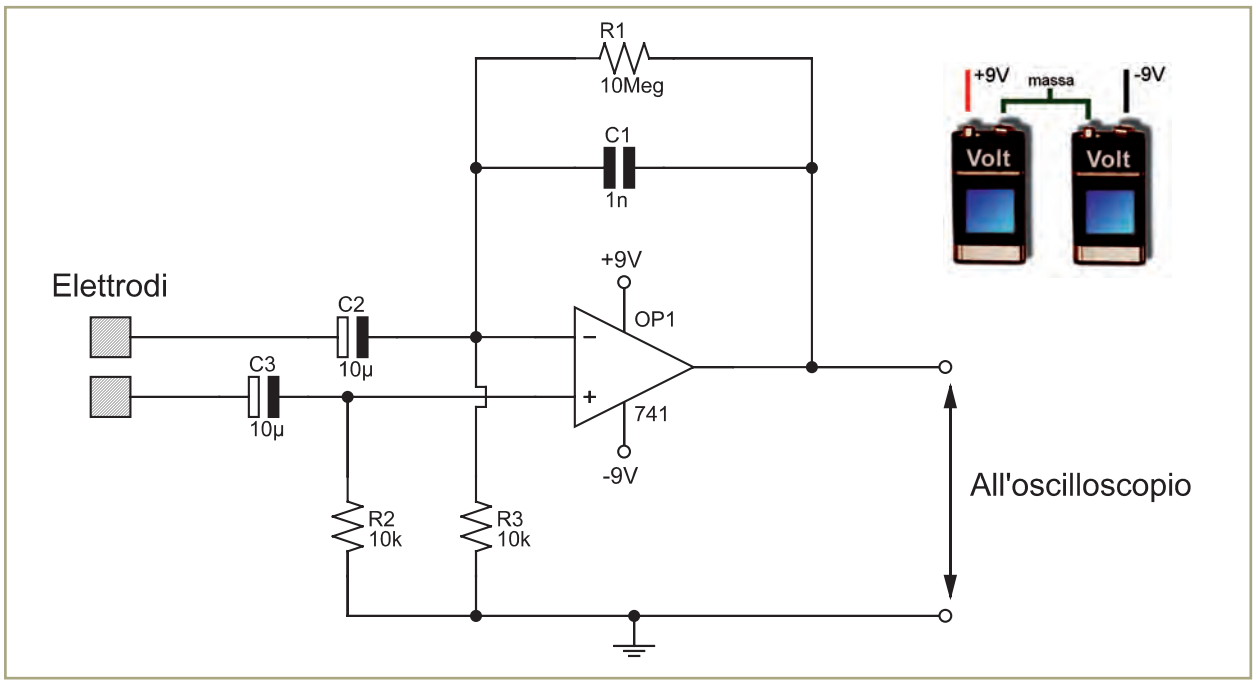
95 - Alimentare un diodo led a 230 Volt

È possibile alimentare un diodo Led direttamente dalla tensione di rete di 230 Volt, naturalmente con le dovute precauzioni. Utilizzando un semplice resistore da 100K con dissipazione da 1W, in aggiunta ad un normale diodo 1N4007, esso si illumina con una corrente di circa 3mA. Il diodo serve per bloccare le tensioni inverse di rete, che nuocerebbero al Led. È possibile abbassare il valore della resistenza sino a circa 10K, a patto di aumentarne la potenza a valori prossimi a 10W, operazione comunque non tanto conveniente.



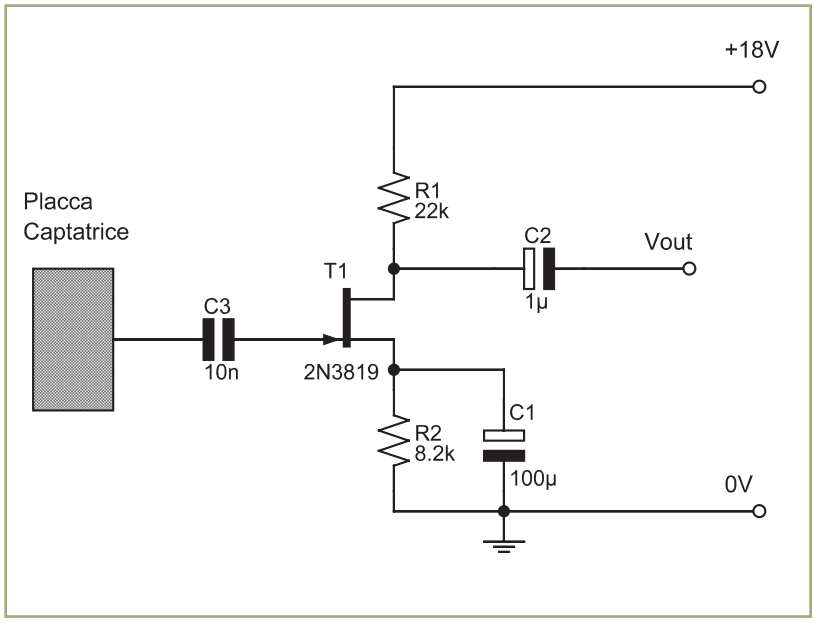
96 - Semplicissimo E.C.G.

Presentiamo lo schema di un elettrocardiografo molto semplice, che naturalmente non può essere utilizzato per scopi professionali. Il suo contributo può però rivelarsi utile per focalizzare l'alta impedenza degli amplificatori operazionali, legata anche all'elevata amplificazione. Lo schema riesce a "catturare" gli impulsi elettrici prodotti dal cuore. La configurazione adottata quale "amplificatore differenziale" permette di rigettare i rumori indesiderati (frequenza a 50 Hz o altri segnali spuri) ed accettare solo la tensione prodotta dal muscolo cardiaco. L'amplificazione raggiunta ha un fattore pari a 1000. Si consiglia di applicare gli elettrodi sul petto del paziente mediante l'uso di un gel umido o acqua, per facilitare il passaggio della corrente. L'alimentazione è ricavata da due pile da 9 volt, collegate in serie. Il nodo centrale di queste rappresenta la massa comune al circuito.

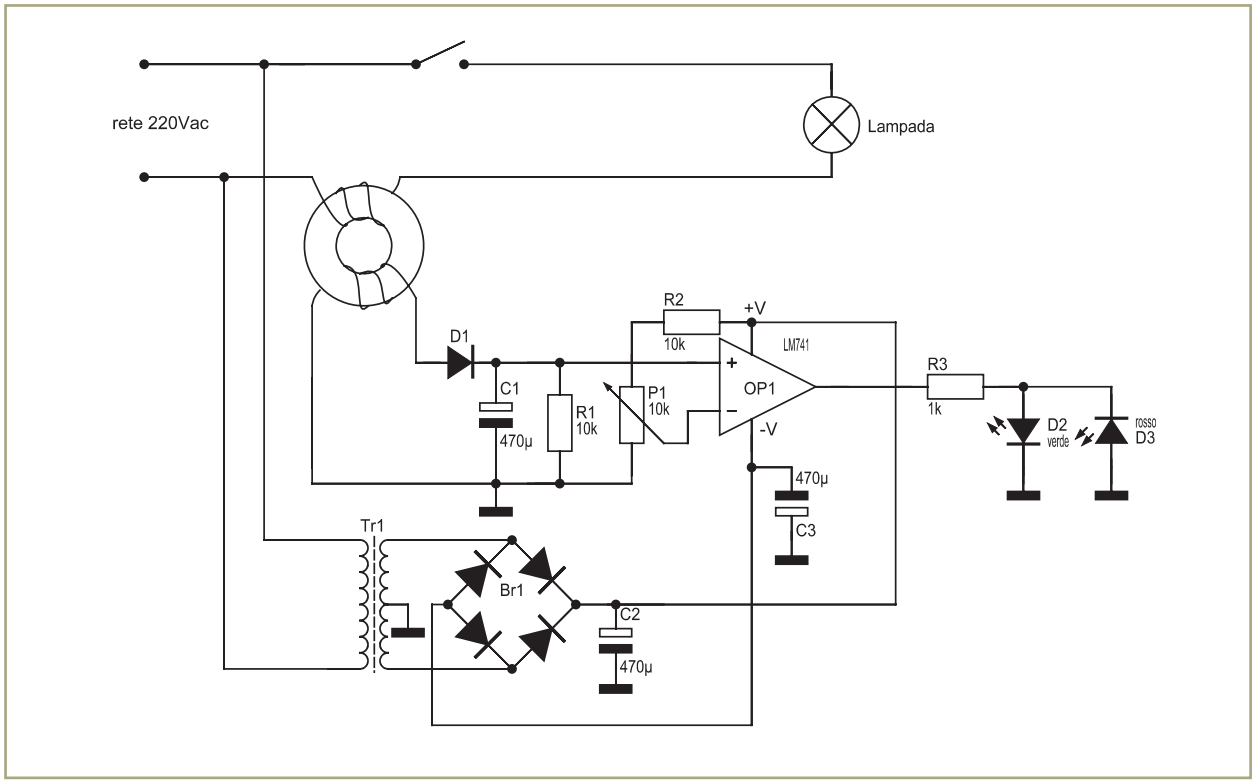


97 - Semplicissimo Elettrometro

Sfruttando l'altissima impedenza dei transistor FET si può realizzare un semplicissimo elettrometro, ossia uno strumento in grado di rivelare cariche elettrostatiche nelle immediate vicinanze dell'ingresso.
Il fatto che il gate del Fet non sia collegato ad alcuna resistenza verso massa, contribuisce ad aumentare l'impedenza, e quindi, la sensibilità dello strumento.
Per visualizzare la tensione delle cariche si deve collegare un voltmetro d.c. all'uscita del circuito e impostare il valore di fondo scala in una posizione più sensibile possibile.



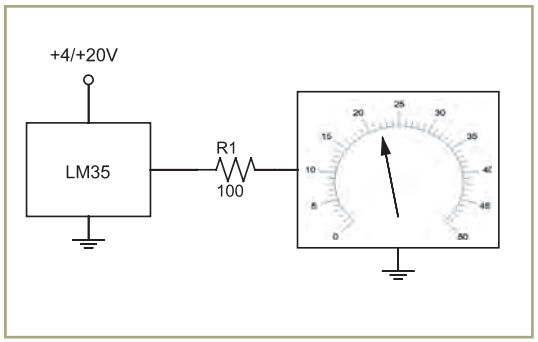
98 - Monitorare lo stato di una lampadina

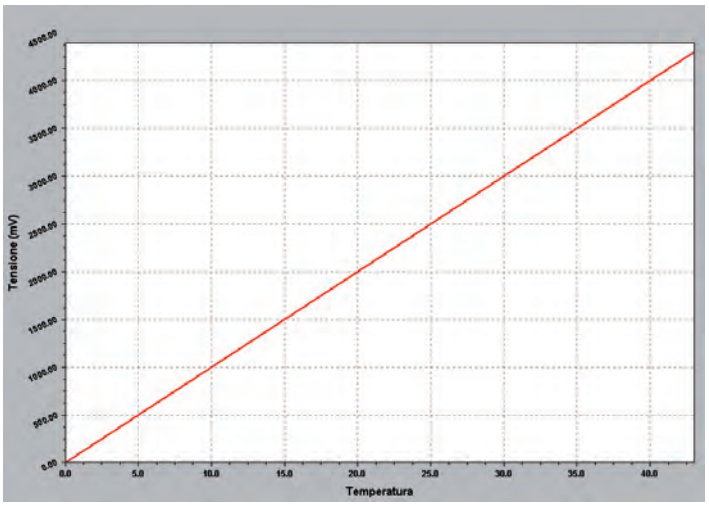


Il circuito consente di accendere un LED verde se la lampada è accesa e un LED rosso se la lampada è spenta. Mediante il nucleo toroidale (va bene uno di qualsiasi tipo) all'anodo di D1 è presente una tensione solo quando nell'avvolgimento passa corrente (quindi solo se la lampada è accesa). D1 e C1 raddrizzano la tensione che viene portata al piedino + del comparatore costituito dall'operazionale. La tensione viene così comparata con una tensione di riferimento ottenuta dalla partizione della tensione di alimentazione mediante P1 e R2. In fase di taratura si dovrà regolare P1 affinché con la lampada accesa il led verde sia acceso. Il numero di spire sul nucleo toroidale può variare da 2 a qualche decina (maggiore è il numero di spire maggiore sarà la tensione al morsetto + dell'operazionale quando la lampada è accesa). Il trasformatore dovrà avere un primario da 220Vac e un secondario di 9+9Vac.

99 - Termometro con LM35

Il circuito integrato LM35 costituisce un preciso termometro analogico, la cui tensione d'uscita è direttamente proporzionale alla temperatura d'ambiente (in gradi Celsius).

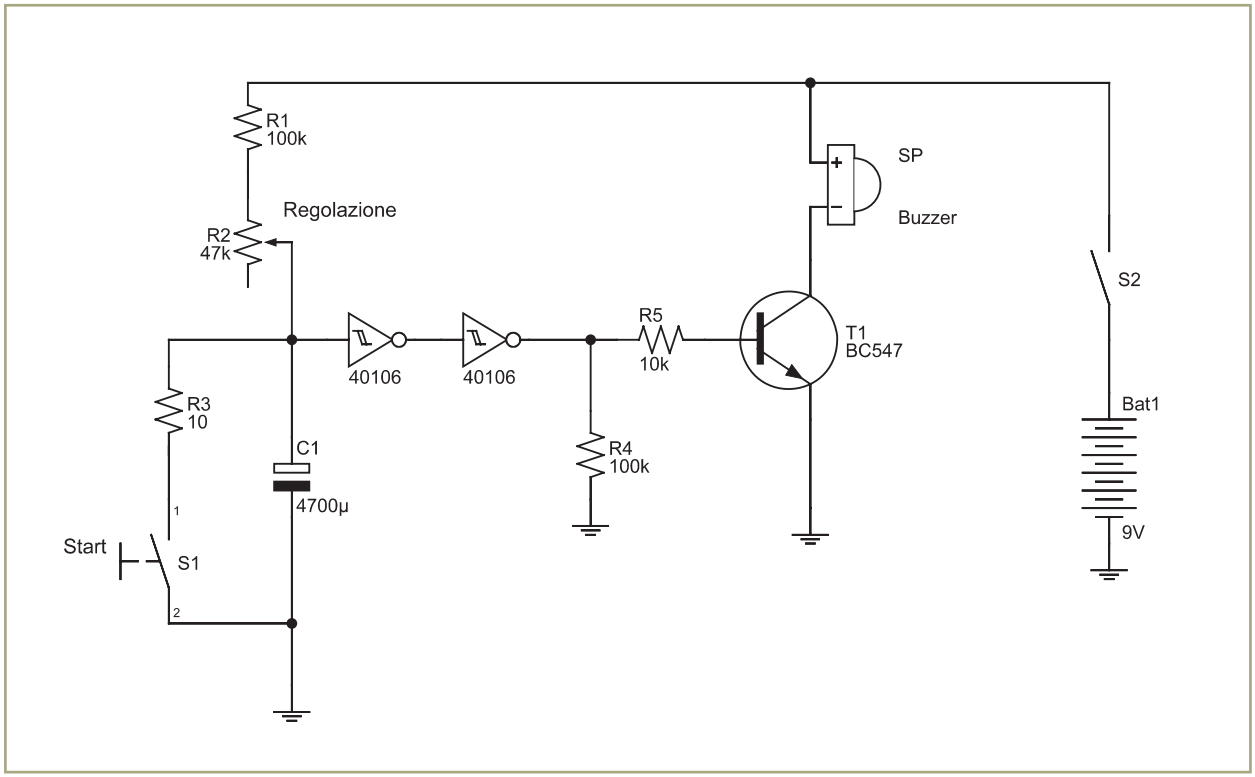




Esso non richiede alcuna calibrazione esterna e il suo campo d'azione spazia da –55° a +150°. Fornisce una tensione di 10mV per ogni grado centigrado di temperatura.
Utilizzato in congiunzione di un voltmetro analogico o digitale, può costituire un valido termometro per la misura della temperatura ambientale interna ed esterna. Nel caso si utilizzasse un voltmetro analogico, è consigliabile disegnare una nuova scala graduata con i valori delle temperature.

100 - Timer per uovo sodo

In cucina è spesso utile disporre di un timer che segni il tempo. E' il caso tipico per la cottura delle uova sode. Il circuito proposto, dopo l'attivazione, resta in una pausa d'attesa per circa tre minuti. Allo scadere del quale, il buzzer collegato inizia a suonare. Il tempo d'attesa è regolabile attraverso un potenziometro, mentre la riattivazione del circuito avviene premendo il pulsante in parallelo al condensatore. E' consigliabile racchiudere il circuito in un piccolo contenitore, dal quale fuoriescano il potenziometro ed il pulsante.





Farnell | Tutta l'elettronica con un click

Oltre 380.000 prodotti elettronici disponibili a magazzino cliccando su **www.farnell.com**

- Prodotti disponibili e **consegna in 24 ore**
- Oltre **153.000 prodotti** provenienti dagli Stati Uniti
- **Novità!** 16.000 nuovi prodotti **"su richiesta"**
- **Novità!** Le più innovative tecnologie wireless su **www.electronicsdesignworld.com**
- **25.000 nuovi prodotti** aggiunti al catalogo

live edge
Electronic Design for the Global Environment

Partecipa al primo concorso che va oltre la progettazione elettronica e vinci **$100.000**
www.live-edge.com/it

www.farnell.com **02 93995.200**

A Premier Farnell Company

Codice MIP **268017**

Farnell

Interfaccia MIDI

Nel presente articolo viene descritta una particolare interfaccia MIDI per PC che ha una particolarità in più che senz'altro verrà apprezzata dai musicisti più esigenti.
Proseguite nella lettura e scoprirete di cosa si tratta.

L'avvento dell'elettricità ha portato enormi cambiamenti nella vita quotidiana, interessando ogni settore. Anche il mondo della musica ha risentito della presenza dell'elettricità. Così musicisti ed elettronici si sono ingegnati per creare strumenti musicali in grado di generare i suoni mediante un flusso modulato di elettroni. Sostanzialmente si è cercato di creare strumenti musicali elettrici; e si pensi ai primi organi elettrici.
Con l'invenzione dei tubi termoionici, prima, e del transistor, successivamente, gli strumenti musicali elettrici sono diventati più evoluti e più ricchi di prestazioni. Ecco che è nata la denominazione di strumenti musicali elettronici.
In seguito, l'introduzione dei circuiti integrati nelle schede elettroniche ha aperto la porta a nuove ed incredibili possibilità negli strumenti musicali elettronici. Queste apparecchiature hanno dato la possibilità ai musicisti di creare sonorità che ben diverse dai semplici e forse un po' rozzi suoni dei primi organi elettrici; inoltre non solo sono in grado di riprodurre con buona precisione i tradizionali suoni, ma riescono a creare suoni mai sentiti prima, aprendo nuovi orizzonti musicali.

Lo standard MIDI

La sigla MIDI indica non solo le specifiche elettriche, che poi vedremo, ma anche il protocollo di comunicazione e il relativo significato dei dati in transito.
Ora vediamo più in dettaglio il MIDI.

Specifiche elettriche

Dal punto di vista elettrico, il segnale è digitale di tipo TTL, con valori di 0 – 5 Volt.Il trasferimento dei dati, organizzati in byte, è di tipo seriale asincrono con un bit di start, 8 bit dei dati e un bit di stop. Il primo bit ad essere trasferito è quello meno significativo, LSB.
La connessione avviene mediante un cavo schermato a 2 oppure 3 conduttori: i positivo a +5 Volt, il segnale ed eventualmente la massa. La massa è utilizzata anche per il collegamento con la schermatura del cavo. La schermatura è importante per l'immunità ai disturbi. Il riferimento del segnale non è quindi la massa, ma il positivo, anche se in genere nei diagrammi, il segnale dei dati viene riferito a massa, come un normale segnale seriale asincrono, simile al protocollo RS232 (a parte i livelli di tensione, però!).
La velocità di trasmissione è un po' insolita, se si fa il paragone con le velocità dell'interfaccia seriale per PC: 31250 bps (bit per second). I connettori che si trovano nelle apparecchiature musicali sono 3: IN, OUT e THRU. Rispettivamente sono: l'ingresso dei dati, l'uscita e uscita di ripetizione dell'ingresso. Il terzo connettore può sembrare inutile, ma in realtà non lo è. Esso è fondamentale per poter connettere più sintetizzatori e/o expander in cascata, in modo tale che il PC o il sequencer riesca a pilotare tutti i sintetizzatori.

Protocollo

Dopo quanto scritto, viene spontanea una domanda: che tipo di dati vengono trasmessi? La risposta è molto semplice: vengono trasmessi solamente comandi e gli eventuali parametri ad essi associati; non viene trasmesso alcun suono, nemmeno in forma digitalizzata. Così vengono trasmessi comandi come: nota-on e nota-off e indicazione su quale nota, cambia suono, modifica volume, modifica pitch-bender, ecc. Il numero di strumenti, o di canali, che lo standard MIDI riesce a gestire è pari a 16; anche se esistono delle estensioni per arrivare a 32. Così i comandi possono fare riferimento ad un singolo canale MIDI oppure a tutti. La tabella numero 1 elenca alcuni dei comandi più usati.



di Federico Battaglin

Collegamento al PC

Quando si utilizza un PC e quindi un programma, nasce la necessità di rendere il computer MIDI compatibile, ovvero dotato dell'interfaccia MIDI. Per fortuna le schede audio per PC hanno un connettore, in genere utilizzato per il joystick, che rende disponibili i segnali di MIDI IN e MIDI OUT.. L'anello mancante è proprio un circuito elettronico di interfaccia che colleghi fisicamente il PC con il mondo musicale MIDI. E questo è quello che viene presentato ora. A dire il vero, schemi elettrici di interfaccia MIDI ce ne sono tanti sia pubblicati su altre riviste sia apparsi su Internet. Alcuni schemi sono molto semplici, altri invece sono standard e molto simili tra loro.

Lo schema che ora verrà presentato è un po' particolare per vari motivi, che ora vedremo:

1. Funzione di base: interfaccia MIDI standard;
2. MIDI IN: optoisolato con dispositivo ad alta velocità;
3. MIDI OUT: 5;
4. MIDI THRU: 1;
5. Funzione avanzata: expander autonomo.

Codice Midi	Significato
0h 8n	Questo comando è relativo alla disattivazione della nota attivata in precedenza. È possibile disattivare una nota o sapere quale nota è stata disattivata. Nel codice midi, il parametro "n" indica uno dei 16 canali.
0h 9n	Questo comando è relativo all'attivazione di una nota. È possibile attivare una nota o sapere quale nota è stata attivata. Vi è inoltre un dato aggiuntivo che indica il valore della forza con cui si preme o si vuol premere il tasto, il cosiddetto "initial touch". Nel codice midi, il parametro "n" indica uno dei 16 canali.
0h Bn	Questo comando interviene con tutta una serie di controlli e parametri del proprio sintetizzatore. È necessario impostare o riconoscere un numero di riferimento (0-120) sia per in sintetizzatore sia per il dispositivo esterno. Segue poi il valore del parametro da modificare o da rilevare. In genere si ha: 7= Volume 14= Modulation Wheel 64= Sustain Switch 65= Portamento Switch Nel codice midi, il parametro "n" indica uno dei 16 canali.
0h Cn	Questo comando agisce sul cambio di voce, o, più volgarmente, strumento. È possibile interagire sia con le voci interne sia con quelle su "card" esterna e con le voci di "performance", se esistono. Nel codice midi, il parametro "n" indica uno dei 16 canali.
0h Dn	Questo comando permette di accedere al parametro del cosiddetto "after touch". La maggior parte dei sintetizzatori è sensibile alla variazione di pressione del dito una volta che questo ha premuto un tasto: serve per creare degli effetti particolari. Nel codice midi, il parametro "n" indica uno dei 16 canali.
0h En	Questo comando fa riferimento al controllo "Pitch bender", ovvero a quella rotellina, in genere a sinistra della tastiera che permette una variazione (sia verso l'alto che verso il basso) della tonalità delle note. Dato che l'estensione di variazione è piuttosto ampia e superiore al valore massimo di un solo dato, viene spezzata in due parti, la parte più segnificativa seguita da quella meno significativa. Nel codice midi, il parametro "n" indica uno dei 16 canali.
0h FE	Questo comando chiamato "Active Sense" serve per sincronizzare i dispositivi musicali.
0h Fx	Questo comando è molto importante, in quanto indica un messaggio si sistema. Nel codice midi, la "x", generalmente è uguale a "0". Con il comando "F0" si accede a tutta una serie di parametri, molti dei quali propri del sintetizzatore o dell'apparecchio musicale con cui si sta interagendo. Al comando "Fx", spesso, seguono molti parametri, cui bisogna leggere le specifiche dell'apparecchio in questione.

Tabella 1
Alcuni dei comandi MIDI

Come accennato poco sopra, esistono alcuni schemi elettrici estremamente semplici e altrettanto economici: essi sono costituiti ad un semplice collegamento elettrico tra la porta joystick e i connettori MIDI. Una simile soluzione presenta una serie di problemi, tale per cui vale spendere un pò di soldi in più e aver una maggiore sicurezza: pilotaggio delle uscite, non isolamento con il MIDI IN, possibili ritorni di segnale di massa.

Lo schema proposto invece evita i suddetti problemi per non avere problemi nellpèinterfacciamento dei dispositivi collegati ai cavi MIDI. Inoltre per una maggiore flessibilità, il numero di uscite è stato aumentato arrivando a 5.
È possibile estendere questo numero ad 8, 10, ecc. Ma la particolarità di questo circuito è un'altra: la duplice funzionalità di interfaccia MIDI e di EXPANDER. Vediamo da più vicino:

1. INTERFACCIA MIDI: questa funzionalità, come già si sa, permette la connessione tra il PC e il mondo musicale MIDI;
2. EXPANDER: quando si ha la necessità di connettere più strumenti musicali in cascata si fa uso della connessione MIDI THRU. Il problema è che non è consigliabile porre in cascata più di 3-5 dispositivi, pena un'introduzione di un certo ritardo, a seconda del tipo di strumento che funge da ponte MIDI IN e THRU.
La funzionalità di expander permette di riportare il segnale di MIDI IN direttamente alle 5 uscite di MIDI OUT oltre all'uscita MIDI THRU. In questo modo è possibile avere il segnale di MIDI IN direttamente duplicato su 6 uscite! E il tutto senza dover utilizzare il PC!

Schema elettrico

Finalmente arriviamo a vedere da più vicino lo schema elettrico riportato nella figura 3.
Innanzitutto vediamo subito quali sono le particolarità di questo schema. Un fotoaccoppiatore veloce con 10Mhz di banda passante, di tipo 6N137 assicura un perfetto trasferimento dei segnali digitali ben superiore rispetto ai fotoaccoppiatori standard, per esempio, il 4N35. Questo si traduce in fronti di salita e discesa ben

STRUMENTI MUSICALI ELETTRONICI

Come accennato nel paragrafo precedente, gli strumenti musicali elettronici sono in grado di generare un enorme numero di suoni e vengono anche chiamati sintetizzatori musicali. Come suggerisce la parola stessa, tali apparecchi riescono a sintetizzare, creare nuovi suoni, mediante un grande numero di regolazioni di vario tipo. In tal modo il musicista è in grado di modificare, in tempo reale, le sfumature del suono utilizzato.
Ogni tasto della tastiera musicale fa capo un oscillatore che può essere programmabile, la cui uscita viene miscelata con le uscite di tutti gli oscillatori. Per semplicità potevano esserci strumenti di tipo monofonico dotati di un solo oscillatore. Segue poi uno stadio di generazione di inviluppo: con regolazioni è possibile determinare l'andamento dell'ampiezza del suono; per esempio in un pianoforte il suono inizia subito e decade con un certo tempo; nel flauto invece il suono può terminare subito, appena cessa il flusso d'aria. Una sigla ricorrente è l'ADSR, il cui acronimo è: Attack, Delay, Soustain, Release, che indica la varie fasi dell'inviluppo. Usa sezione assai importante è lo stadio dei filtri, che intervengono direttamente sul suono, permettendo la modifica delle sfumature. Quanto detto riassume molto velocemente la tipica struttura di base di un sintetizzatore.

I primi sintetizzatori apparsi nel mondo musicale, e si sta parlando della fine degli Anni Cinquanta, sono di natura analogica in quanto l'unico elemento attivo è, prima, solamente il transistor e successivamente, anche il circuito integrato. Si comprendo così come i sintetizzatori analogici sono presenti ancora sul mercato negli Anni Settanta. Il più famoso tipo di sintetizzatore usato negli Anni Settante e primi Anni Ottante è il Moog, nome dell'americano, Robert Moog.
Nella figura 1 è possibile vedere un esempio di sintetizzatore musicale di tipo analogico. Si notino le innumerevoli manopole per la regola-



Figura 1
Esempio di sintetizzatore analogico

netti e non arrotondati, situazione, questa, quando si è ai limiti della banda passante. Viene impiegato anche un buffer ottale quale componente per il pilotaggio delle uscite. La sua funzione è quella di amplificatore di corrente per linee. Per il circuito in questione, sono consigliati i modelli HCT e, preferibilmente, HC. Sono da evitare gli altri modelli, in quanto i consumi statici e dinamici sono troppo elevati rispetto alle prestazioni richieste. Lo schema elettrico può essere suddiviso in varie sezioni, che andremo ad analizzare una per una.

Alimentazione

Questo stadio viene usato solamente per la funzione di EXPANDER, in quando il circuito non è alimentato dal PC. Il circuito accetta una tensione di 8-12 Volt stabilizzati con una corrente di 300mA. Il diodo D1 protegge da involontarie inversioni di polarità. C1 serve per livellare la tensione d'ingresso, mentre C2 la tensione in uscita.
IC1 invece è un normalissimo regolatore di 5 Volt per il corretto funzionamento dei circuiti integrati.

MIDI IN

Lo stadio di MIDI IN è isolato galvanicamente dal resto del circuito per evitare loop di segnale di massa se vengono collegati i cavi di MIDI IN e MIDI OUT. Il nucleo dello stadio di MIDI IN è costituito dal fotoaccoppiatore veloce OC1, di tipo 6N137, il quale ha bisogno di un'alimentazione esterna. Il La resistenza R2 a monte del fotoaccoppiatore serve per limitare la corrente attraverso il diodo led interno al fotoaccoppiatore. Il diodo D2 invece per proteggere il led interno da inversione di polarità. A valle del fotoaccoppiatore, il segnale esce dal pin 6 e viene applicato alla prima porta del buffer e collegato anche alla resistenza R1 di pull-up, dato il tipo di uscita del fotoaccoppiatore. Il valore di R1, 560 Ohm, può sembrare abbastanza ridotto, ma è necessario per garantire una migliore pulizia del segnale, soprattutto per quanto riguarda i fronti di salita e discesa. Impiegando un valore superiore, per esempio 10Kohm, tali fronti si presenterebbero più arrotondati, con un peggioramento della qualità del segnale; questo verrebbe accentuato ancor più all'aumentare della lunghezza

zione di molti aspetti della generazione del suono.
Con l'avvento dell'elettronica digitale, anche il sintetizzatore subisce delle notevoli modifiche.
Intanto spariscono tutte le manopole ed interruttori cui si era abituati e le regolazioni sono di tipo parametriche, di tipo digitali. Fanno apparsa un bel display e un tastierino con numerosi pulsanti per la programmazione del sintetizzatore, come mostrato nella figura 2.



Figura 2
Esempio di sintetizzatore digitale

Ecco, non si parla più di regolazione, ma di programmazione. I parametri vengono visualizzati, modificati e salvati. Il fatto di salvare i parametri consente di memorizzare i suoni creati e di richiamarli in seguito, mediante la pressione di uno o pochi tasti del tastierino.
Quanto scritto è una carrellata molto sintetica sui sintetizzatori, non essendo lo scopo dell'articolo una descrizione dei sintetizzatori.
1983: la svolta
Negli Anni Ottanta i sintetizzatori sono di tipo digitale con la programmazione parametrica. Comincia a diffondersi anche l'uso del Personal Computer (PC) e si sente l'esigenza di unire i due mondi: ma come?
Il 1983 è la data ufficiale della nascita dello standard MIDI, acronimo di Music Instrument Digital Interface. Al di là della sterile sigla, questo significa che ora i sintetizzatori e altri strumenti musicali possono comunicare tra loro e con il PC, grazie ad un standard. Così, se uno strumento è MIDI compatibile, significa che è interfacciabile con altri strumenti elettronici e il PC. Ecco che nasce un nuovo apparecchio che affianca il sintetizzatore, quando non c'è il PC: il sequencer. Questo apparecchio non è altro che un registratore digitale di eventi MIDI, fondamentale per poter memorizzare un brano, modificarlo ed arricchirlo di nuove sonorità. Inoltre dal sintetizzatore nasce una particolare variante senza la tastiera musicale: l'expander. Come suggerisce il nome, tale apparecchio aumenta il numero di suoni che posso venir suonati e/o aumenta il numero di polifonia, ovvero il numero di note che possono venir suonate in contemporanea.

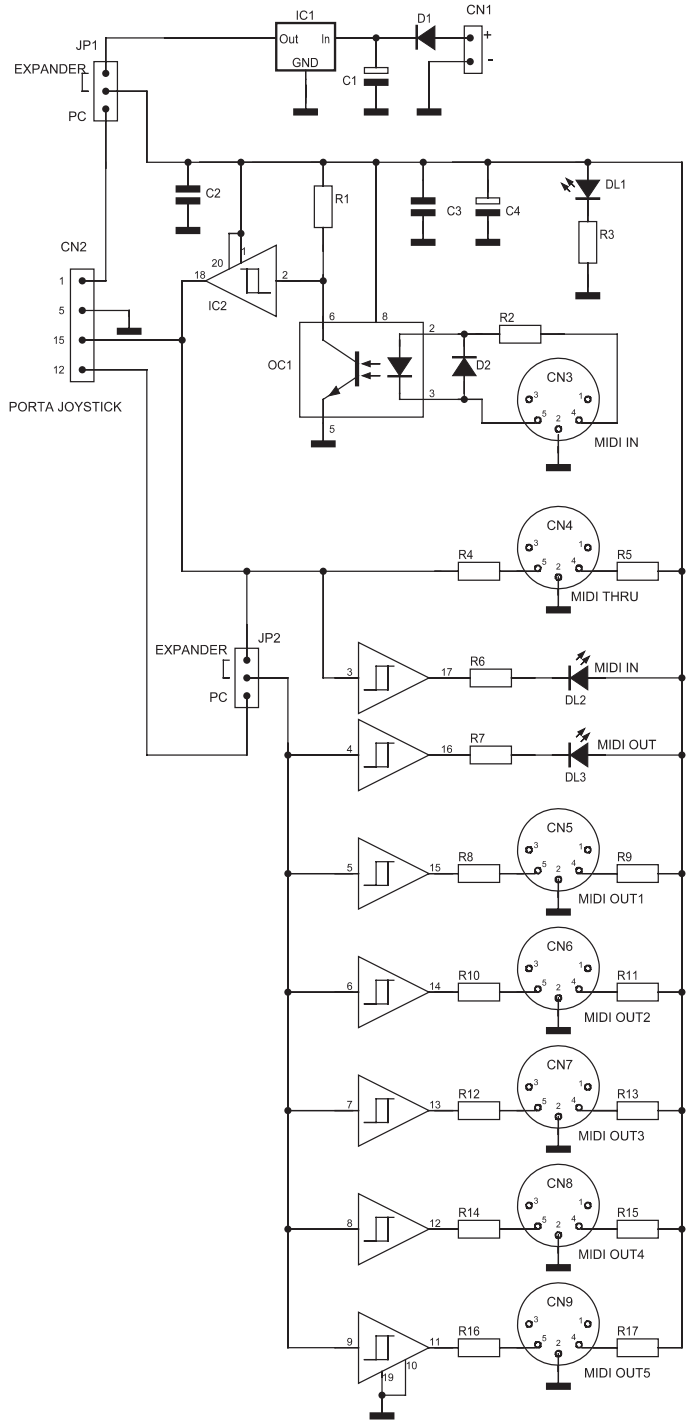


Figura 3
Schema elettrico

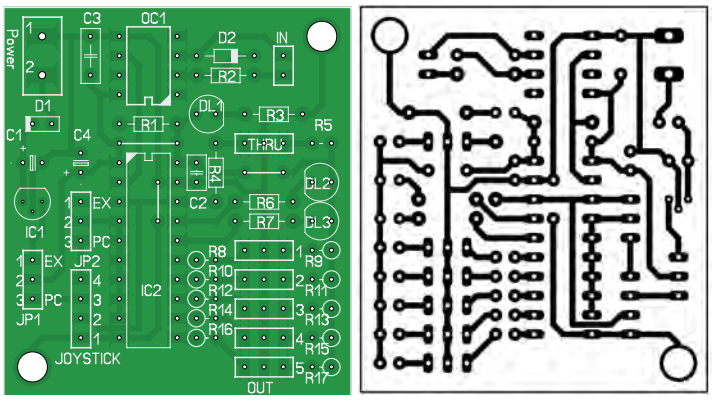


Figura 4
Master in scala 1:1 e disposizione dei componenti

dei cavi. Infine C3, posizionato più vicino possibile al fotoaccoppiatore funge da filtro sulla rete di alimentazione, utile per ridurre le emissione e diminuire la suscettibilità ai disturbi esterni.

MIDI THRU

Lo stadio di MIDI THRU è molto semplice in quanto il segnale all'uscita del primo buffer viene riportato con connettore THRU.

MIDI OUT

Il segnale proveniente dalla porta joystick (pin 12 di CN2) oppure dal MIDI IN (pin 18 di IC2) viene riportato al MIDI OUT mediante un buffer contenuto in IC2. Le uscite MIDI OUT 1 fino a MIDI OUT 5 sono tutte identiche tra loro. Tutte le porte di uscita, MIDI THRU e MIDI OUT sono contornate da due resistenze di ugual valore, precisamente da R2 da 220 Ohm: la loro funzione è di limitare la corrente uscente dai buffer e di alimentazione della porta, soprattutto nel caso di corto circuiti.

PORTA JOYSTICK

Del connettore CN2 a vaschetta a 15 poli della porta joystick vengono utilizzati solamente 4 segnali: massa, positivo, in e out. Il segnale uscente dal pin 18 di IC2, proveniente dal MIDI IN viene collegato all'ingresso al pin 15. Mentre l'uscita della porta joystick va alle uscite MIDI OUT.

Montaggio

Il montaggio del circuito elettronico è abbastanza semplice. È possibile utilizzare un ritaglio di basetta di tipo "millefiori". Se invece si preferisce un qualcosa di più comodo, ho disegnato disegnato un circuito stampato pronto all'uso. Nella figura 4 è possibile vedere la disposizione dei componenti e delle piste di collegamento. Per semplicità e per chiarezza di visualizzazione non è visibile il piano di massa in realtà presente. La cornice in azzurro attorno al bordo bianco della scheda indica, almeno per il software di CAD utilizzato, la presenza del piano di massa. Il prototipo montato e privo dei LED e dei cavetti di collegamento è visibile in figura 5.
Qualunque supporto si scelga per il montaggio dei componenti elettronici, è bene utilizzare un saldatore a bassa potenza, circa 20-30W, dotato di punta sottile e

SELETTORE FUNZIONI

Mediante i jumper JP1 e JP2 è possibile scegliere la modalità di funzionamento. Il JP1 serve la selezione della fonte di alimentazione: dalla porta joystick del PC (funzionalità interfaccia MIDI) oppure esterna (funzionalità expander). Il JP2 serve la selezione della fonte dei dati per il pilotaggio delle porte MIDI OUT: dalla porta joystick del PC (funzionalità interfaccia MIDI) oppure dalla porta MIDI IN (funzionalità expander). Eventualmente è possibile sostituire i due jumper con un deviatore a 2 posizioni e a 2 vie (DPDT, double pole, double throw) da collocare in una posizione più comoda, per esempio sul pannello di un contenitore.

impiegare uno stagno di buona qualità, in filo a ridotto diametro, per esempio 1mm. I componenti dovrebbero venire saldati con un certo criterio, ovvero partendo da quelli a profilo più

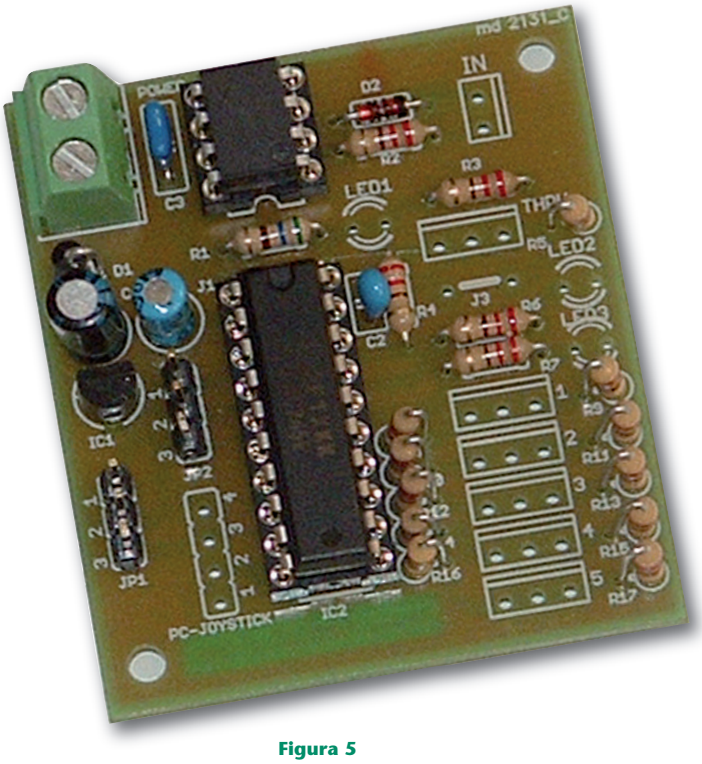


Figura 5
Il circuito elettronico montato

circuiti stampati in 24 ore

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

Potrete scegliere tra singola e doppia faccia con foro metallizzato. Con solder e serigrafie per uno stampato di alta qualità o solo piste stagnate per un prototipo a basso costo.
Prezzi a partire da* € 14,38 (doppia faccia foro metallizzato 7,50x7,50 cm) e da € 9,13 (singola faccia 7,50x7,50 cm) per FR4 1,6 mm con rame 35 µm, **tutti comprensivi di attrezzatura.**
Nessuna limitazione sul numero dei fori, sul numero degli utensili (diametri) e sul tipo di scontornatura (anche tondeggiante).
Distanza minima tra le piste e pista minima 8 mils (0,20 mm).

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

+ QUALITÀ - TEMPO

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

Codice MIP 268023

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

**È consigliabile utilizzare collegamenti molto corti tra il circuito stampato e i connettori MIDI e il connettore della porta Joystick.
Il motivo è da ricercare nell'evitare che i fili non coperti da schermatura captino disturbi provocando possibili malfunzionamenti, anche saltuari.
Nelle foto delle figure 6 e 7 è possibile vedere la piedinatura dei connettori utilizzati.**

basso, come i due ponticelli presenti, a quelli a profilo più alto, tipo i condensatori elettrolitici. È necessario prestare un'opportuna attenzione ai componenti polarizzati: condensatori elettrolitici, regolatore di tensione, diodi, led, circuiti integrati.

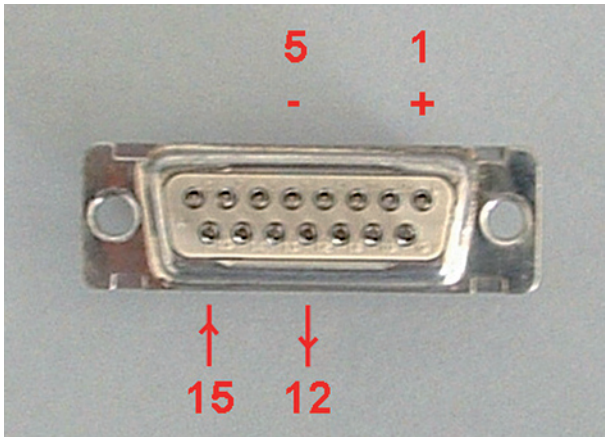


Figura 6
Piedinatura del connettore Joystick

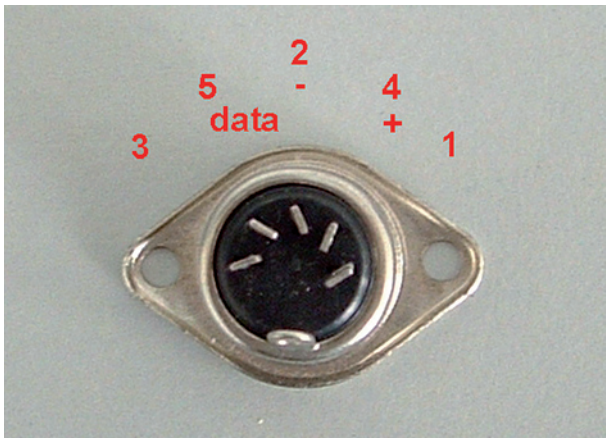


Figura 7
Piedinatura del connettore MIDI

Per questi ultimi è consigliabile utilizzare uno zoccolo, meglio se a tulipano, in modo da non danneggiare i delicati componenti durante la fase di saldatura e di poterli eventualmente sostituire agevolmente. Per quanto riguarda i jumper, si può optare di portarli fuori dal circuito stampato e utilizzare un doppio deviatore, magari a levetta, da pannello, da fissare sul contenitore.
A montaggio terminato, conviene controllare attentamente se non si siano dimenticate saldature, che non ci siano corto circuiti tra le piste e le saldature. Se tutto sembra essere a posto, ancor prima di inserire i circuiti integrati negli zoccoli, conviene alimentare i circuito selezionamento la modalità di expander e verificare le corrette tensioni nei vari nodi.
Se anche questo test è stato superato in maniera positiva, allora, tolta l'alimentazione, si può procedere all'inserimento dei circuiti integrati, prestando attenzione al verso giusto.

Collaudo

Finalmente è arrivato il momento tanto atteso: il collaudo e il normale utilizzo. Ad apparecchiature spente, collegate il circuito al PC e ai vostri strumenti musicali e infine accendete il tutto.
È necessario predisporre gli strumenti musicali al trasferimento di dati MIDI, in quanto la funzionalità MIDI potrebbe essere normalmente disattivata. Sul lato PC, invece, fate partire un qualunque programma per la gestione di strumenti musicali MIDI, per esempio un sequencer.
È inutile che suggerisce dei nomi di software, perché ognuno ha le proprie preferenze e in Internet ce ne sono davvero tanti, anche di tipo freeware. Ora, se tutti i collegamenti sono corretti e i settagli opportuni, allora siete davvero pronti per iniziare!

More Info Please!
Inserisci il Codice **268018** alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

PESCARA 2007

ASSOCIAZIONE RADIOAMATORI ITALIANI

Sezione di PESCARA
Via delle Fornaci, 2
Tel 085 4714835 Fax 085 4711930
http://www.aripescara.org
e-mail: aripescara@aripescara.org

PROTEZIONE CIVILE

DXCC DESK

42ª FIERA MERCATO NAZIONALE DEL RADIOAMATORE DI PESCARA

24 - 25 NOVEMBRE 2007

SILVI MARINA (TE) - FIERA ADRIATICA - S.S.16 (Nazionale Adriatica) - Km. 432
SABATO 9:15 - 19:00 DOMENICA 9:00 - 19:00
AMPIO PARCHEGGIO GRATUITO - RISTORANTE - SELF SERVICE INTERNO



con il patrocinio di REGIONE ABRUZZO, Provincia di Pescara, Comune di Silvi

Codice MIP 268025

Sistemi elettronici a bordo auto

Uno dei principali trend di sviluppo tecnologico nel campo dei veicoli civili e industriali è rappresentato dall'utilizzo di sistemi totalmente elettrici per la trasmissione e l'attuazione dei comandi di guida, in sostituzione dei tradizionali apparati meccanici e/o idraulici. Questi sistemi sono chiamati x-by-wire e la loro introduzione nei veicoli permetterà di ottenere livelli di sicurezza di guida, efficienza di impiego e flessibilità di manovra mai raggiunti prima e renderà operanti funzionalità innovative di gestione e di manutenzione dei veicoli.

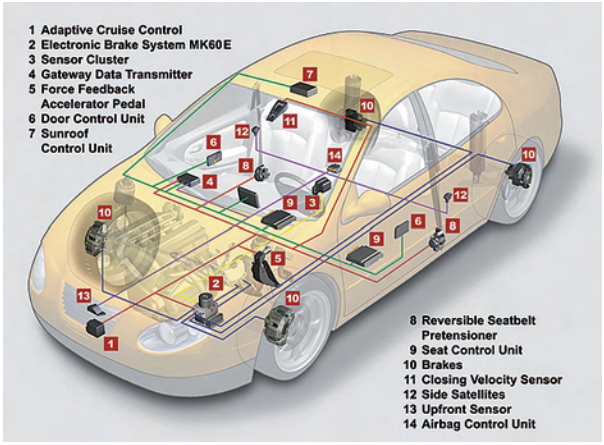


Figura 1
I sistemi x-by-wire consento di aumentare l'affidabilità dell'autovettura e dei passeggeri poiché garantiscono il funzionamento anche in condizioni di guasto. Sistemi di questo tipo sono definiti fault-tolerant

Sistemi di controllo meccanici ed elettronici

Un sistema x-by-wire è un apparato composito in cui dispositivi elettrici eterogenei (trasduttori, attuatori elettromeccanici, unità elettroniche, reti di comunicazione) cooperano al fine di eseguire i comandi di guida impartiti dal conducente. La caratteristica fondamentale richiesta ai sistemi x-by-wire è la sicurezza funzionale che deve essere almeno pari a quella degli apparati tradizionali dei quali prendono il posto e che deve essere conseguita ricorrendo a soluzioni adeguate alle esigenze di una produzione di massa come è quella autoveicolare. Tali sistemi liberano il conducente dalla gestione di compiti di routine e lo assistono durante situazioni critiche, fornendo in qualsiasi momento un sistema fault tolerant.
La tecnologia x-by-wire è stata introdotta inizialmente nell'industria aerospaziale, con lo scopo appunto di sostituire i sistemi idro-meccanici, ritenuti meno affidabili di quelli elettronici. X-by-wire è stata dunque presa in prestito dal settore aerospaziale ed è oggi utilizzata in autovetture quali Mercedes-Benz e BMW.

In generale le autovetture sono costituite da vari componenti che interagiscono tra loro in modo ben definito. Questi sistemi si possono suddividere in tre categorie fondamentali:

- Elettronica di consumo, che comprende dispositivi come cellulare, sistema di navigazione ed accesso ad Internet. Essi hanno bisogno di elevata banda, ma non sono critici ai fini della sicurezza.
- Body electronics, in cui rientrano sistemi come aria condizionata, antifurto, display, modulo portiere e luci. Essi richiedono una comunicazione soft real-time, che esegua operazionei senza errori per la maggior parte del tempo; tempi di inattività occasionali e di breve durata sono accettabili.
- Elettronica di sistema, che comprende ABS, controllo motore e cambio. Essi necessitano di una comunicazione hard real-time poiché essenziale per garantire la sicurezza del veicolo. Con il termine hard real-time si intende che ciascun task deve essere concluso entro un intervallo di tempo predefinito. Tali sistemi devono essere anche fault-tolerant.

l'approccio x-by-wire



di Savino Giusto

L'articolo focalizzerà l'attenzione sui sistemi della terza categoria.

Requisiti nel settore automotive

Il settore automotive ha molti vincoli che devono essere soddisfatti, affinché i relativi prodotti possano essere considerati veramente affidabili e sicuri.
È necessario che siano operativi anche in ambienti critici, con diversi livelli di temperatura e rumore. Un altro fattore diventato essenziale nella moderna società è il fatto di essere eco-compatibile. Le autovetture sono prodotte su larghissima scala e devono quindi soddisfare tutte le richieste della produzione di massa come basso costo, modularità dei sistemi e fattibilità. Un'autovettura consiste di componenti sviluppati da differenti società che sono integrati in diverse configurazioni. La diagnosi on-board è una delle caratteristiche chiave. In questo modo viene fornita una diagnosi in tempo reale; inoltre, per guasti intermittenti il sistema deve essere anche in grado di memorizzare il codice di errore ed informare in tal modo il conducente che è necessario un determinato tipo di manutenzione.
Queste sono le richieste che in generale devono essere offerte da ogni sistema a bordo auto. Per i sistemi x-by-wire, in particolare, si richiedono comportamenti deterministici e supporto fault-tolerant, che non comprometta in alcun modo la sicurezza dell'autovettura e dei passeggeri. Il sistema continuerà ad eseguire le funzionalità per cui è stato progettato anche in presenza di un guasto. Per esempio, se c'è un problema al sistema sterzante, la funzionalità dello sterzo deve essere mantenuta finché non sarà raggiunta la più vicina area di servizio. L'autovettura deve cioè continuare a funzionare in uno stato cosiddetto fail-safe.

Sistemi elettronici x-by-wire

Esempio pratico: brake-by-system

I sistemi x-by-wire possono essere applicati a diversi sistemi automotive. Un esempio pratico è rappresentato da EMB, acronimo di Electro-Mechanical Brakes. Si tratta in sostanza del sistema brake-by-wire che sostituisce le parti idrauliche (fluido) e meccaniche dell'impianto frenante con sensori e collegamenti elettrici. Una possibile architettura del sistema EMB è mostrata in Figura 2.
La prima casa costruttrice di automobili ad annunciare una tecnologia di questo tipo fu Mercedes, che nell'agosto 2001, tramite un comunicato stampa, annunciava il suo sforzo nella realizzazione

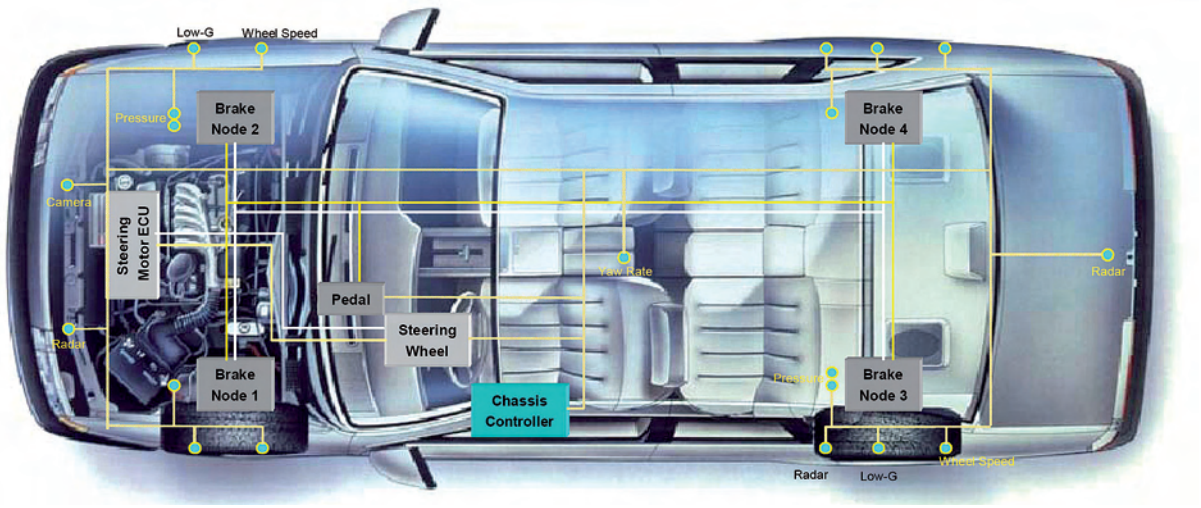


Figura 2
Architettura tipica di un sistema brake-by-wire. Il sistema ha il compito di sostituire le parti meccaniche ed idrauliche dell'impianto frenante con sensori elettronici che ne controllano il funzionamento

di un sistema frenante di tipo elettronico. Si trattò del primo sistema brake-by-wire e fu introdotto sul modello Mercedes-Benz SL500.
Per ciascun freno c'è un'unità di controllo elettronico (anche detta ECU) che esegue tutti i task relativi a quel freno, tra cui anche l'elaborazione delle informazioni provenienti da sensori di velocità, di sterzo e del motore. Ci sono, inoltre, alcuni attuatori, l'unità di potenza e di memoria. La comunicazione delle informazioni di ciascuna ECU avviene attraverso un canale che utilizza un'architettura time-triggered. Tale canale confluisce nel BBW (Brake-By-Wire) manager per la raccolta ed elaborazione di tutti i dati. Lo schema a blocchi di quanto descritto è riportato in Figura 3.
Il controller è connesso al canale di comunicazione time-trigger tramite il quale interagisce con le ECU, l'unità di potenza, gli attuatori ed i sensori. Come si nota dallo schema a blocchi, il pedale del freno è soltanto un emulatore del classico freno meccanico. La sua pressione non comporta la compressione del fluido frenante, bensì invierà l'informazione (sotto forma di segnale elettrico) all'unità di controllo, la quale a sua volta comanderà l'attuatore. L'informazione analizzata dal controller presente nel caliper (il dispositivo in cui sono montati i pad che agisco sul disco per frenare, come illustrato in Figura 4) include anche i dati provenienti da ABS (Anti-lock Braking System), VSC (Vehicle Stability Control) ed altri. Tutti questi valori servono al controller per decidere la forza da imprimere per la frenata mediante il modulo di power management, al fine di evitare lo slittamento delle ruote.
In caso di guasto, il sistema entra in uno stato fail-safe, definito come freno aperto con nessun potere frenante.
Un dispositivo come quello appena descritto presenta un tempo di reazione molto più veloce di uno idro-meccanico. Questo si concretizza in uno spazio di frenata minore, aumento di affidabilità e sicurezza dell'autovettura e dei passeggeri.

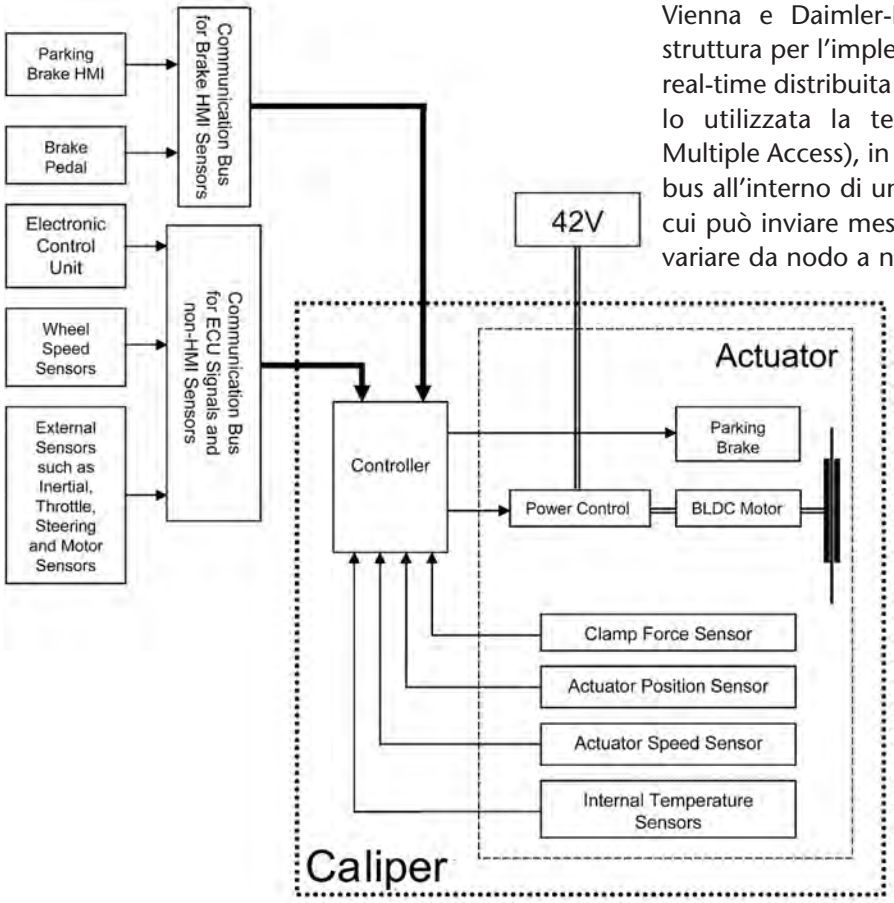


Figura 3
Schema a blocchi di un tipico sistema brake-by-wire

Protocolli di comunicazione

Gli attuali sistemi per automobili impiegano complesse reti per connettere tra loro sotto-sistemi. A partire dagli anni '80, c'è stato un largo uso del protocollo di comunicazione CAN, sigla che significa Controller Area Network. Esso utilizza un'architettura di tipo event-triggered, che però non risulta ottimale per la realizzazione di sistemi x-by-wire. A seguito di ricerche in questo settore sono state introdotte tecniche di comunicazione di tipo time-triggered come TTP/C e FlexRay.

Time-Triggered Protocol (TTP/C)

TTP/C è nato dalla collaborazione tra l'università di Vienna e Daimler-Benz Research; definisce una struttura per l'implementazione di un'applicazione real-time distribuita ed affidabile. Questo protocollo utilizzata la tecnica TDMA (Time Division Multiple Access), in cui ciascun nodo si connette al bus all'interno di un predefinito slot temporale, in cui può inviare messaggi. La durata dello slot può variare da nodo a nodo.
L'unità base è il nodo, che comprende l'interfaccia di rete, il controller per la comunicazione time-triggered e l'host.
Un generico nodo è costituito da due blocchi fondamentali: il sotto-sistema host a cui spetta il compito di eseguire le applicazione real-time, il sotto-sistema di comunicazione (controller) che ha, invece, il compito di trasmettere messaggi affidabili in real-time.
Quest'ultimo è basato su due canali che trasmettono le informazioni indipendentemente. L'interfaccia tra questi due sotto-sistemi è chia-

Il nostro primo lettore...

www.costruirehifi.net

COSTRUIRE HIFI

Da sempre in edicola!!!

Codice MIP 2680229

Blu Press

www.blupress.it



Figura 4
Attuatore che agisce sul disco per effettuare la frenata dell'autovettura. In un sistema EMB il comando viene impartito dal controller sulla base delle informazioni provenienti da tutti i sensori

mata CNI (Communication Network Interface); si tratta, come è schematizzato nella Figura 5 di una memoria RAM dual-port, dotata di una linea di interrupt tra l'host ed il controller. Quando l'host vuole inviare dati sulla rete, esso effettua un'operazione di push nella memoria CNI; il controller trasporta tali dati sul bus al momento opportuno. Il nodo ricevitore, a questo punto, effettua una pull dalla memoria CNI in attesa che l'informazione sia elaborata dal relativo host.
L'architettura TTP/C supporta due differenti topologie di rete: a bus ed a stella.

Nel primo caso (riportato in Figura 6) ciascun nodo è dotato di un bus guardian che gestisce l'accesso del controller al bus. La topologia a stella (riportata in Figura 7) implementa due centri stella che agiscono da bus guardian.

FlexRay

FlexRay è un nuovo protocollo di comunicazione di rete appositamente sviluppato per applicazioni automotive, con particolare riguardo a sistemi x-by-wire. Le sue linee guida sono state definite dal FlexRay Consortium [2] che comprende membri come Volkswagen, BMW, DaimlerChrysler, General Motors, Robert Bosch GmbH, NXP Semiconductors e Freescale. Tra le caratteristiche principali di FlexRay bisogna annoverare:

- Elevata velocità di trasmissione dati (10Mbps).
- Architettura time-triggered.
- Ridondanza.
- Sicurezza.
- Tolleranza ai guasti.

Un sistema FlexRay è costituito da un bus e vari nodi processore (ECU) connessi secondo topologie differenti. Ogni ECU ha un clock indipendente e da ciò nasce la necessità di sincronizzarli tutti. Secondo lo standard, il massimo drift (o variazione) permesso è pari a 0.15% rispetto al valore di riferimento. Questo significa che la differenza tra il clock più lento e quello più veloce potrà essere al massimo di 0.3%. Durante il normale funzionamento, viene periodicamente effettuata una sincronizzazione per minimizzare questo drift.

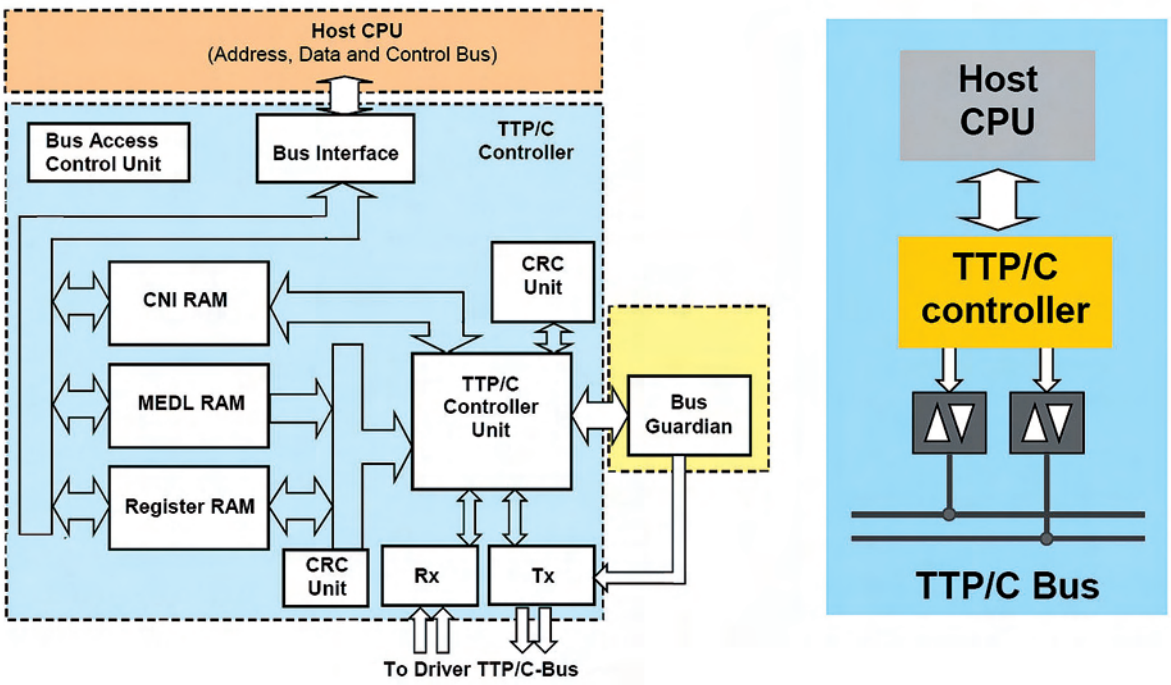


Figura 5
Schema a blocchi di un generico nodo di una rete TTP/C. L'accesso al mezzo avviene attraverso la tecnica della TDMA, in cui ogni nodo dispone di uno slot temporale per trasmettere i messaggi

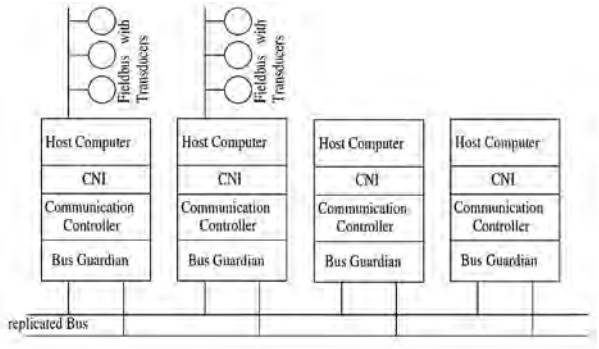


Figura 6
Topologia di rete a bus del protocollo TTP/C

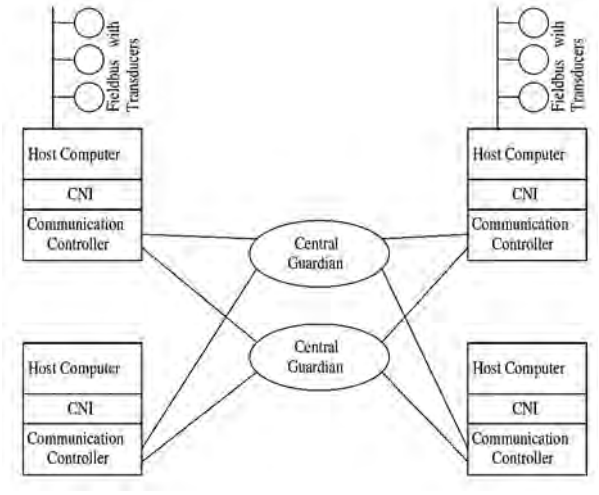


Figura 7
Topologia di rete a stella del protocollo TTP/C

FlexRay supporta varie topologie di rete, al fine di rendere più affidabile la comunicazione. I nodi possono essere configurati come un bus a singolo o a doppio canale oppure in configurazione a stella. Nella rete non devono, comunque, esistere anelli di richiusura. Infine, sono possibili anche configurazioni ibride sia a bus che a stella.
Nel protocollo FlexRay, ogni ciclo consiste di un segmento statico, di uno dinamico, di un segmento simbolo e di un tempo di inattività.
Il segmento statico implementa una multiplazione a divisione di tempo (TDMA) per la gestione dell'arbitraggio del bus. Il segmento dinamico implementa, invece, un arbitraggio basato su degli slot temporali variabili. La simbol window è un periodo di comunicazione in cui un simbolo può essere trasmesso sulla rete per permettere la realizzazione di un percorso ridondante. Infine, il tempo di inattività è un periodo in cui non ci sono comunicazioni che conclude il ciclo. Le porzioni statiche e dinamiche non interferiscono tra loro ed il sistema può operare sia in una che nell'altra modalità. In particolare, le applicazioni time-critical utilizzano lo schema di schedulazione statico con un timeslot predefinito; in questo modo si implementa un'architettura di tipo time-triggered. Altre applicazioni, invece, meno critiche possono utilizzare la modalità dinamica per implementare un'architettura di tipo event-triggered.

Meccanismi di guasto

Si è detto che i sistemi x-by-wire sono di tipo fault-tolerant, intendendo con ciò che in caso di guasto devo essere in grado garantire comunque il funzionamento, evitando il blocco dell'intero sistema.
Ma quali sono i meccanismi che possono provocare un guasto? Di seguito sono definiti e descritti quelli principali che possono verificarsi:

- Babbling idiot;
- Masquerading;
- Slightly-off-specification (SOS);
- Crash/Omission (CO);
- Disturbi elettromagnetici.

Babbling idiot failure

Questo guasto si verifica quando una ECU invia messaggi in maniera non sincronizzata, alterando la trasmissione sul bus. Il protocollo TTP/C evita questo problema con l'uso delle unità guardian. Se un messaggio non temporizzato è rilevato, esso viene immediatamente invalidato. Il protocollo FlexRay, invece, teoricamente non elimina questo problema durante la trasmissione dinamica, mentre lo evita durante quella statica. Quindi si può concludere che i sistemi x-by-wire non soffrono di questo problema.

Masquerading failure

Si verifica se un nodo assume l'identità di un altro nodo. Il sistema può essere danneggiato se il nodo che presenta l'errore invia messaggi; il ricevitore in tal caso non sarà in grado di rilevare il problema. Con il protocollo TTP/C questo non si può verificare poiché la temporizzazione e le informazioni dei messaggi sono conosciute a priori. Stesso discorso vale per FlexRay, nel caso di trasmissione statica.

SOS Failure

Può verificarsi nel processo di conversione analogico-digitale. Poiché i sotto-sistemi sono realizzati da differenti produttori, i valori di temporizzazione, frequenza e tensione posso essere leggermente differenti rispetto alle specifiche. Il controllo di tali valori ed il conseguente rilevamento dell'errore SOS viene effettuato tramite i bus guardian nel protocollo TTP/C; essi, infatti, confrontano il messaggio da controllare con uno di default. Un ulteriore controllo è effettuato tramite il CRC (Cyclic Redundancy Check). Se l'errore è rilevato, allora il messaggio deve essere invalidato. Stesso compito devono svolgere i bus guardian nel protocollo FlexRay.

CO failure

Il guasto Crash/Omission si verifica quando un nodo subisce un crash oppure il canale di comunicazione smette di operare. Con il protocollo TTP/C il rilevamento del guasto è demandato al membership management service. Ogni nodo conosce lo stato degli altri. In questa maniera, se un nodo subisce un crash, gli altri nodi sono in grado di rilevarlo. Il protocollo FlexRay, invece, non fornisce nessun meccanismo per rilevare il guasto, perciò è compito dell'applicazione gestirlo.

Disturbi elettromagnetici

Disturbi di natura elettromagnetica possono causare blocchi temporanei nella comunicazione. Questi problemi devono essere risolti in fase di progettazione, con opportune tecniche di schermatura dei cavi o con la sostituzione dei cavi in rame con quelli in fibra (i quali sono meno suscettibili ai disturbi e.m.). Questi guasti sono molto rari, ma quando si verificano possono provocare seri malfunzionamenti dell'intero sistema. Un modo per risolverli consiste nel congelare il nodo nello stato precedente al guasto, se la mancanza di comunicazione è inferiore a 50ms. Mentre il protocollo TTP/C garantisce questo tramite il membership management service, il FlexRay non fornisce nessun supporto. Anche in questo caso il rilevamento è demandato all'applicazione.

Uno sguardo alla progettazione x-by-wire

Il controller FlexRay di Fujitsu con tecnologia X-by-Wire

In Figura 8 è mostrato un semplice ed economico nodo FlexRay, realizzato utilizzando il microcontrollore Fujitsu MB91F467D come host ed il dispositivo Fujitsu MB88121B come controller del bus. Inoltre, il trasceiver TJA1080 è un ottimo candidato per essere utilizzato come driver del bus.

Un soluzione alternativa a quella appena presentata è quella di utilizzare un unico dispositivo con host e controller integrati. Un esempio è il modello Fujitsu MB91F465XA.

Per semplificare lo sviluppo del software, i progettisti possono far riferimento ad un ampia scelta di tool di differenti vendor. Tra questi si ricordano DECOMSYS, TZM e Vector CANtech.

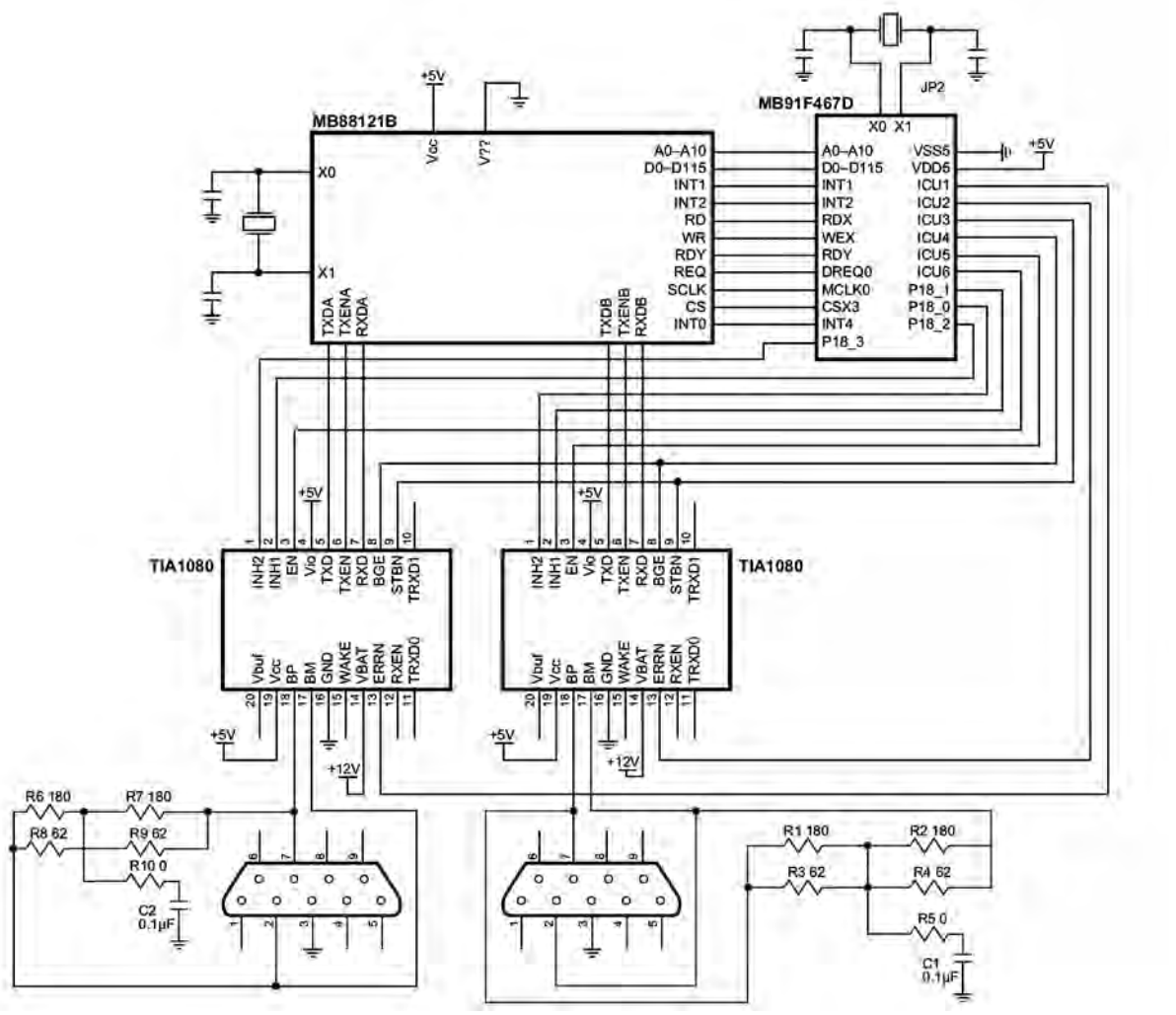


Figura 8
Schematico di un semplice ed economico nodo FlexRay utilizzando il micro MCU MB91F467D e il controller FlexRay MB88121B

Freescale

La risposta di Freescale Semiconductor alle applicazione x-by-wire è rappresentata dalla serie di prodotti disponibili al link [4]. In particolare, bisogna evidenziare il micro a 32-bit MPC5561, ideale per sistemi di assistenza alla guida che incorpora sensori di imagine e tecnologia radar. Oltre a questo da sottolineare la presenza di un controller FlexRay, specificatamente indicato per sistemi avanzati in condizioni di sicurezza. In alternativa, Freescale offre anche la possibilità di utilizza esclusivamente il controller FlexRay MFR4310, la cui documentazione è prelevabile dal link [5].

Conclusioni

Nell'articolo si sono analizzati sistemi elettronici di tipo fault-tolerant a bordo auto. Questi risultano molto più affidabili e sicuri di quelli idro-meccanici tradizionali e possono essere integrati con quelli pre-esistenti.

Tra questi moderni sistemi elettronici, grande successo hanno riscosso gli x-by-wire. Essi necessitano di un meccanismo di comunicazione in tempo-reale. Quindi i protocolli di comunicazione candidati a questo scopo sono TTP/C e FlexRay, i quali sono di tipo time-triggered.

Negli ultimi anni varie case automobilistiche hanno impiegato questa soluzione. Per esempio, i modelli SL e Classe E della Mercedes Benz, Toyota Estima, alcuni modelli di Nissan e BMW.

Riferimenti utili

[1] cordis.europa.eu/brite-euram/home.html
[2] www.flexray.com/
[3] www.fujitsu.com/us/services/edevices/microelectronics/microcontrollers/datalib/flexray/
[4] www.freescale.com/webapp/sps/site/homepage.jsp?code=FLEXRAY&fsrch=1
[5] www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MFR4310
[6] www.vector-informatik.com/vi_davinci_networkdesigner_flexray_en,,223.html

More Info Please!

Inserisci il Codice **268026** alla pagina **www.farelettronica.com/mip**
oppure utilizza il **modulo** a pagina **3**



Codice MIP 268033

Lampadina allo stato solido

Ecco come realizzare una lampadina allo stato solido, ossia formata da diversi LED ad alta luminosità. Non richiede alimentatore in quanto funziona direttamente a 220V.

La presenza in commercio di led ad alta luminosità non è più una novità, sono reperibili di potenze e colori diversi. Partendo da questa idea sarà possibile realizzare la lampada con le caratteristiche desiderate la cui particolarità è di essere alimentata a 220V direttamente dalla rete, senza alcun trasformatore. I led che ho utilizzato sono, ovviamente, di recupero. Si tratta di modelli utilizzati su alcune vetture come segnalatori di direzione, spesso inseriti all'interno dei due specchietti retrovisore esterni. Come conseguenza sono di colore arancio, ma la realizzazione è valida per qualsiasi led, di qualsiasi tipo e colore. Ovviamente la maggiore resa si ottiene utilizzando led ad alta luminosità di colore bianco.

Considerazioni costruttive

La tensione di soglia dei led è variabile con il colore, e non solo. Per i led non ad alta luminosità vale 1.5V per quelli rossi e 2.1V per i modelli verdi e gialli. I led più recenti ad alta luminosità hanno tensioni più alte, da 2 a 4 volt circa. In assenza di datasheet a cui far riferimento il sistema più rapido per determinare la tensione di soglia di un led è misurarla… si alimentano a 12 -15V alcuni led (tre o quattro in serie tra loro) con in serie un trimmer (330 – 560Ω), quale sostituto della classica resistenza di limitazione. Regoliamo il trimmer fino ad ottenere la massima luminosità, quando andando oltre non otteniamo un aumento significativo della luce emessa vuol dire che stiamo superando il ginocchio della caratteristica tensione-corrente dei diodi. Con il trimmer in questa posizione misuriamo prima la tensione ai capi di ogni singolo led, quindi la corrente che circola nel circuitino. Nel mio caso la tensione era pari a 2,35 V, come valore medio perché led identici



danno risultati che possono essere lievemente diversi (ho misurato 2,45 V, ma anche 2.28), la corrente era pari a 35 mA.
È evidente che alimentare un solo led direttamente a 220V è dispendioso; la resistenza dovrebbe dissipare circa 8W, per una serie di molti led le cose si mettono meglio: se i led fossero una cinquantina dovremmo dissipare nella resistenza poco meno di 4W… non va ancora bene.
La soluzione è nell'evitare l'utilizzo della resistenza di limitazione e utilizzare un condensatore la cui reattanza provochi la caduta di tensione necessaria e limiti la corrente così come farebbe la resistenza, senza tuttavia dissipare potenza. È vero che in questo modo si provoca uno sfasamento della tensione di rete, ma è anche vero che in genere i carichi

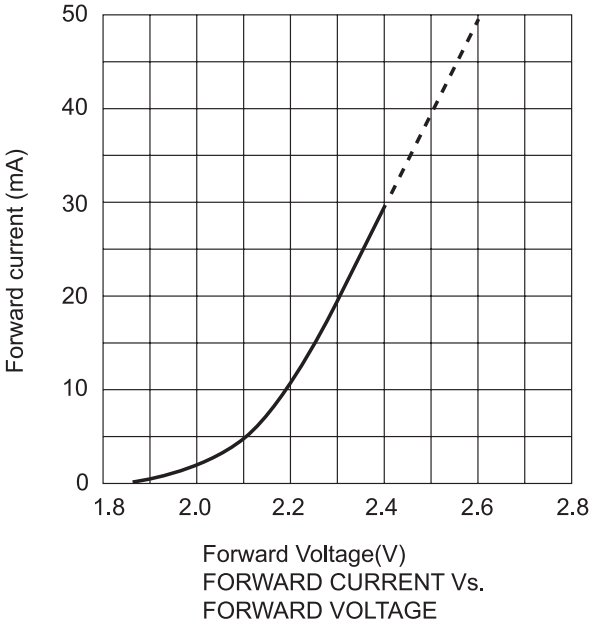


Figura 1
Caratteristica tensione-corrente di un led giallo



di Daniele Cappa

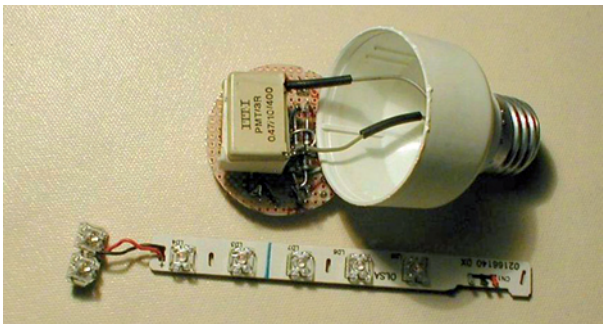


Figura 2
Vista interna della lampada

domestici sono probabilmente di tipo induttivo, dunque rischiamo di provocare un involontario, ma utile, rifasamento. Una premessa, il led è strutturalmente un componente che mal si adatta al funzionamento in alternata, la sua tensione inversa di rottura è generalmente piuttosto bassa, con 4 o 5V si hanno correnti inverse dell'ordine di alcune decine di microA, una corrente inversa enorme. Ancora un piccolo aumento e il diodo rischia la distruzione per effetto zener. In un diodo comune, il solito 1N4004, 1N4007 o simili la tensione inversa di rottura può essere pari a 800 – 1000V, a questi livelli di tensione la corrente inversa è pari a soli 5 μA, dunque un led il cui funzionamento è previsto in alternata è bene sia preceduto da un diodo normale oppure da un ponte a diodi.

Il prototipo e il suo dimensionamento

Ho utilizzato lo zoccolo, completo di vitone "E27", recuperato da una lampada a basso consumo passata a miglior vita. Il prototipo è dunque stato assemblato su un ritaglio di basetta millefori di forma circolare (diametro esterno pari a 48 mm) su cui trovano posto 24 led. Nella figura 2 sono visibili, davanti alla lampada, le piastrine da cui sono state ricavati i led, probabilmente si tratta del segnalatore laterale inserito nello specchietto esterno di alcune Mercedes.
Il modello utilizzato consuma 35mA con una tensione pari a 2.35V, dunque 24 led hanno bisogno di 56.4V (24 x 2.35). Considerando la caduta del ponte a diodi (1.2V) sul condensatore devono cadere 162,4V (220 – 1.2 – 56.4), dunque la sua reattanza dovrà essere pari a 4640 Ω, a 50 Hz corrisponde a una capacita di 686nF che ovviamente arrotonderemo al valore commerciale più vicino (680nF). Nel prototipo ho utilizzato due condensatori al poliestere in parallelo da 470 e 220 nF (690nF totali), la tensione di lavoro dovrà essere per entrambi pari a 400V, non di meno pena una bella esplosione del componente! Lo schema è ora più chiaro, il condensatore si limita a fornire la caduta di tensione necessaria e funziona da limitatore di corrente ai led, il ponte a diodi fornisce ai led corrente continua, anche se pulsante a 100Hz. La serie di led si accende a luce piena, con il risultato di una lampada che potremmo valutare a occhio (!) da circa 20W… consumandone solo due… un rapporto stimato di uno a dieci non è niente male!
Come è visibile dalla figura 4 il prototipo utilizza led di colore arancio, poi è da considerare che la macchina fotografica in presenza di poca luce aumenta i tempi di apertura dell'obbiettivo con il risultato di vedere foto molto più luminose di quanto non sia in realtà. Un chiaro esempio sono i

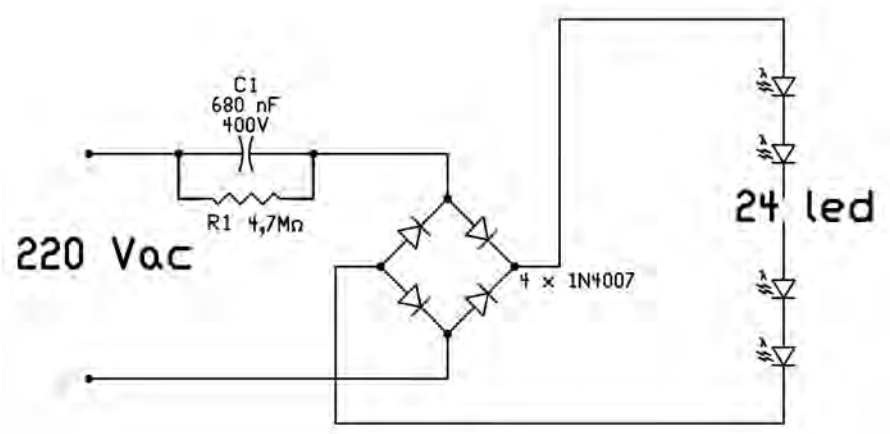


Figura 3
Schema elettrico della lampada a stato solido

display delle due radio, che sono straordinariamente luminosi. La lampada lasciata accesa per alcune ore diventa lievemente tiepida, né il condensatore e neppure i diodi del ponte devono assolutamente riscaldarsi! È necessario un avvertimento… stiamo lavorando in presenza di tensione di rete, dunque la lampada non andrà mai toccata quando è inserita nel portalampade, qualsiasi operazione sia necessaria durante le prove e il montaggio va eseguita con entrambi i fili di rete scollegati. Ogni volta è necessario scaricare il condensatore ponendo per un attimo in corto i due terminali dello zoccolo E27, i led emetteranno un breve, ma intenso lampo, a sottolineare quanta energia è accumulata nel condensatore. La carica del condensatore permane per molte ore dopo lo spegnimento della lampada, alcuni modelli sono in grado di rimanere carichi per giorni! La resistenza R1 risolve in parte questo problema, posta in parallelo al condensatore lo scarica, molto lentamente in verità, permettendo di maneggiare la lampada senza problemi già pochi secondi dopo il suo spegnimento. È evidente che un prototipo assemblato con questo sistema non è assolutamente stagno, patisce l'umidità e va rigorosamente tenuto lontano da mani che non sappiano come toccarlo; dunque lontano da bambini, nonni, o chiunque patisca la presenza della rete sulle proprie dita.

Vediamo ora il dimensionamento per altri led...

Due numeri circa la tensione di funzionamento dei led ad alta luminosità… un modello a luce gialla ha la tensione di funzionamento compresa tra 2.3 e 2.8V, a luce arancio (il mio) tra 2 e 2.4V; modelli a luce bianca hanno tensioni più alte, tra 2.9 e 3.8V. La corrente di lavoro è sempre compresa tra 20 a 50 mA. E' comunque bene controllare le caratteristiche su un datasheet, oppure in modo empirico con tester, alimentatore e resistenza o trimmer.
Abbiamo ora tensione e corrente di funzionamento del led utilizzato, ammettiamo (come puro esempio)



Figura 4
La lampadina in funzione

$$Z = \frac{V}{I} = \frac{220\text{-}56{,}4\text{-}1{,}2}{0{,}035} = 4640\Omega$$

$$C = \frac{1}{2\pi fZ} = \frac{1}{2 \cdot 3{,}1416 \cdot 50 \cdot 4640} = 686nF$$

Figura 5
I calcoli necessari a determinare il valore di C1

che il nostro led fornisca una luminosità adeguata con un corrente di 30mA, a cui corrisponde una tensione di 3.3V. Sul supporto che utilizzeremo trovano posto 35 led; la tensione di funzionamento dei led è dunque (3.3 x 35 = 115.5V), per ottenere la caduta di tensione sul condensatore dobbiamo sottrarre la tensione necessaria ai led e la caduta sui diodi del ponte: (220 – 115.5 – 1.2 = 103.3V). Dunque l'impedenza a 50Hz del condensatore dovrà essere pari a 3443.33 Ω (103.3 / 0.03), caduta di tensione diviso corrente (i 30mA del led). Alla frequenza di rete il valore del condensatore che otteniamo dalla seconda formuletta è di 924 nF, che ovviamente arrotonderemo a 1µF. Impiegheremo un componente di tipo ceramico, o poliestere, con una tensione di lavoro non inferiore a 400V. Assolutamente NON sono adatti modelli multistrato che raramente superano i 50V!! La resistenza in parallelo al condensatore non è critica, il suo compito è di scaricare il condensatore in tempi ragionevoli, i citati 4.7MΩ lo scaricano in circa tre secondi, valori più alti sono meno veloci, mentre valori più bassi sono più rapidi, ma devono dissipare più potenza… valori compresi tra uno e dieci megaOhm vanno comunque bene pur utilizzando esemplari da ¼ di watt.
Con queste poche righe abbiamo visto come sia possibile realizzare un oggetto che solo pochissimi anni fa sarebbe stato considerato una follia elettronica. Dal punto di vista economico tutto dipende dal prezzo a cui è possibile reperire i led ad alta luminosità, possibilmente di colore bianco, ma questo non esclude la possibile realizzazione di altri gadget utili. Un esempio può essere il sostituto del classico punto luce (quello il cui scopo è di non farci sbattere la testa durante le levate notturne) di solito realizzato con una lampadinetta al neon. Tre led come quelli impiegati potrebbero brillantemente risolvere il problema, fornendo una intensità e un fascio di luce sconosciuto alla vecchie spie al neon!
Utilizzando i calcoli riportati in figura 5 prestate la massima attenzione alle dimensioni fisiche del condensatore di cui avete appena calcolato il valore.

More Info Please!
Inserisci il Codice **268034** alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

...Un momento di confronto unico!!!

Nel Panel Relatori sono presenti ruoli e funzioni aziendali specifiche per illustrare nel dettaglio gli argomenti in agenda. Rappresentanze Istituzionali, Manager di alto profilo e Specialisti nel settore, condivideranno la loro esperienza con i partecipanti. Ecco i protagonisti:

...ben 20 Relatori confermati:



Antonello Colosimo
Vice Alto Commissario per la lotta alla contraffazione
Corte dei Conti



Maria Ludovica Agrò
Direttore generale
Ufficio Italiano Brevetti e Marchi



Paolo Balistrieri
Responsabile Anticontraffazione
LVMH Italia - Gruppo Louis Vuitton



Vanni Volpi
Responsabile del Dipartimento di Proprietà Intellettuale e della lotta anti-contraffazione
Gruppo Gucci



Giorgio Floridia
Professore ordinario - Avvocato
Studio Legale Floridia



Daniela Mainini
Avvocato
Studio Legale Mainini e Associati



Furio Suggi Liverani
Direttore dipartimento Ricerca ed Innovazione
Illycaffè



Daniele Lingua
Direttore Proprietà Intellettuale
Somartec (Ferrero)



Daniele Pieraccioli
Direttore Proprietà Intellettuale
Dompé



Francesco Fallica
Colonnello
Guardia di Finanza



Diego Pallini
Partner
Notarbartolo & Gervasi



Marco Carone
Avvocato, Dottore commercialista
Studio Legale Carone & Partners



Sergio de Sanctis, Direttore Ricerca Sviluppo e Innovazione
Gruppo Sapio



Raimondo Galli
Avvocato
Studio Legale Galli & Associati



Giancarlo Migliori
Titolare
MrgoodIDEA



Vanessa Franchini
Avvocato
Gianni Origoni Grippo & Partners



Roberto Dini
Amministratore Unico
Metroconsult



Antonio Mario Pizzoli
Consulente in Proprietà Industriale ed intellettuale
SIB - Società Italiana Brevetti



Luca Sandri
Avvocato
Studio Legale Sandri e Lameri



Giorgio Secchi
Area Manager Far East
Luxlife

Marchi, Brevetti e Contraffazione

Le ultime novità in materia di proprietà industriale e sistema brevettuale e tutti gli strumenti per vincere la lotta alla contraffazione

Milano, 4 e 5 dicembre 2007 - Starhotel Ritz

Un'occasione unica per confrontarsi con professionisti ed aziende e fare il punto della situazione sulle ultime novità previste dal Codice della Proprietà Industriale, dalla Riforma del sistema brevettuale e sulle modalità con cui le Istituzioni e le Aziende affrontano la sfida della contraffazione.

Al centro del dibattito i seguenti argomenti:

- Le **ultime novità** in materia di marchi e brevetti
- Le strategie e le tecniche vincenti per **valorizzare** al meglio il proprio **patrimonio intellettuale**
- Come le Istituzioni e le Aziende affrontano la **minaccia** della **contraffazione**
- Le **tutele legali**
- Le tecniche di gestione del patrimonio industriale dalla **minaccia** della **Cina** e dei **Paesi emergenti**

...in agenda 6 casi di successo:

Come un Gruppo internazionale gestisce e tutela il proprio patrimonio intellettuale e le sue privative: l'esperienza del **Gruppo Gucci** nella tutela dei suoi marchi e nella lotta anti-contraffazione

Come l'investimento in proprietà intellettuale può costituire una leva strategica di nuovi business: il caso **Illycaffè**

Come **Louis Vuitton** gestisce la lotta alla contraffazione e la tutela del proprio brand

Come un gruppo industriale protegge l'innovazione tecnologica: il caso dei gas tecnici e industriali del **Gruppo Sapio**

La contraffazione di un farmaco 'blockbuster': il caso eritropoietina - **Dompè**

La Protezione dei diritti di proprietà industriale per beni di largo consumo in Cina e Turchia - **Somartec (Ferrero)**

Workshop pre-convegno
3 dicembre 2007
La tutela dei modelli e dei disegni industriali
a cura di
Avv. Gualtiero Dragotti

Media Partner

Codice MIP 268037

Per ricevere maggiori informazioni riguardo al convegno in oggetto, compili questo coupon e lo invii a:
Istituto Internazionale di Ricerca - Tel. 02.83847.242 - Fax 02.83895.118 - E-mail: acanzi@iir-italy.it

Marchi, Brevetti e Contraffazione priority code: D3270FARE

Nome:.....
Cognome:.....
Funzione:.....Società.....
Tel.:.....Fax:.....
N° di cellulare:.....E-mail:.....
Settore Merceologico:.....
Indirizzo:.....
Città.....Cap:.....Prov.:.....

I dati saranno trattati dall'Istituto Internazionale di Ricerca secondo quanto previsto dal D. Lgs 196/03 sulla Tutela della Privacy

Istituto Internazionale di Ricerca
1987-2007 • 20 anni in Italia

www.farelettronica.com



Il portale della rivista

La tua azienda non conosce Fare Elettronica? Richiedi una copia omaggio!

Per far conoscere Fare Elettronica ad un pubblico sempre più vasto abbiamo dato alle aziende la possibilità di valutare prima di acquistare secondo la formula "try&buy"

Fare Elettronica è oramai giunta al suo ventitreesimo anno di vita, tuttavia vi sono potenziali lettori che non conoscono ancora la Rivista ma ne sono attratti vuoi per motivi professionali, vuoi per motivi "scolastici" o per semplice passione. Ma che fare se non si ha una edicola vicina che permetta di sfogliare la rivista e decidere se acquistarla o meno? Noi della Redazione ci siamo posti questo problema e abbiamo trovato la soluzione più semplice: vuoi valutare la rivista? Basta chiederla! Dalla home page del portale della rivista (www.farelettronica.com) dal menu "Rivista" è infatti possibile accedere alla voce di menu "Richiedi copia saggio". Cliccando su questa voce si aprirà il form in cui potete inserire alcune note alla vostra richiesta prima che questa venga inoltrata cliccando sul pulsante "Invia la richiesta". La vostra richiesta giungerà al nostro personale di redazione che provvederà ad inviarvi gratuitamente a casa vostra la prima copia raggiungibile. Niente numeri arretrati o obsoleti, ma l'ultimo numero in uscita!! Per poter inviare l'ultimissimo numero è possibile che la vostra richiesta non sia soddisfatta immediatamente, ma state sicuri che riceverete la vostra copia saggio! Dopo aver sfogliato la rivista deciderete con tutta calma se salire a bordo del "Treno Fare Elettronica" che vi porterà nei meandri di questa affascinante materia attraverso progetti, dossier, monografie, eventi e news sempre aggiornati! Per poter richiedere la copia saggio è necessario essere registrati ed aver inserito correttamente il proprio recapito al quale verrà spedita la rivista. L'iniziativa è riservata alle aziende o ai titolari di partita IVA.



con il patrocinio del
MINISTERO delle COMUNICAZIONI

mostra mercato

Fxpo Elettronica® 2007

ERBA (Como)

LARIO FIERE
V.le Resegone,3

10·11 novembre 2007

ore 9/18



RIF. FARE ELETTRONICA

Codice MIP 268039

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

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

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

NEWS ON LINE!
Vai nel sito www.blunautilus.it
e iscriviti a NEWSLETTER:
10 giorni prima di un evento
riceverai una e-mail di promemoria

VALE COME RIDOTTO

Radioregolamenti, codici e abbreviazioni

Abbiamo già trattato una quarantina di gruppi di questo linguaggio internazionale che rimane un caposaldo nella preparazione degli operatori radio e basterebbero già da soli per allacciare un dialogo abbastanza variegato fra persone di differente lingua, ma continueremo la descrizione degli altri gruppi che riguardano specifici argomenti.

QSA? (Quale è la forza dei miei segnali?)
È una domanda prettamente tecnica, che non riguarda la comprensibilità (QRK), né il livello dei disturbi (QRM;QRN) ma semplicemente la forza del segnale! Potremmo infatti ricevere un segnale molto disturbato, tanto da risultare quasi incomprensibile, ma notare una sua forza o, per usare altro termine analogo una sua consistenza molto buona! La valutazione del segnale si dichiara con cinque diverse cifre: 1 = scarsamente percepibile; 2= debole; 3= quasi buona, 4 = buona; 5 = molto buona. La domanda è più frequente di quanto si possa pensare perché permette all'operatore sia di regolare la potenza del proprio trasmettitore, sia di valutare se sia utile o meno scegliere altra frequenza di lavoro o altra banda di collegamento: se un corrispondente mi dichiarava QSA 4, QRM 4, lo invitavo a cambiare frequenza di lavoro; Se invece mi dichiarava QSA 1 e non avevo possibilità di aumentare significativamente la mia potenza di emissione, lo invitavo a provare una banda di frequenza che offrisse miglior propagazione. Sovente un QSA 1 in banda dei 4 MHZ, diventava QSA 3 in quella degli 8 MHZ, o viceversa!

QSB? (I miei segnali sono afflitti da evanescenza?)
Il ben noto fenomeno del "fading",dovuto a mutazioni improvvise nelle condizioni di propagazione, influenza spesso i collegamenti rendendo segnali anche molto forti momentaneamente quasi impercettibili! Notavo più frequentemente tale disturbo sulle "bande alte", ossia sui 16 e sui 22 MHZ, utilizzate per collegamenti su grandi distanze, ma chiunque utilizzi apparati in banda VHF o UHF li avrà notati sicuramente: per queste bande a frequenza molto alta il fading è provocato dagli spostamenti nell'ambiente, con transito in "zone d'ombra". Mentre per i ricevitori VHF e UHF l'evanescenza puo' ridursi utilizzando circuiti pluriantenna, ossia con antenne dislocate, scelte automaticamente per l'utilizzo del segnale più robusto, nel campo delle onde corte si puo' contare soltanto sulla dinamica dei circuiti di controllo automatico di guadagno (CAG), e se il QSB è tale da impedire il collegamento non rimane a volte che cambiare banda di frequenza!

QSC? (Siete una stazione di nave a basso traffico?)
Fatta eccezione per le navi passeggeri, le navi da carico, anche di grosse dimensioni, rientrano nel QSC, ossia sono previste e organizzate per un basso volume di traffico radio, almeno per il traffico inteso come commerciale (telefonate, telegrammi, fax). Ciò non significa che il traffico in appoggio alla navigazione, alle informazioni meteo, ai segnali di sicurezza, non possa essere di notevole volume e di estrema importanza! In ogni caso, nella mia esperienza di Marconista, questo gruppo non ho avuto mai modo di ascoltarlo o di utilizzarlo.

QSD? (I miei segnali risultano mutilati?)
Difetti nella manipolazione telegrafica o nella emissione del microfono sono più frequenti di quanto si creda: i relè sono congegni utilissimi, ma come tutti i dispositivi meccanici sono soggetti ad inceppamenti e ad ossidazioni, che si traducono appunto in "mutilazioni" nella trasmissione! Una volta, quando già ero giunto alla fine di un lungo telegramma, la stazione di Roma mi accusò "QSD" e fui costretto a sospendere la trasmissione: aperto l'apparato trasmittente, mi adoperai per un paio d'ore a ripulire e centrare accuratamente un massiccio relè di manipolazione i cui contatti si erano ricoperti di una tenace patina di salsedine. Ma anche le cornette telefoniche mi procura-

Il codice Q (V)



di Franco Malenza

rono qualche problema, in quanto i miei "clienti", membri degli equipaggi che telefonavano alle loro mogli, si accanivano furiosamente sul cavo, attorcigliandolo, percotendolo o tirandolo con violenza; alfine si provocava qualche falso contatto e l'immancabile accusa di QSD da parte della stazione costiera!

QSE? (Quale è la deriva stimata del mezzo di salvataggio?)
È il primo di sedici gruppi che riguardano le operazioni di ricerca e di salvataggio. Ma è anche ormai un pò superato dai tempi e dalle tecnologie. Con il termine drift, traducibile marinarescamente in "deriva" si intende lo spostamento complessivo dovuto all'effetto del vento, della corrente e dalla deviazione di rotta. Navi o velivoli impegnati in operazioni di salvataggio a seguito di un affondamento, sapendo che una lancia di sopravvissuti si trovava in alto mare, si adoperavano per calcolarne lo spostamento, anche quando dalla stessa lancia fosse stata dichiarata una posizione approssimativa, al fine di orientare le ricerche in una zona delimitata. Attualmente le lance di salvataggio, munite di GPS o di impianto satellitare, possono infischiarsene di calcoli di deriva o di stime di posizioni approssimative! Quando invece l'apparato radio della lancia era costituito da un trasmettitore ad onda media (500 Khz), alimentato a manovella e munito di manipolatore automatico che ripeteva lo SOS, il radiogoniometro delle navi era il più sicuro mezzo per una localizzazione.

QSF? (Avete effettuato il salvataggio?)
Mi accadde una sola volta di sentire, dopo un concitato traffico di salvataggio per un peschereccio canadese, durante una furiosa burrasca al largo di Norfolk, una nave norvegese dichiarare. "QSF. proceeding to Norfolk base with 4 persons not injured", Il cargo Norvegese tranquillizzava così quanti avevano preso parte alle operazioni di salvataggio e dichiarava di trasportare a Norfolk i quattro tratti in salvo, che non necessitavano di particolari cure. Attualmente il traffico radio riguardante soccorsi viene diretto da appositi centri costieri in grado di ordinare operazioni con velivoli, elicotteri o speciali imbarcazioni, ma esistono ancora zone di mare dove l'iniziativa deve essere presa direttamente da navi nella zona! Le zone marine sono più estese di quanto si possa immaginare e le condizioni meteorologiche rendono spesso difficoltosa l'organizzazione di soccorsi per i centri di assistenza che si trovino a grande distanza: le navi in zona sono pertanto ancora le principali protagoniste del soccorso.

QSG? (Devo trasmettere n. ... telegrammi per volta?)
È un gruppo ormai in disuso. Si dovrebbe impiegare quando sia difficile interrompere la trasmissione del corrispondente e gli si chiede pertanto di inviargli un certo numero di messaggi prima di ottenere la conferma (QSL) a trasmettere i successivi. In tutta la mia carriera di Marconista non mi è mai successo di adoperarlo, tuttavia qualche vecchio collega che aveva adoperate vecchie apparecchiature, mi narrò di averlo usato.

QSH? (Siete in grado di rientrare alla base usando il vostro impianto radiogoniometrico?)
La richiesta era particolarmente rivolta a natanti che fossero stati impegnati in operazioni di soccorso e salvataggio. Il radiogoniometro e il Radar erano, fino a un decennio fa, gli unici e preziosi strumenti che permettevano di destreggiarsi nella nebbia, oggigiorno la domanda potrebbe estendersi ai più moderni sistemi GPS. In ogni caso la domanda prevedeva una richiesta di eventuale assistenza per una radiolocalizzazione che permettesse al natante o al velivolo di correggere la propria rotta al fine di un "rientro" al porto o alla base di destinazione: Mi è successo di sentirlo usare per il caso, non troppo raro, di incontri in mare per il trasbordo di malati o di mezzi, soprattutto quando i Radars erano influenzati da eccessivi bersagli e si adottava il radiogoniometro per incrociare la propria rotta con quella del natante da incontrare.

QSI (Non sono in grado di interrompervi nella vostra trasmissione; oppure "potete informare ... che non sono in grado di interrompere la sua trasmissione, su Khz ...)
Ecco uno dei rari gruppi che non prevede risposta, ma solo affermazione. Nei collegamenti radio tele-

grafici si usa interrompere il corrispondente con una serie di punti, in telefonia con il classico "break", ma quando tale interruzione non sia percepita si attende il momento opportuno per dichiarare il QSI. Malgrado le moderne apparecchiature siano provviste di desensibilizzatore automatico che agevola la ricezione di segnali durante le pause di emissione, non sempre tali circuiti sono efficienti e paradossalmente nelle recenti radio ricetrasmittenti previste dal sistema GMDSS le interruzioni non sono possibili.

QSJ? (Quale tassazione devo registrare, inclusa la tassa interna?)
Mentre lo scambio di messaggi di servizio è normalmente a titolo gratuito, i telegrammi o conversazioni telefoniche di tipo privato o per conto nave sono ovviamente soggetti a tassazione. La tassazione è composta da una quota che va accreditata alla compagnia che tratta la nave, (vedi il QRC) e dalla tassa prevista per la linea interna del Paese (chiamata LL). Tale tassazione varia a seconda delle varie stazioni costiere, delle singole Nazioni e anche dal tipo di conversazioni, se private o riservate agli equipaggi! Curioso il fatto che dette tassazioni sono per tradizione riferite non al classico Dollaro USA o al moderno Euro ma all'antico FRANCO ORO SVIZZERO! Il Golden Franc è stato per anni considerato la moneta più stabile ed affidabile su cui basare il cambio ed era compito del Marconista tenersi aggiornato, mese per mese, sul cambio assegnato. A titolo indicativo, nel Maggio 1997 un Gold Franc equivaleva a 2335,27 lire, mentre a luglio dello stesso anno valeva 2362,68 lire e a luglio 2000 era salito a 2709,64 lire. Attualmente, per il traffico via satellite, gran parte delle tassazioni sono direttamente praticate in Euro o in Dollari USA e vengono direttamente inviate alle compagnie senza la dichiarazione del Marconista, figura quasi soppressa! Tuttavia è ancora d'uso comune il chiedere alla fine di un collegamento il QSJ? alla stazione con cui si sia trattenuto traffico.

QSK? (Potete sentirmi fra i vostri segnali? Se potete sentirmi, posso io interrompervi durante la vostra emissione?)
Si è già trattato questo argomento alla voce QSI. In sostanza si chiede se sia o meno possibile un dialogo con interruzione in qualsiasi momento.

QSL? (Mi potete confermare di aver ricevuto?)
Questo gruppo di codice è quasi un simbolo, particolarmente per la telegrafia! Ad esso si rifanno le famose cartoline QSL in uso fra radio amatori per confermarsi reciprocamente via posta i collegamenti, con tanto di rapporti sulla qualità dei segnali! Alla fine di un telegramma o di un messaggio si attende comunque la dichiarazione del corrispondente e il QSL, eventualmente accompagnato dall'ora precisa, vale a conferma della avvenuta perfetta ricezione, che verrà registrata sul libro di stazione e che costituisce prova della ricezione o trasmissione del messaggio. Il QSL non ammette dubbi o incertezze: è il segnale che il messaggio è stato completamente ricevuto e la promessa che verrà recapitato! Nato per la telegrafia è valido anche per la telefonia o per altra forma di comunicazione: talvolta lo si sente in alfabeto fonetico con la formula "Quibeck Sierra Lima" e non c'è Marconista o addetto alle comunicazioni che non lo abbia pronunciato decine di volte!

QSM? (Devo ripetervi l'ultimo telegramma trasmessovi (o qualche telegramma precedente?))
È appunto per il caso di incertezza su qualche cifra o parola, che ci si dichiara disponibili a ritrasmettere uno o più messaggi o che si richiede una conferma, e si usa questo gruppo per evitare ogni dubbio ed ottenere un chiaro QSL. Attualmente è impiegato raramente.

QSN? (Mi sentite, oppure sentite ... sulla frequenza di Mhz ... ?)
Lo si impiega normalmente quando si adoperino contemporaneamente due diversi apparati e con l'uno si chieda conferma del collegamento dell'altro. Per esempio quando in onde corte si chieda conferma della ricezione del ricetrasmettitore VHF. Talvolta si avvisa una stazione costiera che un'altra nave la chiama su di una certa frequenza: "QSN ICJB on channel 16?" Avete sentito che la nave ICJB vi sta chiamando sul canale 16? Scomparso il vecchio e glorioso traffico telegrafico in onde medie, i collegamenti su medie distanze si attuano normalmente sulla telefonia in onda medio corta (2182 Khz) e nella banda VHF e non è raro il caso che tali apparecchiature vengano impiegate contemporaneamente, per cui il gruppo QSN torna utile ad accertarsi che il collegamento sia attivo per entrambe!

Esistono ancora 55 gruppi del codice Q che descriveremo in seguito, alcuni dei quali rivestono notevole valore nei "dialoghi" via radio: le tecnologie moderne non hanno infatti eliminate le difficoltà di linguaggio e per gli operatori questa via "telegrafica" offre a tutt'oggi insostituibili vantaggi.

More Info Please!
Inserisci il Codice 268040 alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

Industrial

FRAMOS

think future

imaging solutions

Telecamere

Lumenera Lw570/Lw575

- 5 Megapixel (2592 x 1944 Pixel)
- Sensore CMOS Micron da 1/2.5"
- 7 fps alla massima risoluzione
- Qualità immagine 8 e 12 bit
- Colore

Lumenera USB2.0 Lm-Series

- Risoluzione da VGA a 1.4Megapixel
- Qualità immagine 8 e 12 bit
- Attacco industriale USB2.0 e GPI/O
- Dimensioni irdotte (44x44x56mm)
- Back focus regolabile
- SDK disponibile





Codice MIP 268043

FRAMOS ELECTRONIC VERTRIEBS GMBH
Centro Direzionale Colleoni
Pal. Taurus Ing. 2
Via Colleoni 3
20041 Agrate Brianza (Milano)

Lumenera corporation

Phone · +39.039.68 99 - 635
Fax · +39.039.68 98 - 065
info@framos.it · www.framos.it
info@framos.de · www.framos.de
info@framos.co.uk · www.framos.co.uk







Prima parte: *Settembre 2007*
Trasduttori di posizione (I)

Seconda parte: *Ottobre 2007*
Trasduttori di posizione (II)

Terza parte: *Novembre 2007*
Trasduttori di posizione (III)

Trasduttori

In questa puntata, ultima di teoria, continueremo la trattazione affrontando le varie tipologie di encoder e di commutatori codificati.

Commutatori codificati

Cominciamo parlando della categoria di trasduttori sicuramente più utilizzata che è, per l'appunto, quella dei commutatori codificati. La "fama" di questi componenti deriva principalmente dalle applicazioni "consumer", ovvero videoregistratori, stampanti, fotocopiatrici, autovetture, ecc., che, almeno fino a qualche anno addietro, ne facevano largo uso. Effettivamente il commutatore codificato è un componente economico, costruibile secondo le forme più disparate e, cosa non secondaria, fornisce un'informazione in formato numerico senza quasi bisogno di elettronica aggiunta. Costruttivamente si presenta come un commutatore rotante con il perno libero di ruotare sulle posizioni significative, salvo una debole ritenzione meccanica spesso necessaria per meglio individuare le posizioni stesse. In figura 1a possiamo osservarne la struttura di un modello che, per semplicità, è a soli 2 bit.

Come possiamo vedere in figura 1b, per ognuna delle 4 posizioni del commutatore, viene presentata una determinata configurazione d'uscita che, nel nostro caso, è in codice binario puro. Anche se la scelta di quest'ultimo sembrerebbe scontata e naturale, in realtà, come vedremo meglio, esistono altri tipi di codifica che garantiscono migliore immunità alle transizioni spurie. Il numero delle posizioni significative del commutatore equivale al numero dei bit disponibili in uscita elevato al quadrato e quindi, in definitiva, interessa direttamente la risoluzione del trasduttore. Nell'esempio in figura 1a abbiamo immaginato un commutatore codificato a 2 bit e quindi la risoluzione, in caso di movimento angolare, sarà 360° / 2^2 = 90°. Decisamente bassa, se confrontata con altri tipi di trasduttori! In commercio esistono anche trasduttori a 4 bit, ovvero con 16 posizioni significative, ma c'è da dire che la risoluzione non è il punto di forza di questi componenti. Infatti, il loro uso principale è il controllo di posizione di cinematismi a basso grado di precisione. Dicevamo, all'inizio, che un grosso vantaggio di questi trasduttori è di fornire un dato quantizzato puro, senza parti elettroniche aggiuntive. In realtà, e c'era da aspettarselo, le cose non sono poi così semplici. Non dimentichiamo, infatti, che abbiamo a che fare con dei contatti striscianti e quindi siamo esposti agli inconvenienti che questa tecnologia impone. Questi ultimi, in sostanza, sono di due tipi: incertezza di chiusura ed apertura del contatto elettrico (rimbalzo) e rumore generato durante il trascinamento dell'organo di contatto, chiamato "spazzola", sulle piste conduttive. Per il rumore, elettronicamente parlando, possiamo fare poco anche se, data la natura digitale del segnale d'uscita, esso non è particolarmente penalizzante. Accorgimenti in questo senso possono essere presi in fase costruttiva, ad esempio ricoprendo con un sottile strato d'oro o d'argento la spazzola e le piste conduttive. Questi metalli, com'è noto, possiedono proprietà tali che impediscono la formazione, sulla loro superficie, di composti ossidati che sono i principali responsabili del rumore di contatto. Quest'ultimo può anche aumentare a causa della presenza di polvere ed impurità varie, ed è per questo che i migliori modelli di commutatori codi-

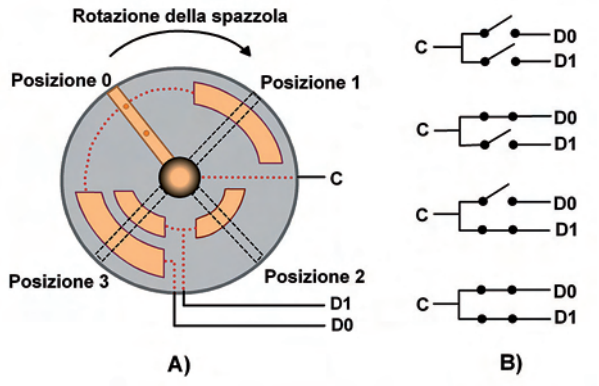


Figura 1 (a, b)
Commutatore codificato a 2 bit (a) e configurazione dei bit in uscita (b)

di posizione



di Grazia Ancona

ficati sono costruiti quasi del tutto ermetici. Per eliminare i rimbalzi, invece, abbiamo tre possibilità: 1) implementare, fra commutatore e circuito, un'interfaccia; 2) provvedere con apposite routine gestite dal firmware; 3) l'una e l'altra quando il dispositivo opera in applicazioni particolarmente critiche. La soluzione 1), che potremmo definire "hardware", è essenzialmente basata sull'uso di una rete integratrice a resistenza/capacità seguita da una porta logica, generalmente un inverter, del tipo trigger di Schmitt. Nella figura 2*a* troviamo un esempio di questa configurazione ove, per semplicità, si è immaginato un solo bit del commutatore codificato, nella figura denominato D0. Quest'ultimo, per simulare un funzionamento dinamico, si chiuderà, verso massa, dall'istante t_0 all'istante t_3 (vedi figura 2*b*).

Cerchiamo di capire la funzione dei componenti interessati. R1, R2 e C1 sono collegati in modo da comporre una classica rete integratrice, con R1 avente anche funzioni di pull-up. La funzione di R2 è quella di limitare la corrente di scarica, fornita da C1, sui contatti del commutatore nell'istante di chiusura. Per il dimensionamento della rete resistenza/capacità occorre fare qualche considerazione. Osservando la figura 2*b* possiamo notare come nell'istante commutazione, definito t_0, non avvenga una netta transizione di livello nel punto C del circuito (vedi grafico C). Due fattori concorrono per rendere imperfetta la forma d'onda nel nodo C. Il primo, responsabile degli impulsi molto brevi che appaiono prima della transizione, dipende dalle incertezze del contatto elettrico. Il secondo, responsabile dell'accentuata asimmetria dell'onda

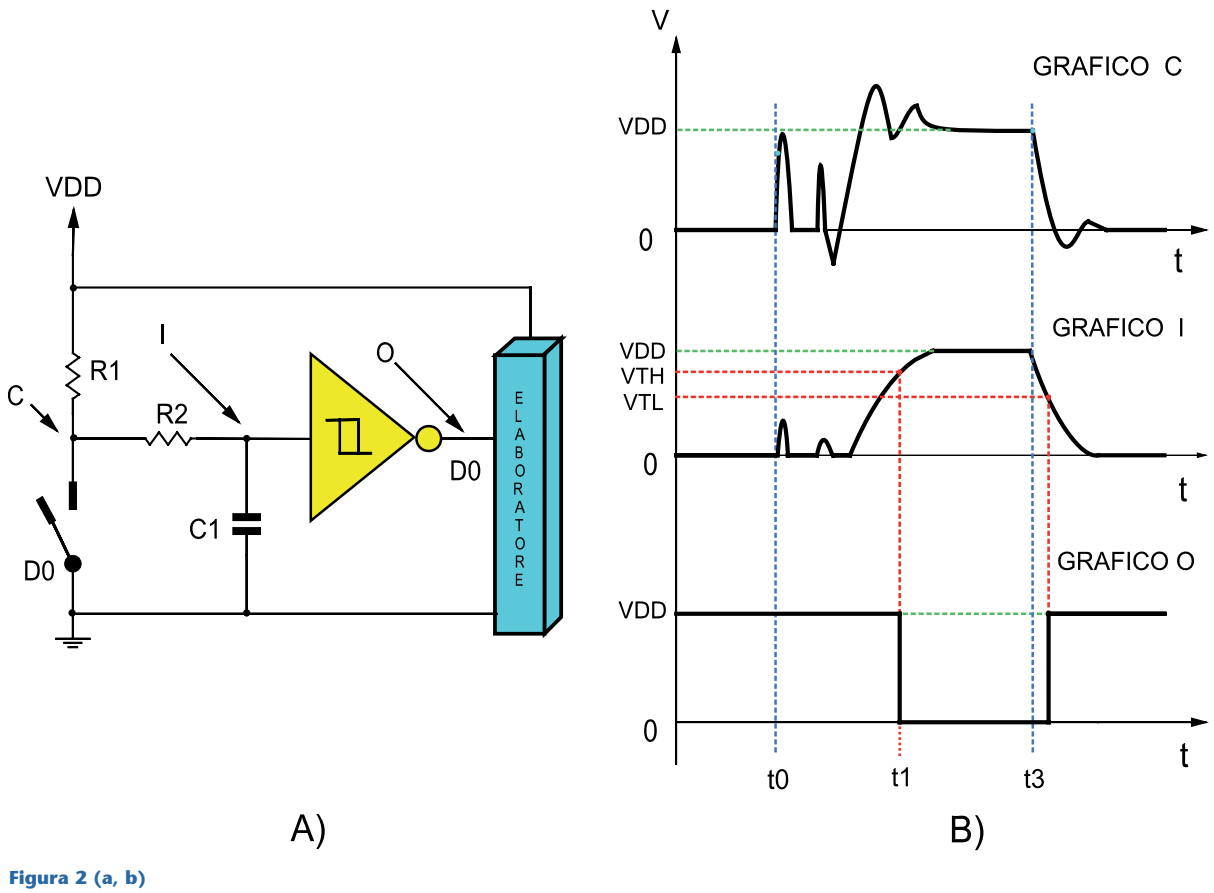


Figura 2 (a, b)
Tipico circuito antirimbalzo (a) e grafici esplicativi (b)

quadra, è causato dalle costanti di tempo comunque presenti (ad esempio l'induttanza parassita della resistenza R1). La forma d'onda che si presenta al nodo I (vedi grafico I), ingresso dell'inverter a trigger di Schmitt, è, per effetto della costante di tempo introdotta da R1 e C1, sicuramente migliore. In realtà, a ben vedere, presenta ancora una certa asimmetria, che potremmo definire di origine "capacitiva", dovuta alla presenza della suddetta rete integratrice. Nel grafico O appare, invece, la forma d'onda in uscita dell'inverter, pronta per essere immessa nell'elaboratore. L'effetto del trigger di Schmitt è evidente: l'onda quadra ha i fronti di salita e discesa ripidi e regolari, diremo che è in forma "canonica" assoluta. Questo "miracolo", naturalmente, ha un piccolo prezzo: il ritardo. Infatti, osservando il diagramma O, possiamo vedere che l'istante di chiusura dell'interruttore D0 è diverso dal corrispondente che giunge all'elaboratore, nel disegno t_1. In altri termini il sistema acquisisce il dato con un ritardo temporale pari a:

$$Ritardo\ di\ acquisizione = t_1 - t_0$$

Nel disegno di figura 2*b* questo intervallo di ritardo è stato un po' esagerato, comunque, in special modo nei sistemi di controllo in "real-time", occorre tenerne conto. Quella descritta non è la sola soluzione, vi sono altre configurazioni, più complesse e meno usate, che impiegano flip-flop tipo set-reset. L'utilizzo di questi componenti diminuisce, a scapito della semplicità circuitale, il ritardo di acquisizione sopra accennato. I costruttori di microprocessori e microcontrollori, sensibili a questo problema, offrono componenti che hanno una o più porte d'ingresso munite di trigger di Schmitt. In questo caso, se il numero delle porte d'ingresso è sufficiente a coprire i bit del commutatore codificato, basta la sola rete resistenza/capacità Per il calcolo di quest'ultima dovremmo sapere il tempo massimo di rimbalzo, di solito fornito dal costruttore, oppure, se questo non è noto, calcolare un tempo medio di 5 millisecondi. Bisogna poi tenere presente che questo tipo di trasduttori possiede una corrente massima, che difficilmente sarà superata poiché stiamo manipolando dati e non energia, e spesso anche una corrente minima. Non stupitevi per quest'ultima affermazione, di solito non riportata chiaramente nei cataloghi, ma comunque importante per la performance del trasduttore nel tempo. La corrente minima consente, durante il funzionamento, l'"autopulizia" dei contatti che, altrimenti, potrebbero essere ricoperti da sottilissimi strati di ossido. Un ulteriore problematica di progetto riguarda i gates in tecnologia CMOS, come ad esempio quelli appartenenti alla famiglia 74HCTXX, che possiedono una corrente di bias di circa 1 µA (in condizioni statiche). Questa corrente "fantasma", causata dai diodi di protezione d'ingresso di cui è fornita ogni porta logica, non permette l'uso di resistori con valori troppo alti, pena cadute di tensione inaccettabili. Valori superiori a 100 KΩ, per il resistore R1 dello schema di figura 2a, sono da considerarsi fuori limite. Fatte tutte queste considerazioni possiamo scrivere la relazione che ci permette di calcolare, con buona dose di approssimazione, il valore dell'accoppiata R1-C1 aventi, come dato noto, il tempo massimo di rimbalzo ed almeno il valore di uno dei due componenti (la capacità, in genere, è scelta arbitrariamente in base ad un valore disponibile e/o commerciale):

$$Tempo\ di\ rimbalzo = R1 \times C1$$

in cui devono valere le seguenti condizioni:

1. $R2 = R1 / 20$
2. $R1 \leq 100.000$

Accennavo, poco sopra, alla grande approssimazione della relazione che lega, fra loro, tempo di rimbalzo e la costante di tempo data da R1-C1. In effetti, nel calcolo di questi ultimi componenti, dovrebbero entrare le tensioni della soglia di commutazione della porta logica, logaritmi neperiani e via dicendo. Tuttavia, la relazione che ho presentato, fornisce un grado di precisione sufficiente per risolvere la maggior parte dei problemi di progettazione. Finora abbiamo preso in considerazione la soluzione "hardware" che, generalmente, è quella che dà risultati migliori a prezzo di complessità e costi superiori.

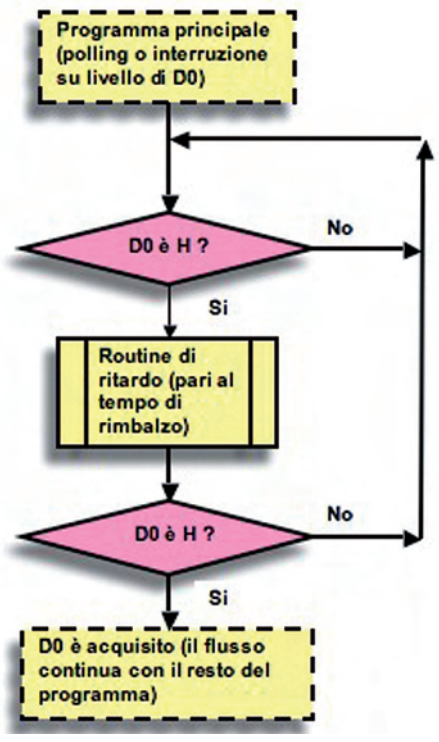


Figura 3
Diagramma di flusso per routine antirimbalzo software

Come accennavamo, tuttavia, il software dei dispositivi programmabili ci consente altre scelte… meno "impegnative". In questo senso, una delle tecniche più usate, è quella mostrata nel diagramma di flusso di figura 3.
L'algoritmo implementato è molto semplice, in sostanza si tratta di un ritardo che viene attivato non appena la CPU rileva un cambiamento di stato. Tutto ciò avviene nei primissimi istanti di commutazione (tempo t_0 in figura 2b), quando il livello del bit interessato è sostenuto da rapidi e ripetuti impulsi. Dopodiché la CPU, prima di effettuare un'altra lettura del bit, attende per un tempo prefissato, generalmente pari al tempo di esaurimento del rimbalzo che ci aspettiamo. Se quest'ultima lettura, che potremmo definire di "conferma", è uguale alla precedente il cambiamento di livello verrà interpretato come tale. In caso contrario si "scarterà" il risultato della prima lettura, in quanto generato da impulso spurio, ed il flusso del programma si riporterà nelle condizioni iniziali. È facile capire, osservando la figura 2b, come la seconda lettura, se di transizione vera si tratta, vada a parare nel tratto in cui l'impulso ha il suo livello positivo decisamente netto. Questo sistema ha lo svantaggio di occupare molto tempo della CPU, specialmente nell'iterazioni di attesa, e di non garantire al 100% il filtraggio degli impulsi spuri. Potrebbe essere, infatti, che la seconda lettura sia "imbrogliata" dalla presenza di un altro spike casuale. Per aggirare questo problema vi è una tecnica, definita "letture multiple", che fa eseguire al programma più letture del bit (di solito quattro o cinque) a brevissimi intervalli di tempo, e, se tutte le letture coincidono, la transizione di livello è presa per buona. Anche in questo caso, per evitare di usufruire troppo delle risorse temporali della CPU, fatto reso ancor più gravoso se i bit da leggere sono tre o quattro, è possibile usare, microcontrollore permettendo, un'interruzione software temporizzata. In questo caso, riportato nel diagramma di flusso di figura 4, vi è un risparmio di tempo pari a quello impiegato nelle iterazioni di attesa dell'esempio precedente.
Per utilizzare questa tecnica, come accennavo, bisogna disporre di una CPU dotata di timer interno la cui uscita possa interessare un'interruzione software. I microcontrollori della MICROCHIP, ad esempio, sono quasi tutti dotati di questa interessante funzionalità. Naturalmente occorrerà inizializzare, con apposite istruzioni, sia i registri dedicati che il tempo di ritardo dell'interruzione che, come avrete capito, deve coincidere con il tempo di rimbalzo. Le considerazioni fin qui viste, naturalmente, valgono non soltanto per i commutatori codificati ma anche ogni qualvolta dobbiamo elaborare un segnale fornito da un qualsiasi contatto interessato a movimenti rapidi e ripetuti.

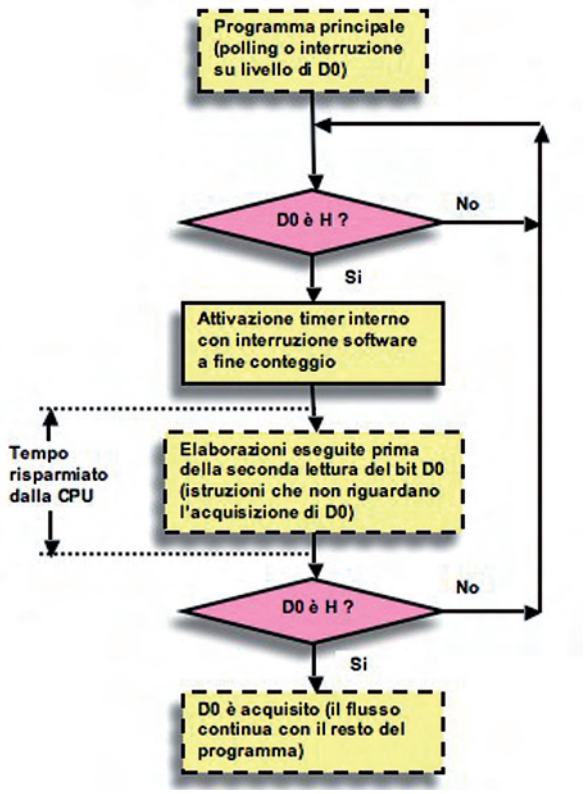


Figura 4
Diagramma di flusso per antirimbalzo software gestito con interruzione

Encoder assoluti

Gli encoder assoluti sono, per certi aspetti, la contropartita ottica dei commutatori codificati. Sono definiti assoluti poiché, per ogni posizione angolare significativa assunta, rendono disponibile un dato univoco, quest'ultimo quasi sempre in formato parallelo. Nella loro struttura più semplice sono composti da un disco di materiale plastico, generalmente plexiglass, il cui centro è solidale con l'asse di rotazione. Questo disco ruotante, per ogni posizione significativa, presenta una combinazione di finestre opache e trasparenti presenti, queste ultime, in numero pari al numero dei bit disponibili in uscita. Montata coassiale a queste finestre, ma fissata rigidamente sul corpo del trasduttore, vi è una batteria di fotodiodi, da un lato, e di LED, generalmente ad emissione infrarossa, sull'altro. Con questo tipo di struttura, il fascio di luce proveniente dal LED potrà essere intercettato da una finestra opaca o, viceversa, interessare il relativo fotodiodo se la finestra è trasparente. Nella figura 5 possiamo osservare un disegno che riprende, in termini grafici, quanto detto.
Osservando il disegno, appare evidente come il dato venga codificato dalla combinazione delle porzioni opache e trasparenti dell'organo ruotante, che per l'appunto è chiamato disco interferente. Naturalmente, a valle di ogni fotodiodo, vi è un circuito elettronico che ne elabora il segnale in modo da renderlo adatto ad essere interfacciato direttamente da logiche TTL-CMOS o affini.

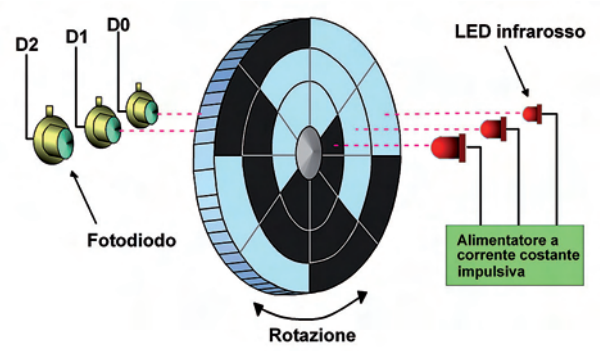


Figura 5
Principio di funzionamento di un encoder ottico assoluto

Accennavamo, parlando dei commutatori codificati, all'uso di sistemi numerici alternativi al classico binario puro. In effetti, quest'ultimo codice, potrebbe dare ambiguità di lettura in quanto, per alcune transizioni, possono esserci cambiamenti contemporanei di più di un bit. Questa condizione, considerando gli inevitabili disallineamenti e tolleranze presenti in trasduttori che possono ruotare anche a 2500 giri/minuto, potrebbe portare a casuali errori sul dato fornito in uscita. Per questo motivo, la codifica riportata sul disco interferente, è fatta con un codice che permette il cambiamento, passando da una posizione significativa ad un'altra, di un solo bit alla volta. Il codice Gray, detto anche codice "riflesso", soddisfa questa prerogativa che possiamo meglio osservare nella tabella 1. Per semplicità, in quest'ultima, si è supposto un dato a tre soli bits.
Osservando il disco interferente di un encoder che usa il codice Gray si capisce perché questo tipo di codifica, molta usata in questo tipo di trasduttori, è anche definita "riflessa". Infatti, è possibile osservare una simmetria fra i due semicerchi del disco, come se la configurazione si ripetesse con la sola eccezione del bit più significativo. Il codice Gray non è la sola alternativa al binario puro, che è definito anche come codice"1248". Nel riquadro di approfondimento, chi è interessato, troverà altri esempi di codifiche che usano i numeri binari.

Numero binario puro	Possibili ambiguità del codice binario (cambiamento di più di un bit)	Codice Gray	Posizione encoder
000	Nessuna	000	0
0**01**	Sì (**01** à **10**)	00**1**	1
0**10**		0**11**	2
0**11**	Sì (**11** à **00**)	0**10**	3
1**00**		**11**0	4
1**01**	Sì (**01** à **10**)	**111**	5
1**10**		1**01**	6
111	Nessuna	1**00**	7

Tabella 1
Numerazione binaria pura confrontata con codice Gray

In commercio vi sono encoder assoluti anche ad 8 bit, con risoluzione quindi di 1,4 gradi angolari circa. Questi, grazie ad apposite decodifiche, presentano in uscita dati in binario puro effettuando internamente una transcodifica Gray-binario. Se il numero dei bit in uscita è elevato, per ottenere trasduttori miniaturizzati e precisi, si preferisce ricorrere a diodi laser invece dei soliti LED. Tra l'altro, In alcuni modelli particolarmente compatti e ad alta risoluzione, i dati sono forniti con un semplice protocollo seriale. Gli encoder assoluti sono molto usati in sistemi di controllo a medio-bassa precisione ma che richiedono, per contro, alta affidabilità e velocità angolari ragguardevoli. La tipologia del segnale in uscita, digitalizzato e compatibile elettricamente con i principali standard, rende questi trasduttori particolarmente adatti al controllo di sistemi numerici in real-time.

Encoder incrementali

Questa categoria di trasduttori, "cugini" degli encoder assoluti visti prima, fornisce in uscita un treno continuo di onde quadre per tutto il tempo di rotazione dell'albero. Questo segnale digitale, di per sé, non contiene nessuna informazione assoluta sulla posizione angolare assunta in un dato momento. In effetti, senza appositi circuiti d'interfaccia, non è possibile ottenere alcuna informazione utile da questo tipo di trasduttori che, tuttavia, sono usati ancor più degli encoder assoluti. Le ragioni di questa "popolarità" sono da ricercarsi nell'alta velocità angolare che tali dispositivi possono garantire (in alcuni tipi fino a 9.000 giri per minuto), nell'incredibile risoluzione (in modelli particolari si arriva anche a 0,0001° angolari per ogni impulso fornito!) e nell'enorme versatilità d'uso che deriva proprio dalla loro… mancanza di "specializzazione". Nella figura 6 è possibile vedere lo schema costruttivo di uno di questi trasduttori in cui sono evidenti, come accennavo, le somiglianze con i loro "cugini", ovvero gli encoder assoluti. Il disco interferente, stavolta, non presenta particolari configurazioni. Vi sono soltanto due ordini di finestre trasparenti, parzialmente disallineate rispetto alla radiale del disco, più una sola finestra su un terzo ordine. Appare evidente come, per ogni apprezzabile rotazione, i fotodiodi sono interessati sempre solo da due condizioni che, in termini di segnale d'uscita, producono per l'appunto un'onda quadra. Quest'ultima è disponibile su tre fili, poiché tre sono i fotodiodi interessati, e presenta un andamento simile a quello di figura 7. Osservandola, possiamo fare alcune considerazioni necessarie alla comprensione del funzionamento di questo tipo di trasduttore.

Cominciamo col dire che i segnali A e B hanno il periodo, segnato con T in figura 7, inversamente proporzionale alla velocità di rotazione. In altri termini, il periodo dei segnali A e B, pur essendo in ogni caso identico, può variare in accordo con la velocità angolare del disco interferente. Questa particolarità non è stata evidenziata nel disegno ove, per semplicità, si suppone che la rotazione del trasduttore sia a velocità costante (impulsi ripetuti con periodo uguale). Si potrebbe osservare che sarebbe bastato un solo segnale, indifferentemente A o B, per trasformare un piccolo arco di rotazione in un impulso ma, in questo caso, non avremmo avuto modo di capire la direzione del movimento. Per ovviare a questo problema i segnali sono due e presentati in uscita con uno sfasamento, sempre costante, di 90°. Infatti, come si può osservare nel disegno di figura 7, supposto che la rotazione cominci nell'istante t0, il primo impulso a presentarsi sarà A. Tale condizione ci permette di conoscere il

Codici binari

Tutti sanno che le informazioni digitali sono spesso rappresentate, nei sistemi che ne fanno uso, dal codice di numerazione binario. Quest'ultimo, nella quasi totalità dei casi, è poi a sua volta trasformato in una struttura di simboli a noi più congeniale come, ad esempio, il sistema numerico decimale. Sono pochi i casi in cui il dato nella sua forma più pura, composta da due soli stati, ci viene presentato così com'è. Un LED che si accende per segnalare un evento è un modello, piuttosto comune, di un'informazione che ci viene offerta in codice binario puro. Il processo che trasforma quest'ultimo in un codice più "umano", o, se vogliamo, più direttamente manipolabile, è chiamato transcodifica. In ogni caso, alla base di tutto, vi è sempre un sotto-sistema che genera un'informazione digitale (un convertitore A/D, un trasduttore, ecc.). Per motivi vari, sempre legati a problematiche di processo, l'informazione fornita dal sotto-sistema a volte non è rappresentata da un numero binario puro. In altri termini, potremmo disporre di una grandezza che pur essendo espressa con una serie di classici simboli binari, ovvero "0" ed "1", non è riconducibile direttamente al numero che si vuole rappresentare. Sappiamo, infatti, che un numero binario può essere convertito in decimale sommando il risultato delle potenze di 2, con esponente uguale al "peso" della posizione della cifra binaria, moltiplicate per la cifra binaria stessa. Facciamo un esempio:

101 = 5 poiché (partendo dalla cifra LSB) si ha: $(2^0 \times \mathbf{1}) + (2^1 \times \mathbf{0}) + (2^2 \times \mathbf{1}) = 5$

E questo per il binario puro. Supponiamo, ora, di scrivere la cifra 5 con il codice Gray che pure utilizza le stesse cifre del binario:

111 = 5 (in codice Gray)

Come si vede chiaramente la stringa binaria in codice Gray è diversa da quella in binario puro, inoltre non può essere trasformata in un numero decimale con l'algoritmo esaminato. L'unica uguaglianza fra i due codici, a parte l'uso dei stessi simboli, è nel numero delle cifre. Oltre al codice Gray, già visto, esistono altri sistemi di codifica che offrono peculiarità sfruttabili in determinate applicazioni. Di seguito ne elenchiamo alcuni dei meno usuali tralasciando, per brevità, i codici che forniscono anche caratteri alfabetici (ASCII, Baudot, ecc.):

Tipo di codice	Particolarità	Vantaggi rispetto al binario puro
AIKEN	Richiede 4 bit per ogni cifra decimale, complementando i singoli bits si ottiene il complemento a 9 dell'intera cifra	Facilita le operazioni di sottrazione
Gray	La cifra può essere composta partendo da un numero binario puro e seguendo un facile algoritmo	Due numeri successivi possono variare di un solo bit (elimina potenziali errori)
Eccesso-3	Usa 4 bit per ogni singola cifra decimale	Mancano le configurazioni "0000" e "1111" in modo da evitare ambiguità sulla mancanza di segnale
Eccesso-3 riflesso	Usa 4 bit per ogni singola cifra decimale	Combina i vantaggi del codice Eccesso-3 con quelli del codice Gray
2 su 5	Usa 5 bit per ogni singola cifra	Ogni numero ha sempre due "1" con evidenti facilitazioni nella correzione degli errori

senso di rotazione, infatti, se il primo impulso ad apparire fosse stato B, il senso di rotazione sarebbe stato opposto. Lo sfasamento di 90° (1/4 di periodo) è necessario per mantenere, in ordine di tempo, i fronti di salita e di discesa dei due segnali più lontani possibile. Veniamo, adesso, al segnale G. Come si può senz'altro intuire dai diagrammi di figura 7, questo segnale non è altro che un indicatore di giro completo effettuato. Infatti, indipendentemente dal tempo occorso, al compimento dei 360° di rotazione verrà emesso l'impulso G che avrà periodo relativo alla velocità angolare istantanea. Il rilevamento di quest'ultimo impulso è necessario, a volte, per fornire lo stato iniziale ai circuiti di conteggio che interfacciano le uscite A e B dell'encoder. Dopo quanto detto appare evidente che la risoluzione di questi trasduttori è direttamente proporzionale al numero d'impulsi forniti in un giro che, nel diagramma di figura 7, è racchiuso fra gli istanti T0 e Tg. Tanto per dare un'idea, sono piuttosto comuni in molte applicazioni encoder incrementali con risoluzioni anche di 4096 impulsi per giro (un impulso ogni 0,08° angolari circa). La parte elettronica associata a questi dispositivi è abbastanza complessa, tale da permettere, in linea di massima, collegamenti diretti a CPU e logiche programmabili senza altri componenti accessori. Spesso, oltre a quanto già visto, vengono fornite linee d'uscita con segnali anche per ½ giro e la possibilità di utilizzare impulsi A e B complementati. Sono state prodotte versioni di encoder incrementali completamente elettromeccaniche, molto simili ai commutatori codificati visti all'inizio, in cui i segnali A e B sono forniti da contatti striscianti. Tali dispositivi, che saranno alla base dell'applicazione presentata nella prossima puntata, sono molto economici e di dimensioni ridotte. In figura 8 possiamo confrontare le dimensioni fisiche dei due tipi.
Naturalmente, per gli encoder elettromeccanici, valgono le stesse considerazioni fatte per i commutatori codificati. Anche in questo caso, mancando totalmente la parte elettronica, sarà necessario provvedere ad un adeguato circuito d'interfaccia. Gli encoder ottici sono particolarmente usati per il conteggio del numero di giri compiuti, per estrapolare la velocità di parti in movimento, per misurare l'accelerazione, ecc. Quelli meccanici trovano largo impiego nell'elettronica "domestica", funzionando come controlli manuali in prodotti a noi familiari come ad esempio videoregistratori, amplificatori audio, ecc.

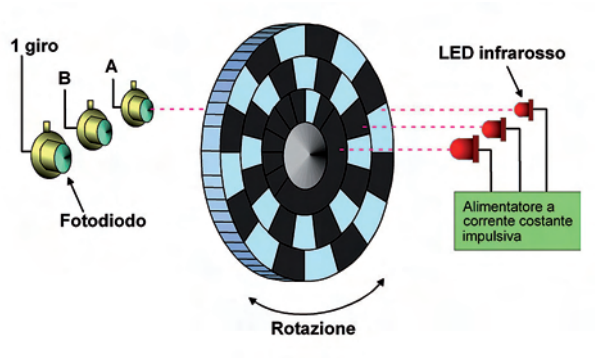


Figura 6
Principio di funzionamento di un encoder ottico incrementale

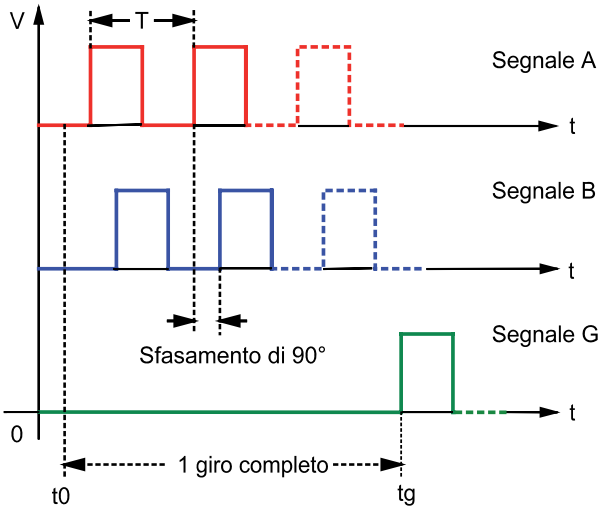


Figura 7
Forme d'onda in uscita da un encoder incrementale

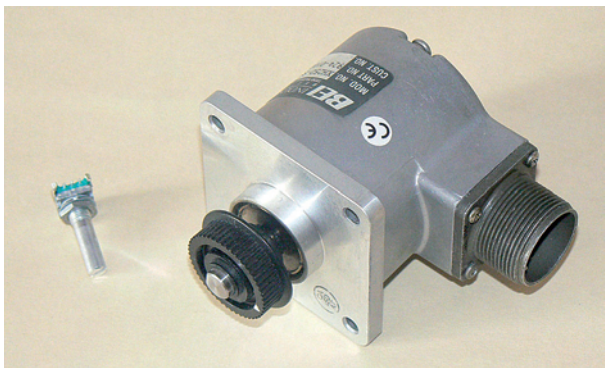


Figura 8
Encoder incrementale ottico (a sinistra) ed elettromeccanico

Conclusioni

Con questa puntata abbiamo terminato la parte teorica, nella prossima, come accennavo, sarà presentata un'applicazione basata su un encoder incrementale reperibile sul mercato a costo irrisorio. Lo scopo di questi articoli, che apparentemente sembrano confinati in un ambito puramente divulgativo, è duplice. Da un lato fornire le basi di un argomento importantissimo e, forse, non reso abbastanza accessibile: i sistemi di controllo. Dall'altro, focalizzare l'attenzione su scelte progettuali che, seppure apparentemente semplici, richiedono comunque un'impostazione di metodo.

More Info Please!
Inserisci il Codice **268044** alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

edutronica.it

il NUOVO punto di riferimento per il MONDO EDUCATIONAL



Sei un docente?

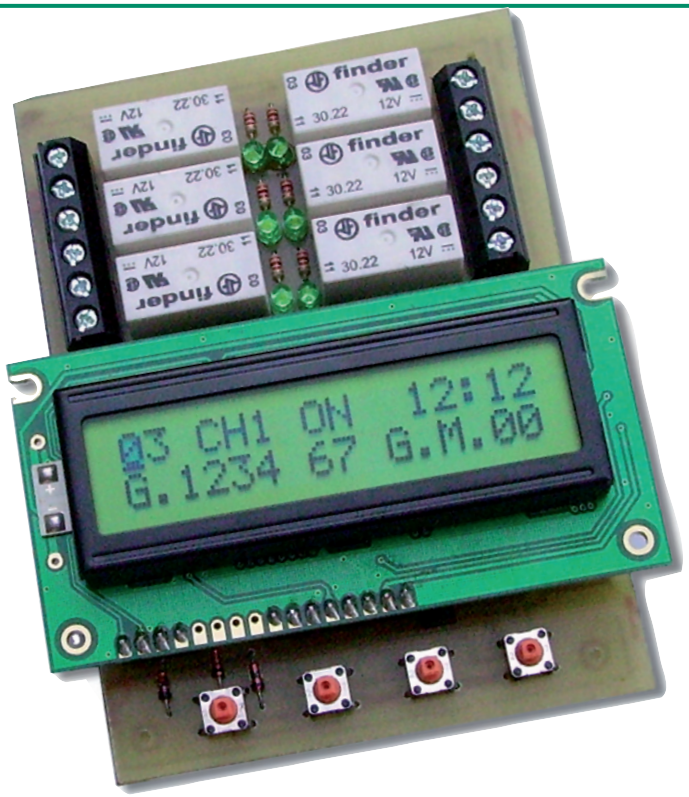
Sei un tutor?

Sei un responsabile di un ufficio tecnico?

La tua risorsa è da oggi online su *www.edutronica.it*

Programmatore orario

Avete mai avuto la necessità di controllare un carico elettrico in maniera automatica senza la necessità della vostra presenza? Con questo circuito ne potrete controllare addirittura 6!



Il progetto in questione è un programmatore orario a sei canali e può trovare un'infinità di applicazioni. Infatti, è possibile controllare un qualsiasi dispositivo elettrico e, attraverso i suoi comodi relè, è possibile controllare anche carichi alimentati a 12 o 24 volt. Per renderlo il più versatile possibile, si può scegliere nella programmazione, oltre al solito orario, anche qualsiasi giorno della settimana nonché il giorno del mese. In questo modo è possibile soddisfare qualsiasi tipo di applicazione.

Schema elettrico

Il cuore del circuito è un PIC 16F628A che, con l'ausilio di un quarzo da 4 MHz e l'appoggio di un display LCD 16x2, diventa un preciso orologio. Verrà utilizzata la periferica TMR2 del PIC come generatore molto preciso di clock partendo da un semplice quarzo da 4 MHz. Lo schema elettrico è riportato in figura 1.

La programmazione viene fatta mediante quattro pulsanti ed un menu guida sul display LCD.

Il circuito integrato ULN2803 è in grado di pilotare i 6 relè delle uscite consentendo allo stesso tempo di ottenere un circuito molto più compatto grazie al minor numero di componenti impiegati. I LED da DL1 a DL6 con le relative resistenze R1..R6 consentono un monitoraggio visivo ed immediato dello stato delle uscite. Tali componenti non sono necessari al funzionamento se non come monitor per cui possono anche essere omessi nel montaggio del circuito.

I relè sono di tipo piuttosto comune. Se nel cassetto disponete dei soliti surplus di origine industriale da 24V, usateli pure avendo l'accortezza di aumentare la tensione di alimentazione a 15-18 V se in alternata, oppure 22-24 V in continua.

Nel completare la descrizione del progetto, da notare il condensatore variabile C1 necessario per aggiustare le piccole variazioni del quarzo in maniera da rendere il clock più preciso possibile. Il trimmer V1 regola semplicemente il contrasto del display e va regolato nella fase di collaudo in modo da leggere le scritte sul display senza difficoltà.

Per aumentare l'affidabilità del dispositivo, è stato aggiunto il circuito composto da D5, D6, R11, C6 e dalla batteria B1. Nel caso di Black-Out la batteria garantisce l'alimentazione al PIC. Se durante la commutazione batteria/rete il PIC dovesse resettarsi, inserite una resistenza di valore compreso tra 100ohm e 1Kohm e aumentate il valore di C6.

Montaggio

Per facilitare il montaggio, è stato disegnato un circuito stampato dove trovano posto tutti componenti. La figura 2 mostra il master del circuito stampato in dimensioni reali, mentre la figura 3 mostra il piano di montaggio dei componenti.



di Silvano Breggion

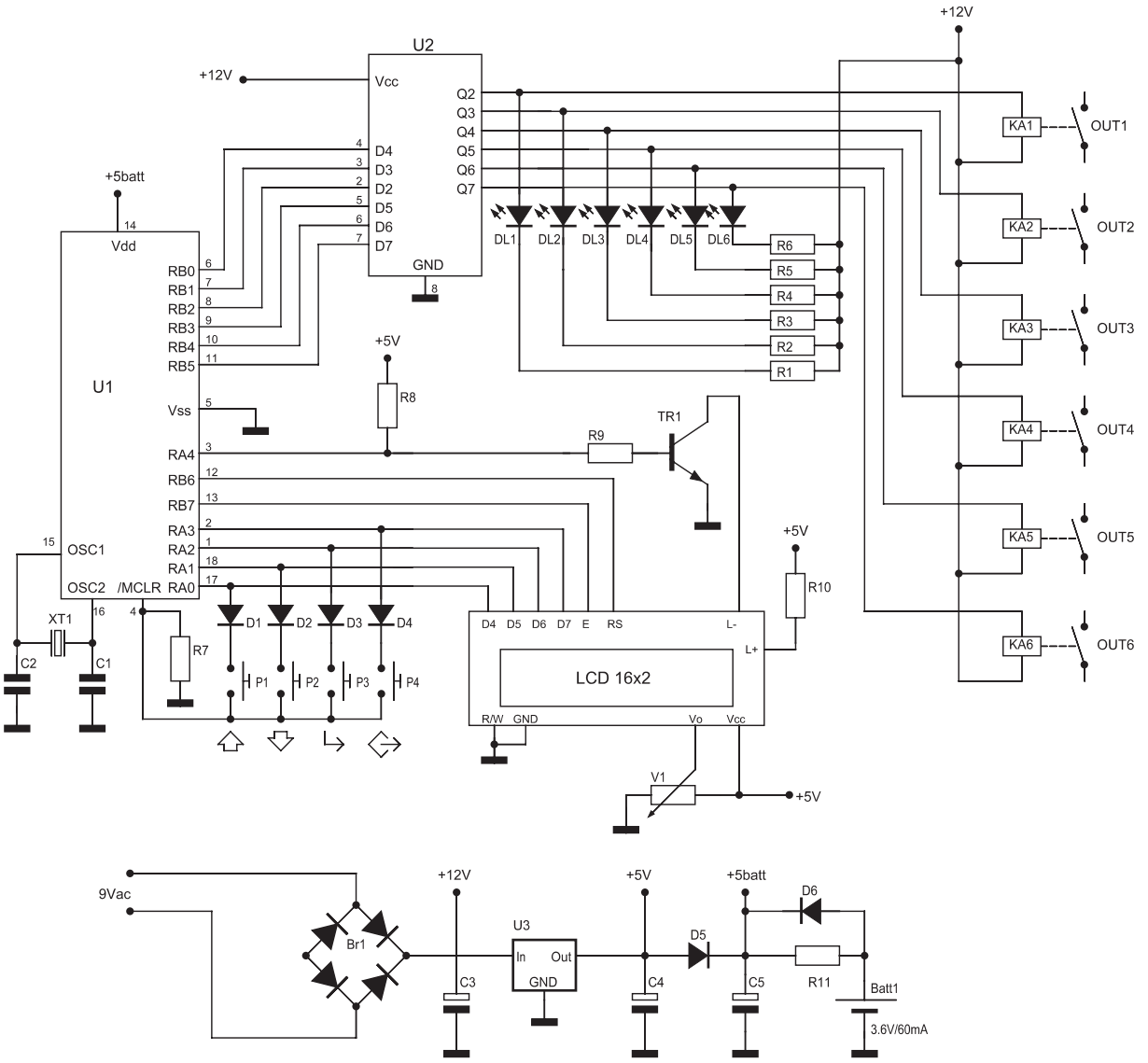


Figura 1
Lo schema elettrico del Timer a 6 canali

ELENCO COMPONENTI

R1-6:	1,5 KΩ 1/8w	C2:	22pF	Tr1:	Bc547b
R7:	10 KΩ 1/8w	C3:	470 µF/35V	U1:	PIC16f628(A)
R8:	4,7 KΩ 1/8w	C4:	47 µF/25V	U2:	ULN2003
R9:	1 KΩ 1/8w	C5:	100 µF/16V	U3:	7805
R10:	22 Ω 1/8w	Xt1:	Quarzo 4MHz	Dis1:	Display 2x16 caratteri (connettore in basso es.: Xiamen LCM1602B)
R11:	180 Ω 1/8w	D1-6:	1n4148		
V1:	Trimmer orizzontale 10 KΩ	Pr1:	Ponte raddrizzatore 1A	K1-6:	Rele' 12v tipo SPDT
C1:	22 pF trimmer capacitivo 6mm	Dl1-6	Led 3mm		

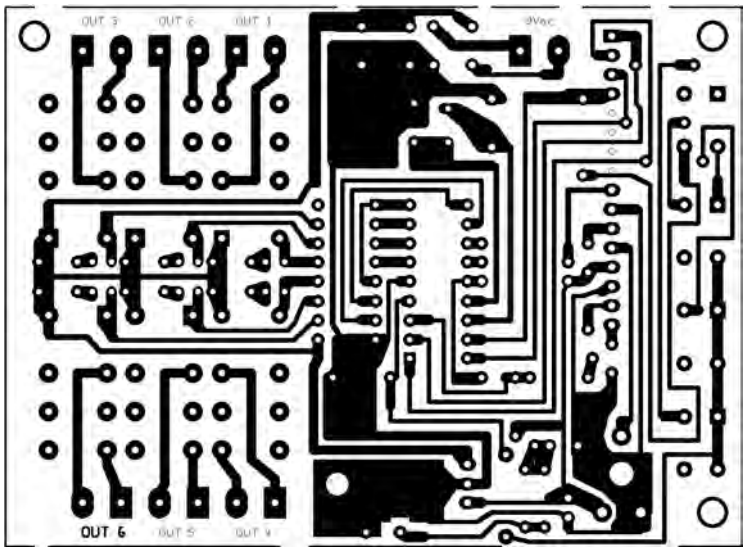


Figura 2
Il master del circuito stampato in dimensioni reali

Come al solito vi consigliamo di montare i componenti radenti al circuito stampato come resistenze, diodi, condensatori, il regolatore di tensione, il quarzo e infine i due zoccoli. Continuate con il montaggio dei relè e le morsettiere, se invece volete semplicemente fare delle prove, attraverso il display sarà possibile visualizzare le uscite che vengono attivate dai tempi impostati. Anche la presenza dei LED in prossimità dei relè, possono aiutare nel collaudo finale.

Per il display è stato previsto il montaggio direttamente da CS, attraverso una semplice coppia di connettori maschio/femmina. A tal fine si è reso necessario il montaggio di alcuni componenti in orizzontale, in particolare U3, C3, C5, C6 e del quarzo XT1. Il circuito stampato non prevede uno spazio per la batteria che va collegata con 2 spezzoni di filo. Ovviamente ciò non impedisce a chiunque di adottare soluzioni diverse, influenzate anche dal tipo di contenitore disponibile.

La figura 4 mostra il dispositivo montato prima dell'inserimento del display.

Collaudo

Prima di inserire i due integrati nei relativi zoccoli, vi consigliamo di alimentare il dispositivo con 12 Vcc o 9Vca circa. Controllate la presenza di 12 Volt tra i piedini 9 e 10 di U2 e 5V tra i piedini 5 e 14 di U1 usando un multimetro.

Se tutto è a posto, dopo avere tolto l'alimentazione, inserite i due integrati nei relativi zoccoli controllando che l'orientamento sia quello corretto. Inserite il display nel connettore, e alimentate il circuito. Il firmware per il microcontrollore è disponibile sul sito www.farelettronica.com.

A questo punto, con il circuito alimentato, sul display dovrebbe apparire brevemente una scritta di default, subito dopo il datario e l'ora nella prima riga quindi lo stato delle sei uscite nella seconda. La forma è la seguente:

Lun 00Gen 00:00
CH:1-2-3-4-5-6-

Se nel vostro dispositivo appare questo, allora potete stare sicuri che il circuito funziona perfettamente. Se il display non visualizza alcunché, regolate il trimmer V1 per il giusto contrasto e poter leggere quanto descritto sopra in maniera chiara.

Programmazione dei tempi

Come già detto l'intera programmazione avviene tramite i quattro pulsanti presenti a bordo del cir-

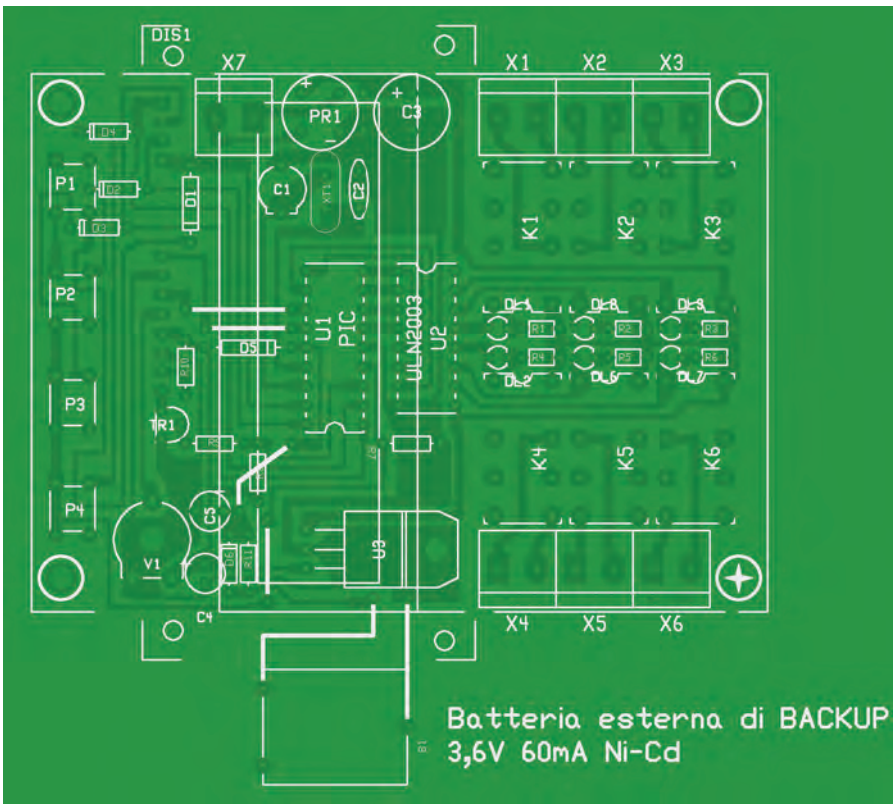


Figura 3
Il piano di montaggio dei componenti

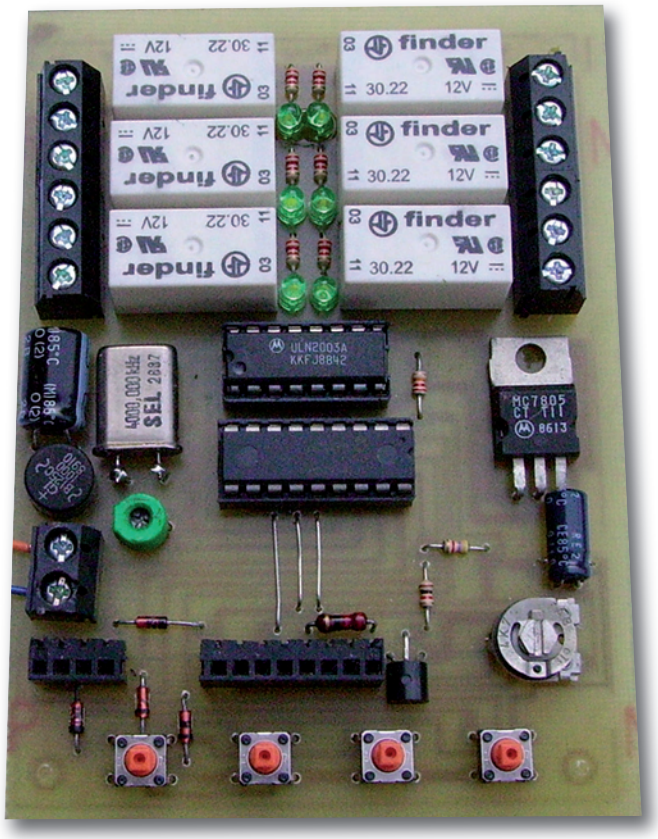


Figura 4
Il prototipo del timer a 6 canali privo di display

cuito stampato. Il primo pulsante da sinistra incrementa i valori (**P1**), il secondo li decrementa (**P2**), il terzo viene utilizzato per uscire dalla programmazione in qualsiasi momento (**P3**) e infine il quarto per cambiare campo (**P4**).
Per entrare nella programmazione è necessario tenere premuto il quarto pulsante per almeno tre secondi. La fase di programmazione viene visualizzata dalla scritta "*Program*" nella prima riga mentre nella seconda è possibile scegliere tra l'aggiustaggio dell'ora e della data e la programmazione degli orari per controllare le uscite.
→Ora/Dat Canali
Il passaggio dall'una o l'altra funzione, avviene tramite il pulsante di "INCREMENTA" (**P1**) e visualizzata da una freccia che si sposta da un campo all'altro.
Ora/Dat→Canali
Scelta la funzione, che ovviamente al primo avvio sarà la regolazione dell'ora e della data, va confermata premendo il tasto di cambio campo (**P4**).
Ammesso che vogliate regolare l'orario e la data, la pagina di regolazione si presenta con il datario completo nella prima riga, e la scritta "program ora/data" nella seconda.
Lun 00 Gen 00:00
Program Ora/Data
Il pulsante "INCREMENTA" (**P1**) cambia l'informazione o il valore nel campo selezionato dal pulsante

comune di **scandiano**

MOSTRA ELETTRONICA
SCANDIANO 2007



27/28 OTTOBRE 2007
SCANDIANO (REGGIO EMILIA)
CENTRO FIERISTICO

4ª FIERA REGIONALE DELL'ELETTRONICA

Orari:
sabato
9,00-18,30
domenica
9,00-18,00

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

con il patrocinio di A.R.I. sez. Scandiano

Codice MIP 268055

INFO:
0522/857436
www.fierascandiano.it

Biglietti:
Intero euro 7,00
Ridotto euro 4,00
Ragazzi fino a 12 anni gratuito

"cambia campo" e visualizzata nel display da un cursore lampeggiante. Per prima cosa va aggiornato il giorno della settimana semplicemente premendo il primo pulsante tante volte fino a visualizzare il giorno preciso. La pressione del **P4** sposta il cursore lampeggiante sulla data del mese. Proseguite l'operazione con la regolazione del mese, delle ore, dei minuti. Ad ogni pressione del **P4** è possibile ricominciare dal giorno della settimana e così in un loop infinito. Per uscire da questa fase di programmazione, è necessario prendere **P3** e ritornare nella pagina di visualizzazione principale.
Per impostare gli orari della gestione delle uscite, bisogna ritornare nella fase di programmazione premendo **P4** per tre secondi, con il **P1** spostare la freccia nel "campo Canali" e premere **P4**. La pagina di programmazione si presenta più complicata rispetto a quella della regolazione dell'ora, vediamo in dettagli di che si tratta.

07 CH6 OFF 21:15
G.1234567 G.M.00

La prima informazione riguarda il numero di stringa presente in memoria e parte sempre dall'ultima. Ci spieghiamo meglio, se in memoria sono presenti i sei stringhe ovvero sei orari, nella fase di programmazione verrà visualizzata la sesta stringa. Ovviamente con i pulsanti **P1** e **P2** è possibile inserire la nuova stringa oppure visualizzare e/o modificare le stringhe in memoria. La logica è la stessa vista per la programmazione dell'ora, **P4** cambia il campo e **P3** permette l'uscita in qualsiasi momento dalla fase di programmazione.

Proseguiamo la descrizione con il secondo campo che indica il canale scelto cioè l'uscita, **CH1** indica l'OUT 1, mentre **CH6** l'OUT 6. La terza informazione riguarda il tipo di uscita, **ON** indica l'eccitazione della relè, mentre **OFF** indica lo spegnimento. Segue l'ora e i minuti in cui il carico verrà acceso o spento, mentre nella seconda riga troviamo i giorni della settimana in cui tale orario verrà rispettato. Infatti dobbiamo programmare sette campi uno per ogni giorno della settimana, 1 indica il lunedì, mentre il 7 indica la domenica. Con il pulsante "INCREMENTA" (**P1**) possiamo cancellare il giorno oppure visualizzarlo.

Se programmiamo l'orario in questo modo '1234_67' l'orario resta valido per tutti i giorni della settimana ma ignorato il venerdì. Altro esempio, '12345__' in questo caso l'orario viene rispettato da lunedì a venerdì, ignorato il sabato e la domenica. Come vedete non c'è nulla di complicato è sufficiente un po' di pratica perché la programmazione diventi veramente cosa semplice. L'ultimo campo riguarda il giorno del mese. Se impostiamo "**00**" non vogliamo tale filtro, cioè tutti i giorni del mese rendono valido l'orario. Se invece desideriamo, per esempio, attivare un carico a mezzogiorno del giorno **10** di ogni mese e staccarlo alle cinque di sera del giorno 15 di ogni mese, è sufficiente impostare il giorno 10 nella stringa di "**ON**" e il giorno "**15**" in quella di "**OFF**".

07 CH1 ON 12:00
G.1234567 G.M.10
08 CH1 OFF 17:00
G.1234567 G.M.15

Informazione molto importante: perché la stringa venga accettata e scritta nell'area di memoria EEPROM, è necessario cambiare il campo di tutta la stringa e ritornare al numero di stringa. Infatti il programma salva la stringa soltanto quando il loop viene completato.
Pertanto se uscite dalla fase di programmazione a metà stringa questa viene persa.
In ogni caso è possibile visualizzare il contenuto in memoria senza necessariamente apportare modifiche.
Ogni stringa è composta da quattro Byte quindi è possibile memorizzare fino al 31 stringhe nei 128 BYTE dell'EEPROM del PIC16F628A, mentre utilizzando un PIC16F648A, che di BYTE ne possiede 256 e modificando il programma, possono essere salvate fino a 63 stringhe (cioè orari). Il valore da cambiare si trova nel file "*program.inc*" all'etichetta "`_Program_canali`" trovate la parte di programma sotto descritta

```
        …
movf    TOP,w
sublw   .31 ; Max 31 orari programmabili
btfss   carry
clrf    TOP
        …
```

sostituite "**.31**" con "**.63**". Non dimenticate il punto prima del numero, indica il formato decimale nel valore.
Questo solo nel caso utilizzate il PIC16F648A che possiede un'area di memoria EEPROM doppia rispetto al PIC16F628A.
Come potete notare è stato utilizzato il modello con la 'A' finale, ma penso che anche la variante senza 'A' possa funzionare ugualmente bene.

More Info Please!
Inserisci il Codice **268052** alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

PUBLIC UTILITIES INTERNATIONAL FORUM 2007

MODERNIZZAZIONE DELLE RETI COME STRATEGIA PER L'EFFICIENZA E IL RISPARMIO

11-12 ottobre Cernobbio (Como)

10ª edizione

Villa Erba
Largo Luchino Visconti

TELECONTROLLO
RETI ACQUA, GAS ED ELETTRICHE 2007

Organizzato da

FIERA MILANO TECH

Via Gattamelata, 34
20149 - Milano
tel. +39 02 3264205
fax +39 02 3264284
segreteria@fieramilanotech.it
www.fieramilanotech.it

Promosso da

ANIE FEDERAZIONE
CONFINDUSTRIA
ASSOAUTOMAZIONE
ASSOCIAZIONE ITALIANA AUTOMAZIONE E MISURA

Via Gattamelata, 34
20149 - Milano
www.anie.it
www.assoautomazione.it

Per sponsorizzazioni e iscrizioni: Tel. 02 3264.205 e-mail: mara.borsani@fieramilanotech.it

Codice MIP 268057

Prima parte: *Luglio/Agosto 2007*
Introduzione al sistema

Seconda parte: *Settembre 2007*
La programmazione in NQC (I)

Terza parte: *Ottobre 2007*
La programmazione in NQC (II)

Quarta parte: *Novembre 2007*
La programmazione in NQC (III)

I sistemi robotici Lego Mindstorms

Continuiamo il discorso iniziato nello scorso numero approfondendo la trattazione del linguaggio di programmazione NQC, che ci consentirà di sviluppare delle interessanti applicazioni per il nostro sistema robotico LEGO MINDSTORMS.

Operatori

Dopo aver introdotto il concetto di identificatore, variabile e costante, vediamo come è possibile costruire espressioni mediante gli operatori che il linguaggio NQC ci mette a disposizione.
Analogamente all'ANSI C, in NQC sono definite quattro classi di operatori: aritmetici, logici, relazionali e sui bit.
Gli operatori aritmetici sono riportati in tabella 1; oltre alle quattro operazioni fondamentali dell'algebra esistono l'operatore modulo %, che fornisce il resto di una divisione fra numeri interi e gli operatori di incremento ++ e decremento --, che rispettivamente sommano e sottraggono uno all'operando.
Gli operatori relazionali hanno la funzione di confrontare due valori (o espressioni). Il risultato di un'operazione relazionale è una condizione logica (vero o falso). La tabella 2 riassume gli operatori relazionali: maggiore (>), maggiore o uguale (>=), minore (<), minore o uguale (<=), uguale (==), diverso (!=).
Gli operatori logici permettono di confrontare condizioni logiche, restituendo un'ulteriore condizione logica. La tabella 3 riporta gli operatori logici: AND, OR, NOT. L'operatore AND (&&) restituisce la condizione vero soltanto quando entrambi i suoi operandi sono vero, l'operatore OR (||) restituisce la condizione vero quando almeno uno dei due suoi operandi è vero. L'operatore NOT (!) è unario, deve precedere l'operando e restituisce vero se il suo operando è falso, falso se il suo operando è vero.

Operatore	Azione
-	Sottrazione e meno unario
+	Addizione
*	Moltiplicazione
/	Divisione
%	Resto della divisione intera
- -	Decremento
+ +	Incremento

Tabella 1
Operatori aritmetici

Operatore	Azione
>	Maggiore
>=	Maggiore o uguale
<	Minore
<=	Minore o uguale
==	Uguale
!=	Diverso

Tabella 2
Operatori relazionali

Operatore	Azione
&&	AND
||	OR
!	NOT

Tabella 3
Operatori logici

Operatore	Azione
&	AND
|	OR
^	XOR
~	NOT
>>	Scorrimento a destra
<<	Scorrimento a sinistra

Tabella 4
Operatori sui bit

La programmazione in NQC (II parte)



di Fabio Riscica

Gli operatori sui bit consentono di manipolare i singoli bit che compongono la word. La tabella 4 riassume gli operatori sui bit che comprendono lo scorrimento (shift), che deve essere seguito dal numero di posizioni di cui si vuole effettuare lo shift e gli operatori AND (&) OR (|) NOT (~), a cui si affianca ora l'operatore XOR (^), che restituisce la condizione vero soltanto se i suoi due operandi sono diversi. Si osservi però che stavolta questi operatori lavorano a livello dei singoli bit, quindi se per esempio scrivo l'espressione 0x33 & 0xCC, il risultato sarà calcolato mediante l'AND bit a bit dei numeri binari 00110011 e 11001100, ottenendo il valore 00000000, che corrisponde al numero esadecimale 0x00.
Se scrivo invece l'espressione 0x17 >> 3, il risultato sarà calcolato mediante lo scorrimento di tre posizioni a destra del numero binario 00010111, ottenendo 00000010 (0x02). Si osservi che i bit che escono per scorrimento vengono persi, quelli che entrano vengono invece posti a 0.



Espressioni

Tutti gli operatori esaminati possono essere combinati con costanti e variabili per costruire espressioni. Nella costruzione delle espressioni bisogna tenere conto della priorità degli operatori, che comunque può essere modificata facendo uso delle parentesi tonde. La tabella 5 riassume la priorità degli operatori studiati.

PRIORITÀ CRESCENTE

(	)			
!	~	++	- -	- (unario)
*	/	%		
+	-			
<<	>>			
<	<=	>	>=	
==	!=			
&				
^				
|				
&&				
||				
=				

Tabella 5
Priorità degli operatori

Le strutture di controllo

Esaminiamo ora le strutture che consentono di condizionare l'esecuzione delle istruzioni di programma in base al valore di determinate condizioni logiche. Facendo riferimento alla teoria dei diagrammi di flusso (tecnica che consente di rappresentare in maniera grafica la struttura di un algoritmo, in modo da agevolare il successivo processo di codifica nel linguaggio di programmazione scelto), NQC mette a disposizione le strutture di selezione semplice, di selezione multipla, l'iterazione a controllo iniziale, finale ed enumerativa.

Selezione semplice

La forma generale di tale struttura è la seguente:

```
if (condizione logica)
        istruzione;
else istruzione;
```

dove l'istruzione consiste in un'istruzione semplice o composta (cioè una sequenza di istruzioni rac-



chiuse tra parentesi graffe). La sezione else è opzionale. Se la condizione logica assume valore vero viene eseguita l'istruzione immediatamente successiva, altrimenti quella che corrisponde all'else (se presente). La struttura condizionale può essere ripetutamente annidata utilizzando come istruzioni ulteriori strutture if .. else.

Selezione multipla

Tale struttura è un'estensione della precedente ed ha la forma:

```
switch (variabile)
{
case costante1 :
  sequenza di istruzioni
break;
case costante2 :
  sequenza di istruzioni
break;
case costante3 :
  sequenza di istruzioni
break;
…
default :
  sequenza di istruzioni
}
```

In tale costrutto il valore di una variabile viene confrontato con una lista di costanti; quando si verifica una corrispondenza viene eseguita la sequenza di istruzioni corrispondenti alla costante individuata. La parte default è opzionale e viene eseguita, se presente, nel caso in cui non si verifichi alcuna corrispondenza.
Si osservi che il costrutto switch consente soltanto di verificare l'uguaglianza di una variabile con un valore costante a differenza del costrutto if .. else, che consente di verificare condizioni di ogni tipo.

Iterazione a controllo iniziale e finale

La struttura while viene classificata come iterazione a controllo iniziale; l'istruzione (semplice o composta) viene eseguita fintanto che la condizione logica è vera:

```
while (condizione logica)
        istruzione;
```

La struttura do .. while viene definita come iterazione a controllo finale: anche stavolta l'istruzione viene eseguita fintanto che la condizione logica è vera, ma la verifica è effettuata dopo l'esecuzione dell'istruzione. Ciò implica, a differenza del caso precedente, che l'istruzione viene eseguita sempre almeno una volta.

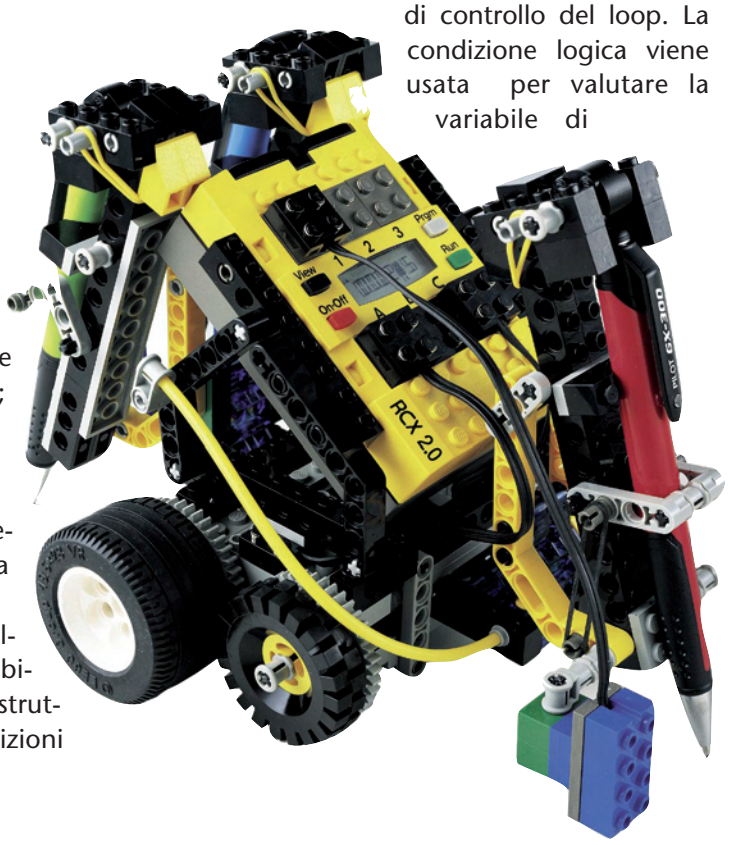
```
do
    {
      istruzione
    }
while (condizione logica);
```

Iterazione enumerativa

Il loop for permette di implementare l'iterazione enumerativa, anche se in realtà la versione messa a disposizione dall'NQC è più potente e flessibile.
La sua forma generale è:

```
for(inizializzazione;condizione;variazione)
      istruzione;
```

L'inizializzazione è un'istruzione di assegnamento con cui si determina il valore iniziale della variabile di controllo del loop. La condizione logica viene usata per valutare la variabile di



controllo in modo da determinare il momento di uscita dal loop, che avviene quando la condizione logica è falsa. La variazione definisce il modo in cui la variabile di controllo cambia il suo valore ad ogni ripetizione del loop (vengono utilizzate in genere istruzioni di incremento o decremento).

Utilizziamo le strutture di controllo in un programma di esempio

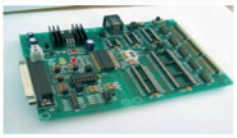
Adoperiamo adesso il Bricx Command Center presentato nello scorso numero per scrivere il seguente programma di esempio:

```
int move_time, turn_time, total_time;

task main()
{
total_time = 0;
do
 {
  move_time = Random(100);
  turn_time = Random(100);
  OnFwd(OUT_A+OUT_C);
  Wait(move_time);
  OnRev(OUT_C);
  Wait(turn_time);
  total_time = total_time + move_time;
  total_time = total_time + turn_time;
 }
while (total_time < 2000);
Off(OUT_A+OUT_C);
}
```

La funzione Random consente di generare un numero casuale compreso tra 0 ed il suo argomento (in questo caso fra 0 e 100). Questo programma fa avanzare il robot per un tempo casuale, quindi gli fa eseguire una curva ancora per un tempo casuale. La struttura iterativa fa ripetere queste due operazioni fintanto che la variabile total_time non è superiore al valore 2000, corrispondente a 20 secondi. Il nostro robot andrà a passeggio per 20 secondi e quindi si fermerà.
Una volta scritto il programma, bisogna compilarlo utilizzando l'apposito comando di menù ed inviarlo al robot mediante il collegamento ad infrarossi. Nel caso in cui fossero presenti degli errori, questi verranno segnalati nella parte inferiore della finestra di programmazione. Una volta che il download è completato, agendo sul pulsante Run del modulo RCX, il programma passerà in esecuzione ed il robot eseguirà la sequenza di movimenti desiderata.

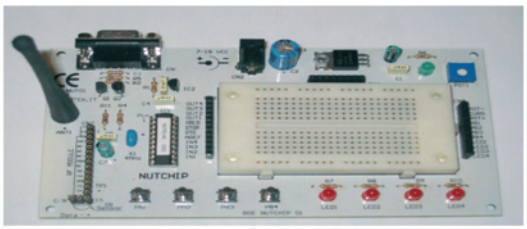
Con Artek hai l'elettronica a portata di un click.
Esplora il nostro sito, ogni mese scoprirai le novità dell'Elettronica, il mondo dei Microcontrollori, nuovi sensori e strumenti per progetti di Robotica.
Inoltre strumenti di misura digitali professionali interfacciati al pc per il laboratorio



Artek ti offre un modulo per programmare i PIC Microchip con funzioni di debug e test a soli 62 Euro IVA compresa.
Visita il nostro sito per sapere di più sul C-Project C-170.



Costruisci un Robot con il BASIC Stamp
il microcontrollore più famoso e diffuso fra gli appassionati di Robotica per la sua semplicità d'uso e la vasta gamma di accessori



BoeNut 01
stazione di lavoro completa per **Nutchip**



Strumenti digitali di misura su porta USB
- due canali
- 12 bit di risoluzione
- ingresso fino a 100 MHZ



Puoi avere questa mini-telecamera **a colori,** completa di **ricevitore** e **microfono** ad un prezzo che non ha eguali !

La nostra vetrina è on-line all'indirizzo www.artek.it : puoi controllare le caratteristiche, i prezzi e ordinare da subito ciò che ti occorre. Puoi contattarci con una e-mail a diramm@artek.it inviando un fax allo 0542 688405 oppure chiamando i nostri uffici allo 0542 643192
dalle 9 alle 13:30 e dalle 14:30 alle 18 dal Lunedì al Venerdì

Codice MIP **268061**

NQC API – Gestione dei sensori	
SetSensor(sensore,configurazione)	Imposta il tipo ed il modo di funzionamento del sensore specificato. Esempio: SetSensor(SENSOR_1,SENSOR_TOUCH);
SetSensorType(sensore,tipo)	Imposta il tipo del sensore specificato. Esempio: SetSensorType(SENSOR_1,SENSOR_TYPE_TOUCH);
SetSensorMode(sensore,modo)	Imposta il modo di funzionamento del sensore specificato. Esempio: SetSensorMode(SENSOR_1,SENSOR_MODE_RAW);
ClearSensor(sensore)	Azzera il valore del sensore (interessa i soli sensori configurati per misurare valori cumulativi come rotazioni o conteggi). Esempio: ClearSensor(SENSOR_1);
SensorValue(n)	Restituisce il valore acquisito dal sensore (0,1 o 2). Esempio: x = SensorValue(0);
SensorType(n)	Restituisce il tipo configurato per il sensore (0,1 o 2). Esempio: x = SensorType(0);
SensorMode(n)	Restituisce il modo configurato per il sensore (0,1 o 2). Esempio: x = SensorMode(0);
SensorValueBool(n)	Restituisce il valore booleano acquisito dal sensore (0,1 o 2). Esempio: x = SensorValueBool(0);
SensorValueRaw(n)	Restituisce il valore raw acquisito dal sensore (0,1 o 2). Esempio: x = SensorValueRaw(0);

Tabella 6
I comandi del linguaggio NQC che possono essere utilizzati per la gestione degli ingressi

NQC API: gestione degli ingressi e dei sensori ad essi collegati

Gli ingressi collegati all'RCX possono essere identificati mediante le costanti SENSOR_1, SENSOR_2 e SENSOR_3. Questi identificatori possono essere utilizzati in tutte le funzioni che richiedono sensori come argomenti, ma non possono essere utilizzati per far riferimento al valore restituito (ad esempio la funzione SensorValue ha come argomento il numero del sensore, che deve essere 0, 1 o 2).
Esistono differenti tipi di sensori prodotti da Lego ed è necessario configurare le corrispondenti porte in maniera coerente prima di essere utilizzate, mediante la funzione SetSensorType. I tipi di sensore supportati sono SENSOR_TYPE_NONE (generico sensore passivo), SENSOR_TYPE_TOUCH (sensore di tocco), SENSOR_TYPE_TEMPERATURE (sensore di temperatura), SENSOR_TYPE_LIGHT (sensore di luminosità) e SENSOR_TYPE_ROTATION (sensore di rotazione).
La configurazione dei sensori può avvenire in differenti modi, così da determinare come il valore restituito dal sensore debba essere processato, mediante la funzione SetSensorMode.
I possibili modi sono SENSOR_MODE_RAW (valore raw compreso tra 0 e 1023), SENSOR_MODE_BOOL (valore booleano), SENSOR_MODE_EDGE (conteggio di transizioni booleane), SENSOR_MODE_PULSE (conteggio di periodi booleani), SENSOR_MODE_PERCENT (valore compreso tra 0 e 100), SENSOR_MODE_FARHENHEIT (gradi °F), SENSOR_MODE_CELSIUS (gradi °C), SENSOR_MODE_ROTATION (16 impulsi per rotazione).
È comunque possibile configurare simultaneamente il tipo ed il modo di funzionamento del sensore mediante la funzione SetSensor, che può direttamente avere per argomento le costanti:
SENSOR_TOUCH, SENSOR_LIGHT, SENSOR_ROTATION, SENSOR_CELSIUS, SENSOR_FARHENHEIT, SENSOR_PULSE, SENSOR_EDGE.

Nella tabella 6 vengono riportati tutti comandi del linguaggio NQC che possono essere utilizzati per la gestione degli ingressi.

Conclusioni

Abbiamo quindi ulteriormente esaminato le peculiarità del linguaggio NQC, appositamente scritto a modello dell'ANSI C per la programmazione dei robot Lego Mindstorms. Nella prossima puntata procederemo presentando alcuni esempi pratici ed ultimando la presentazione di questo linguaggio, che comprende anche funzioni per la generazione di musica e di gestione dei tasks.

More Info Please!
Inserisci il Codice **268058** alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

VI Edizione

TELEMOBILITY

Telematics and Infomobility Forum

CONFERENCE & EXPO

POSITION YOUR BUSINESS

L'UNICO EVENTO EUROPEO DEDICATO A NAVIGAZIONE SATELLITARE E INFOMOBILITÀ

con il patrocinio:

COMMISSIONE EUROPEA
RAPPRESENTANZA A MILANO

RegioneLombardia

Provincia di Milano

14-15/11/2007

Autodromo di Monza

MONZA

LE AREE TEMATICHE

Moduli Wireless GPRS/UMTS e Terminali Veicolari
Ricevitori GPS OEM e Personal Navigation
GPS Software e Cartografia Digitale
Servizi di localizzazione e Infomobilità
Localizzazione Indoor
Sistemi telematici per la Pubblica Sicurezza ed Emergenza
RF-ID e Tracciamento Beni / Persone
Open Source GIS
Location Based Contents and POIs
Car Sensors / Remote Car Maintenance
GPS per sport & fun

Per info:
tel. +39 02 48517925
info@telemobilityforum.com

Organizzato da:

pulvermedia™

wireless
Where events become business






www.telemobilityforum.com

INGRESSO GRATUITO RISERVATO AGLI OPERATORI DEL SETTORE

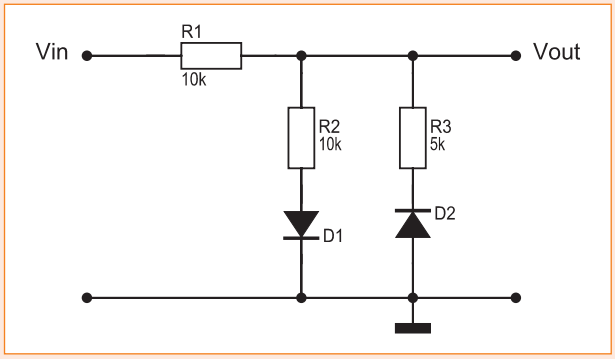
Codice MIP **268063**

Elettro

RISPONDI&VINCI!

FACILE

Determinare la tensione Vout quando Vin=+15V e quando Vin=-20V considerando i due diodi come ideali.

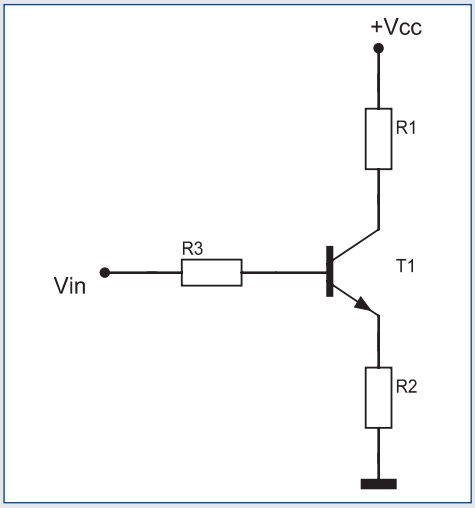


Se rispondi correttamente potrai vincere il righello in alluminio con calcolatrice a 8 cifre e doppia alimentazione.



DIFFICILE

Del transistore T1 sono noti tutti i parametri h. Determinare l'espressione analitica dell'impedenza di ingresso dinamica dell'amplificatore di figura.





Per i più bravi in palio il bellissimo gilet reporter di Fare Elettronica.

Le risposte ai quiz "Facile" e "Difficile" vanno inviate esclusivamente compilando il modulo su www.farelettronica.com/eq specificando la parola chiave "Edison". Le risposte ed i vincitori (previa autorizzazione) sono pubblicati alla pagina www.farelettronica.com/eq a partire dal 15 del mese successivo alla pubblicazione sulla rivista. A tutti i partecipanti verrà assegnato un **buono sconto del 10%** (validità 3 mesi dalla data di assegnazione) utilizzabile per un prossimo acquisto su www.ieshop.it

Quiz n. 268

Spazio in collaborazione con Otello
www.parole.tv

ELETTRO CRUCIVERBA

1	2	■	3	4		5	6		7
8					■	9		■	
10		■	11		12		■	13	
14			■	15			16		
17		■	18						■
■	19	20					■	21	22
23						■	24		
25		■	26			27	■	28	
	■	29		■	30				
31							■	32	

ORIZZONTALI

1. Effetto iniziale
3. Un display LCD non alfanumerico
8. Prodotto delle api
9. La Capitale in sigla
10. Pubbliche Relazioni
11. Divinità greca della salute
13. Vocali di palo
14. Profonda in poesia
15. Tirare su a forza
17. Orson regista e attore (iniz.)
18. La mente umana
19. È ricoperta da piccole piume
21. Napoli
23. Badessa di un convento
24. La piazzola sul pcb
25. Unione Europea
26. Viene dopo l'ottavo
28. Rieti
29. Undici romano
30. Pronostico sull'ippica
31. Riporta le norme di buona educazione
32. Le prime vocali

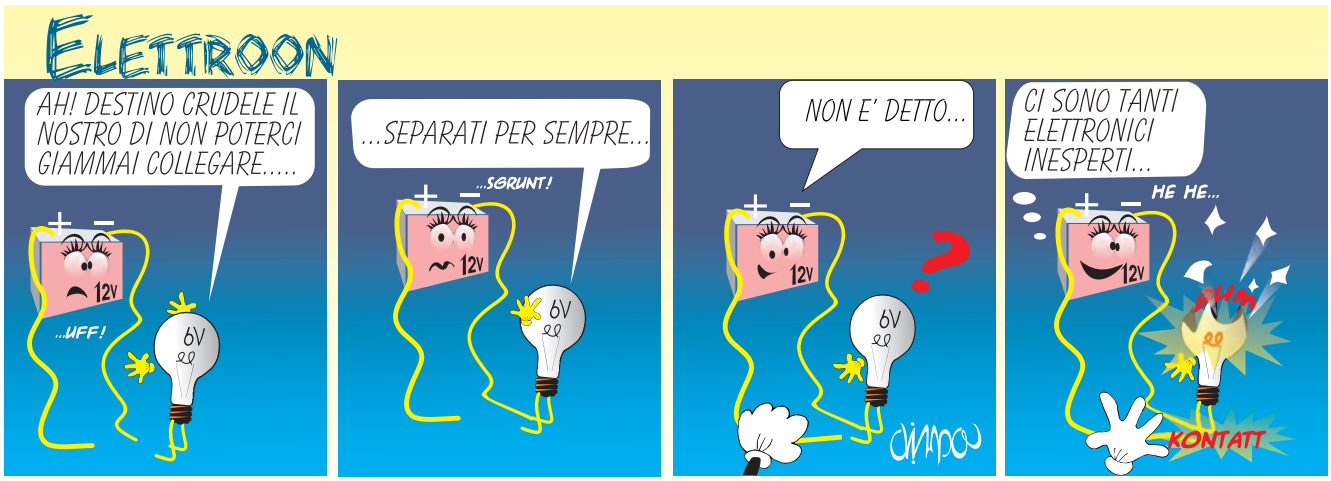
VERTICALI

1. Sacrilego, dissacrante
2. Il programma per il microcontrollore
3. Articolo per uomini
4. Può essere costituito da flip-flop d-latch
5. Ramoscello
6. Imperia
7. Strumento musicale a fiato
12. Incerto, titubante
13. Pietra calcarea da costruzione
16. Esclamazione di sorpresa
18. Grosso fiore simile alla rosa
20. Livorno
22. Pancia, grasso in vita
23. Manuel autore de "Il bacio della donna ragno"
27. Sono tre in mille
29. Taglia molto larga

ELETTRO REBUS

FRASE: (5, 2, 3)





Settima parte: *Settembre 2007*
I Relays e la gestione dei dati

Ottava parte: *Ottobre 2007*
Utilizzare il modulo Led a 7 segmenti giganti

Nona parte: *Novembre 2007*
Applicazioni con i display a 7 segmenti

Il PLC per tutti

In questo numero saranno esaminati i moduli dei display a 7 segmenti, collegabili direttamente al Cubloc. Cablaggio e programmazione molto semplici, per un utilizzo alla portata di tutti.

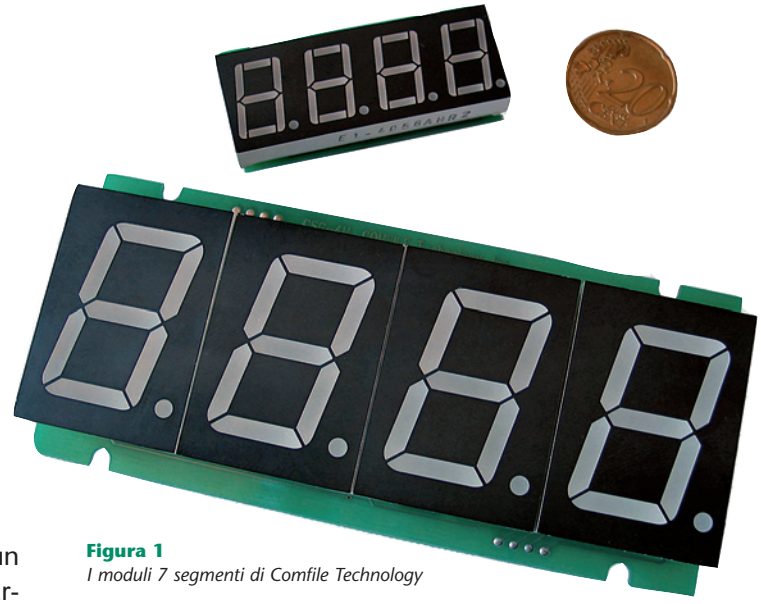


Figura 1
I moduli 7 segmenti di Comfile Technology

La semplicità con cui è possibile interfacciare un modulo a display a 7 segmenti al Cubloc è disarmante! Un facile collegamento elettrico, un codice Basic molto corto e compatto, ed il gioco è fatto.

Utilizzando i normali display a 7 segmenti con un normale microcontrollore (ad esempio il Pic), è necessario predisporre il sistema mediante alcuni accorgimenti, che non sono sempre tra i più semplici. Ecco di solito cosa deve fare un programmatore:

- Adottare un collegamento in parallelo tra i vari display a 7 segmenti;
- Prevedere un algoritmo in grado di accendere un display alla volta (tecnica del multiplexing);
- Implementare una eventuale complicatissima routine di "mantenimento cifre", ma ciò implica un sistema multi-tasking ovvero una gestione di interrupt o polling molto critici;
- Adottare una corrente maggiore per l'alimentazione dei display;
- E molti ancora…

Alcune di queste tecniche sono state ampiamente descritte nel rivista numero 240 di Fare Elettronica. Per il principiante, esse potrebbero costituire un serio problema, a tal punto da farlo desistere dall'idea di realizzare un progetto dotato di queste unità di visualizzazione.

I moduli CSG

Per semplificare al "massimo" la vita, la Comfile Technology ha messo sul mercato molti dispositivi di visualizzazione, utilizzabili in abbinamento al Cubloc. Uno di questi è rappresentato dai moduli della serie CSG. Si tratta di moduli composti da ben quattro display a 7 segmenti, con punto decimale destro, dal funzionamento realmente intelligente. Ogni display è formato da 8 diodi led (7 per i segmenti ed 1 per il punto decimale).
Come c'è da aspettarsi, il cablaggio previsto è composto da "soli" quattro fili, due per l'alimentazione e due per la transizione dei dati. Implementano infatti il protocollo I2C per il transito delle informazioni. Vediamone le caratteristiche salienti, veramente interessanti:

- Pieno supporto dei protocolli CuNet e I2C;
- Naturale collegamento con due fili per i dati e due per l'alimentazione;
- Presenza di 4 cifre;
- Possibilità di collegamento in cascata fino a 4 moduli (per un totale di 16 digit);
- Indirizzamento indipendente per ogni "slave";
- Semplici comandi nativi in Basic per Cubloc;
- Disponibile (almeno per ora) in due formati

Utilizzare il modulo Led a 7 segmenti giganti



di Giovanni Di Maria

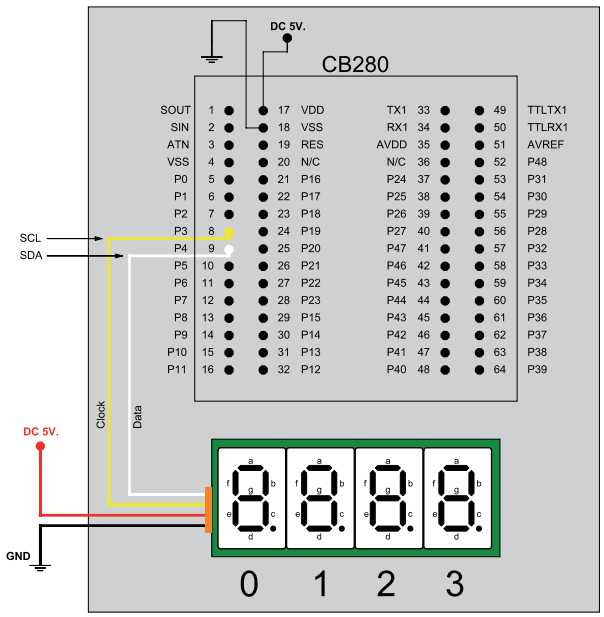


Figura 2
Schema di connessione del modulo CSG al Cubloc, sulla Study Board

diversi: 50mm x 24mm x 17mm (mod. CSG-4S) e 115mm x 50mm x 27mm (mod. CSG-4M utilizzato in questo articolo);
- Luminosità delle cifre molto intensa;
- Eliminazione automatica degli zeri non significativi.

La connessione al Cubloc sulla Study Board

Il modulo del display, come detto prima, necessita solamente di due linee di comunicazione, e precisamente della linea DATA e della linea CLOCK. Esso prevede infatti l'utilizzo del protocollo I2C per la transizione dei dati e delle informazioni. Tali linee possono essere ricavate praticamente da qualunque pin del Cubloc, naturalmente previa la solita connessione di due resistenze di pull-up, per il mantenimento di un livello logico alto, in assenza di segnale.
Come si vede dallo schema di figura 2, il modulo CSG, formato da 4 display a 7 segmenti, è collegato al Cubloc, ed in generale a tutto il circuito, attraverso un bus di 4 linee.
Ogni linea ha la seguente funzione:

- Linea SDA (filo bianco sul modulo CSG) che permette il transito del dato attraverso il protocollo I2C da e verso il Cubloc. Essa fa capo (almeno sulla Study Board) alla porta P4 del Cubloc, pertanto su essa deve essere considerata tale connessione;
- Linea SCL (filo giallo sul modulo CSG) che trasporta il segnale di clock per il protocollo I2C. Essa fa capo (almeno sulla Study Board) alla porta P3 del Cubloc, pertanto su essa deve essere considerata tale connessione;
- Linea VCC (filo rosso sul modulo CSG) che rappresenta l'alimentazione del modulo, a 5 Volt rispetto massa;
- Linea GND (filo nero sul modulo CSG) che rappresenta la massa del modulo, in comune con la massa del Cubloc e quella dell'intera alimentazione.

Naturalmente è possibile collegare il display anche alle altre porte presenti sulla Study Board, basta semplicemente specificarlo in sede di programmazione. Il numero progressivo sotto ogni display (0, 1, 2, 3) indica il numero d'ordine dello stesso. Nei prossimi paragrafi approfondiremo la sua funzione.

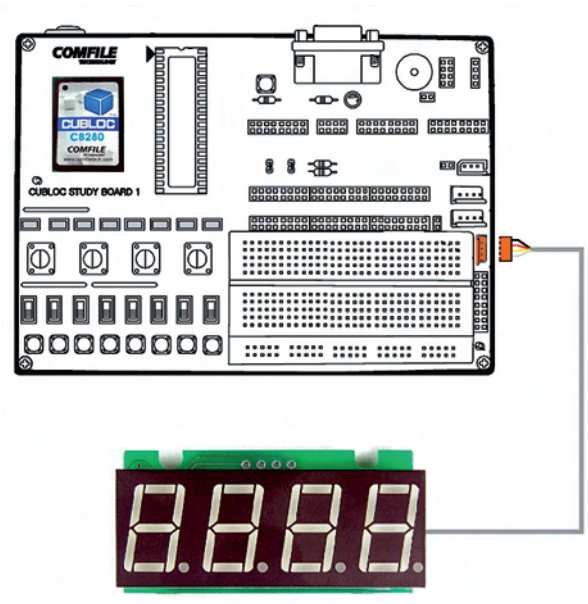


Figura 3
La connessione fisica del modulo CSG sulla Study Board

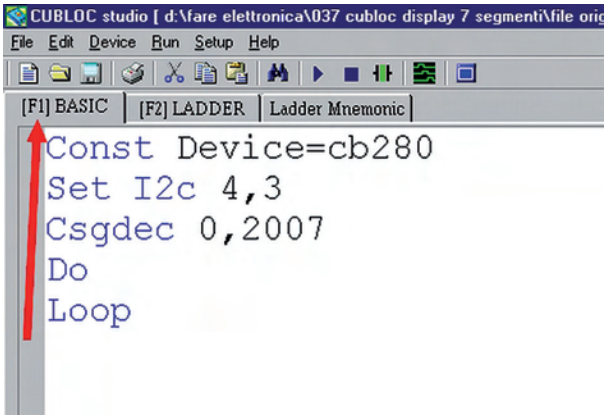


Figura 4
La videata del programma che visualizza il numero 2007



Figura 5
Il display visualizza il numero 2007

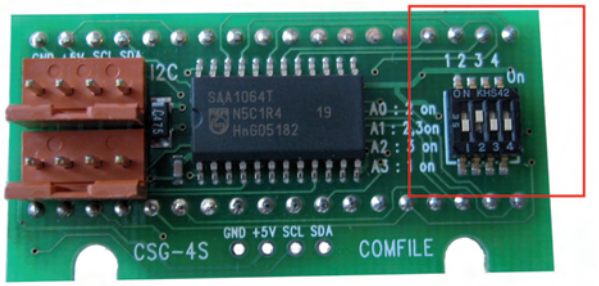


Figura 6
I dip switch per la scelta dell'indirizzo slave

Visualizziamo il nostro primo numero

Come al solito, la migliore strada per imparare è quella di provare praticamente. Andiamo quindi a visualizzare sul modulo del display il nostro primo numero, il "2007", l'anno in corso.
Come detto prima non c'è bisogno di implementare una sofisticata *routine* di *multiplex* e nemmeno di preparare una circuiteria ad hoc. Pensa a tutto il modulo intelligente CSG.
Avviamo pertanto il compilatore Cubloc Studio e digitiamo il primo programma nell'editor riservato al Basic.

```
' LISTATO 1
' Visualizza 2007
Const Device=cb280
Set I2c 4,3
Csgdec 0,2007
Do
Loop
```

Come si vede un programma veramente corto, ma dalla sofisticata funzionalità. Esso è composto solamente da cinque *statements*, tre delle quali passive! Esaminiamole tutte:

- **Const Device=cb280** informa il compilatore che si sta utilizzando il modulo CB280 del Cubloc;
- **Set I2C 4,3** imposta le porte di comunicazione per il protocollo I2C, e nella fattispecie configura la porta P4 del Cubloc quale linea DATA e la porta P3 quale linea CLOCK;
- **Csgdec 0,2007** è il comando più importante, quello che "invia" il numero decimale 2007 verso il display, visualizzandolo;
- **Do-Loop** predispone un ciclo infinito nel quale il Cubloc non effettua nessuna funzione.

Dopo la digitazione del listato e sua relativa compilazione, il numero 2007 sarà immediatamente visualizzato sul display. Degna di nota è la intensa brillantezza delle cifre, visibili anche a molti metri di distanza.

Nota: il commento nelle prime due righe di programma è realizzato tramite il carattere "apostrofo" (').

Il funzionamento

Occorre adesso focalizzare un comportamento molto importante. Dopo che il display ha visualizzato il numero, grazie al comando CSGDEC, esso non "congela" assolutamente l'esecuzione del software ma, al contrario, può espletare altri compiti. Tutto ciò si distingue dal comportamento della programmazione dei normali microcontrollori, che obbligano il programma a gestire continuamente la routine di visualizzazione.

Utilizzare più moduli CSG in cascata

Quando si hanno esigenze maggiori, e occorrono tante unità di visualizzazione, il sistema permette

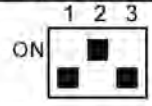
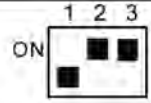
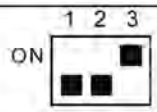
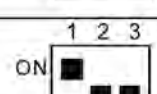
DIP Switch	Slave Address
	0
	1
	2
	3

Figura 7
Le posizioni dei dip switch per la scelta dell'indirizzo slave

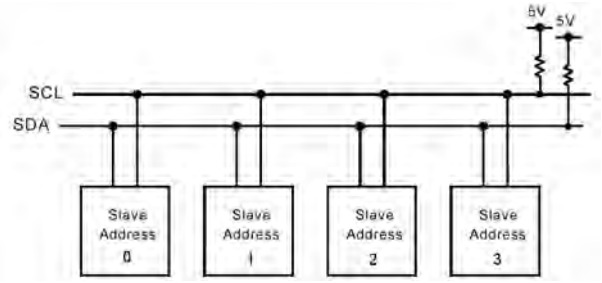


Figura 8
Collegamento in cascata di più moduli CSG

di connettere "in cascata" fino 4 moduli CSG. Ogni modulo si comporta come un'unita slave indipendente e dispone di un proprio indirizzo. La configurazione di questo indirizzo avviene attraverso tre dip switch posti sulla parte posteriore (figura 6).

Come configurare l'indirizzo SLAVE

Ogni modulo può essere configurato con un indirizzo, a scelta tra 0 e 3. Ciò viene fatto utilizzando i primi tre switch presenti sul retro. È possibile scegliere ed utilizzare fino a 4 moduli display in cascata, ognuno indipendente dall'altro. Se si utilizzano almeno due moduli è dunque importante utilizzare codici diversi l'uno dall'altro.

I comandi Basic

Esaminiamo in questo paragrafo l'insieme dei comandi Basic per la gestione completa del display. Si tratta di istruzioni molto semplici e ad "alto livello", in quanto con una sola riga programma vengono realmente svolte tantissime funzioni. Un esempio l'abbiamo visto con il precedente listato.

La lista dei comandi

I comandi supportati dal Cubloc per la gestione dei display CSG sono i seguenti:

- CSGDEC;
- CSGHEX
- CSGNPUT;
- CSGXPUT.

CSGDEC

Serve per visualizzare un numero decimale (da 1 a 4 cifre) sul display. La sua sintassi è:

CSGDEC *indirizzo_slave,dato*

Dove:
- *indirizzo_slave* è l'indirizzo (da 0 a 3) dei display eventualmente collegati in cascata;
- *dato* è il numero vero e proprio da visualizzare.

Esempio:

```
CSGDEC 0,1234
```

Questo comando visualizza il numero 1234 sul display (configurato come slave 0).

CSGHEX

Serve per visualizzare un numero esadecimale (da 1 a 4 cifre) sul display. La sua sintassi è:

CSGHEX *indirizzo_slave,dato*

Dove:
- *indirizzo_slave* è l'indirizzo (da 0 a 3) dei display eventualmente collegati in cascata;
- *dato* è il numero vero e proprio da visualizzare.

Esempio:

```
Csghex 0,65535
```

Questo comando visualizza il numero esadecimale FFFF sul display (configurato come slave 0), risultato ottenuto dalla conversione automatica dal sistema decimale al sistema esadecimale.

CSGNPUT

Visualizza una SOLA cifra nel display alla posizione desiderata. La sua sintassi è:

CSGNPUT *indirizzo_slave, posizione_digit, dato*

Dove:
- indirizzo_slave è l'indirizzo (da 0 a 3) dei display eventualmente collegati in cascata;
- posizione_digit è un numero (compreso tra 0 e 3) che indica la posizione del singolo display;
- dato è il codice ASCII della cifra da stampare, secondo la seguente codifica:
  - da 48 a 57 per il numero compreso tra 0 e 9;
  - da 65 a 70 per le cifre esadecimali comprese tra A e F.

Esempio:

```
Csgnput 0,0,69
Csgnput 0,1,66
Csgnput 0,2,67
Csgnput 0,3,52
```

Questo esempio visualizza il codice EBC4 sul display (configurato come slave 0). Da notare che le singole cifre sono visualizzate su ogni singolo



Figura 9
Due moduli utilizzati

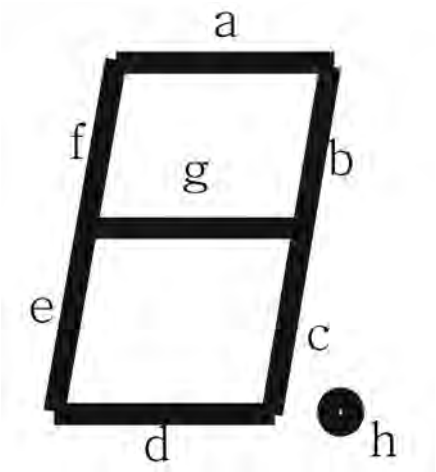


Figura 10
Il nome dei segmenti del display



Figura 11
La scritta LUPA sul display

display. Con questo comando è pertanto possibile intervenire su un solo display a sette segmenti, senza minimamente intervenire sugli altri.

CSGXPUT

Questo comando permette di illuminare o meno il singolo segmento dei display. Si può considerare pertanto come una istruzione a "basso livello" in quanto con essa si può intervenire su un solo segmento. È utile per effettuare visualizzare qualsiasi tipo di carattere oppure per effettuare dei test sui display, utilizzando una scenografia animata e di effetto. La sua sintassi è:

CSGNPUT *indirizzo_slave, posizione_digit, dato*

Dove:

- *indirizzo_slave* è l'indirizzo (da 0 a 3) dei display eventualmente collegati in cascata;
- *posizione_digit* è un numero (compreso tra 0 e 3) che indica la posizione del singolo display;
- *dato* è un **byte** formato dalla configurazione dello stato dei singoli segmenti, secondo la tabella 1 e la figura 10.

CODIFICA DEI SEGMENTI DI UN DISPLAY								
Bit	7	6	5	4	3	2	1	0
Segmento	H	G	F	E	D	C	B	A

Tabella 1
La codifica dei segmenti di un display

Esempio n° 1:

```
Csgxput 0,1,62
```

Questo comando visualizza la lettera U sul secondo display. Il numero 62 è ottenuto pesando i bit "attivi" della tabella. Per formare la lettera U infatti occorre illuminare i segmenti B-C-D-E-F, corrispondenti ai pesi 1-2-3-4-5. Il byte corrispondente è 00111110. Eseguendo la trasformazione da binario a decimale otteniamo:

$2^1+2^2+2^3+2^4+2^5=62$

Esempio n° 2:

```
Csgxput 0,0,56
Csgxput 0,1,62
Csgxput 0,2,115
Csgxput 0,3,119
```

Queste quattro istruzioni visualizzano sui quattro display la parola LUPA, illuminando i singoli segmenti con la codifica binaria.

Il classico conteggio alla rovescia

Approntiamo adesso il progetto della visualizzazione di un numero di 4 cifre, che ad ogni secondo decrementa il proprio valore di un'unità.
Non appena il conteggio arriva ala valore limite di 0, il buzzer della Study Board emetterà un suono. Il conteggio a ritroso è cadenzato da un breve beep.

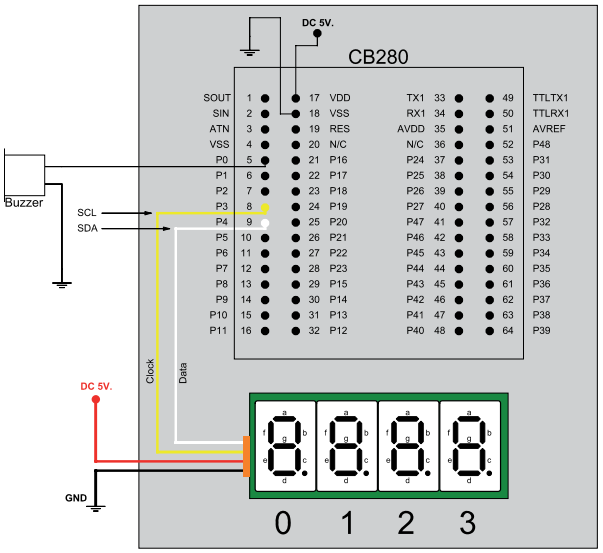


Figura 12
Schema elettrico per il conteggio alla rovescia

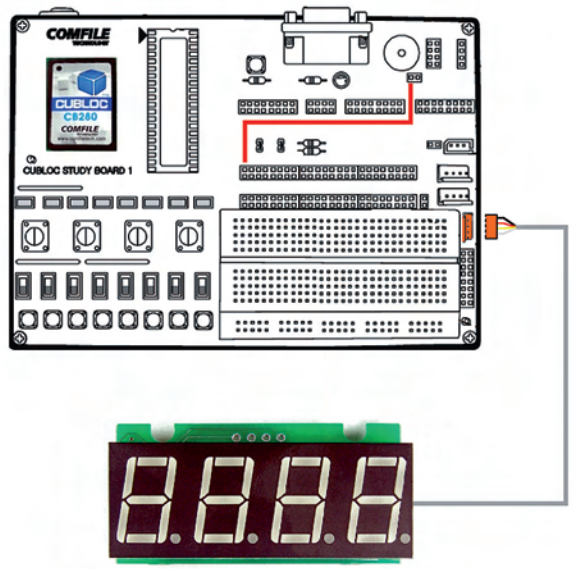


Figura 13
La connessione da effettuare sulla Study Board per il conteggio alla rovescia

Schema elettrico

Lo schema elettrico (figura 12) è al solito molto semplice, essendo tutte le operazioni logiche ed esecutive demandate al Cubloc. Le unità periferiche sono, ovviamente, il modulo display CSG ed il buzzer piezo da 5V ad alta impedenza. Il buzzer è collegato alla porta P0 del Cubloc, configurata naturalmente quale uscita.

Connessioni con la Study Board

L'unica connessione da effettuare è quella che collega il connettore delle porte di I/O del Cubloc con lo speaker piezo, come mostrato in figura 13

Il programma

Il programma è molto semplice e composto da poche righe, benché la sua funzionalità sia veramente elaborata. Dopo la definizione del dispositivo e alla configurazione della porta 0 quale uscita per lo speaker, viene dichiarata la variabile K. Essa è di tipo long, poiché anche se il suo range è positivo, alla conclusione del conteggio deve essere confrontata con un valore negativo. Se venisse dichiarata di tipo integer, il ciclo For non si concluderebbe mai.

Pertanto l'iterazione ha luogo e la variabile K permette il ripetersi del ciclo decrementando il suo valore. All'interno del ciclo FOR/NEXT sono presenti:

- l'istruzione che permette di eseguire un breve beep sul piezo ad ogni decremento (beep 0,100);
- il comando per visualizzare il numero in corso sul display (csgdec 0,k);
- la pausa d'attesa di un secondo (delay 1000).

Al termine del ciclo, un onda rettangolare dal periodo di 2ms è generata dalla porta P0, producendo un suono senza fine sul buzzer. Il display naturalmente visualizza la cifra 0. È ovvio che se si desidera un valore di partenza diverso da quello proposto, basta modificare il valore 9999 con un altro numero, ricordandosi però di non superare le quattro cifre.

```
' LISTATO 2
' Conteggio alla rovescia
Const Device=cb280
Output 0
Dim k As Long
Set I2c 4,3
For k=9999 To 0 Step -1
    Beep 0,100
    Csgdec 0,k
    Delay 1000
Next
Do
    High 0
    Delay 1
    Low 0
    Delay 1
Loop
```

Conclusioni

Come visto i display a 7 segmenti con interfaccia I2C consentono di risparmiare un gran numero di porte sul micro dal momento che la comunicazione avviene mediante solo due fili. Nel prossimo numero vedremo una rassegna di applicazioni pratiche con i display sfruttando quanto visto in questo articolo.

More Info Please!
Inserisci il Codice **268066** alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

INWARE EDIZIONI SHOP

CD-ROM

Annate complete FE su CD-ROM



Annate disponibili dal 2003 al 2006

Le annate complete in formato PDF. Potrai sfogliare comodamente tutte le riviste e stampare (senza perdere in qualità) gli articoli di tuo interesse. Ogni CD-ROM contiene anche software, codice sorgente, circuiti stampati e tutto il materiale necessario per la realizzazione dei progetti proposti. In ogni CD è presente una sezione con contenuti speciali.

COD. **FE-CD2003 € 25,80**
COD. **FE-CD2004 € 30,00**
COD. **FE-CD2005 € 30,00**
COD. **FE-CD2006 € 30,00**

PIC Microcontroller By Example



Il corso completo PIC® Microcontroller By Example in formato PDF

Tutte le lezioni pronte per la consultazione con i sorgenti dei progetti immediatamente utilizzabili nelle tue applicazioni. Il modo migliore per avere sempre sottomano la soluzione per il tuo progetto con i **PICmicro®**. Il **CD-ROM PIC® Microcontroller By Example** contiene una sezione **"Contenuti Speciali"** tutta da scoprire.

COD. **FE-PBE € 15,90**

Tutto sulle Smartcard



La raccolta completa degli articoli sulle smartcard in formato PDF

Gli articoli, i progetti e i download relativi agli articoli sulle Smartcard in un unico CD-ROM pronto da consultare ed utilizzare. Contiene i progetti completi del lettore di smartcard **UniReader** e del **SimCopier** per duplicare il contenuto della rubrica della vostra Sim card.

COD. **FE-SMARTCARD € 15,90**

Annata 2006 Firmware completa su CD-ROM



10 numeri di Firmware su un CD-ROM

Questo CD contiene tutti i numeri di Firmware usciti nel 2006 in formato PDF ad alta risoluzione. Sono presenti tutti i listati dei codici presentati sulla rivista, potrai quindi comodamente utilizzarli insieme agli articoli di tuo interesse. Il Cd contiene inoltre una sezione con interessanti contenuti speciali.

COD. **FW-CD2006 € 30,00**

Scopri i bundle e le offerte che ti abbiamo riservato

PROPOSTE 2007

Display LCD



Questo libro di successo (oltre 2000 copie vendute) rappresenta una delle migliori guide all'utilizzo dei moduli alfanumerici basati sul controller HD44780, moduli grafici con controller KS0108 e non solo. Il testo tratta anche i display LED a sette segmenti e i display LCD passivi. Numerosi gli esempi pratici di impiego dei vari dispositivi: dal contatore a 7 segmenti al termometro LCD fino al pilotaggio dei moduli alfanumerici mediante PICmicro e PC.

COD. FE-06 **€ 16,50**

PICmicro™



La lettura di questo libro è consigliata per conoscere a fondo i PICmicro seguendo un percorso estremamente pratico e stimolante. Il testo descrive l'uso di MPLAB®, e descrive, in maniera approfondita, tutte le istruzioni assembler e molte delle direttive del compilatore. Al testo è allegato un utilissimo CDROM che, oltre ai sorgenti e gli schemi dei progetti presentati nel testo, contiene moltissimi programmi di utilità e molta documentazione.

COD. FE-18 **€ 29,00 (contiene CD-ROM)**

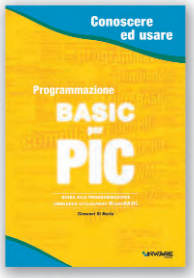
Linguaggio ANSI C



Questo nuovissimo libro descrive le tecniche, gli accorgimenti migliori per sfruttare gli aspetti di "alto e basso livello" del C, entrambi fondamentali quando si vuole sviluppare del *firmware* per sistemi dotati di risorse limitate.
Il testo è particolarmente indicato sia a chi ha già esperienza nella programmazione in *assembler* di sistemi a microcontrollore ed intende iniziare ad utilizzare il linguaggio C, sia per chi conosce già il C e vuole avvicinarsi alla programmazione dei sistemi *embedded*.

COD. FE-25 **€ 24,90**

BASIC per PIC



Un volume indispensabile sia per chi si avvicina alla programmazione dei PIC utilizzando il linguaggio Basic, sia per chi intende affinare le proprie tecniche di programmazione. Una guida alla programmazione embedded utilizzando MikroBASIC, uno dei più completi compilatori per PIC dotato di ambiente IDE e moltissime funzioni di libreria. La trattazione vi guiderà dalla semplice accensione di un LED alla gestione di motori in PWM, alla lettura e scrittura di memorie I2C, alla generazione di suoni seguendo un percorso semplice e ricchissimo di esempi pratici.

COD. FE-27 **€ 24,90**

CPLD



Un libro dedicato a tutti coloro che per la prima volta si avvicinano al mondo delle Logiche Programmabili ed utilizzabile da quanti, già esperti, desiderano approfondire la conoscenza di questi interessanti dispositivi. Gli argomenti teorici sono presentati attraverso semplici circuiti di esempio il cui codice viene descritto nei dettagli. Tra gli argomenti trattati: la sintassi del linguaggio Verilog, la comunicazione seriale, la conversione analogico-digitale e le macchine a stati finiti.

COD. FE-28 **€ 32,90 (contiene DVD-ROM)**

Amplificatori operazionali

nuovo!



Un testo per capire a fondo l'amplificatore operazionale. Le tipologie, le configurazioni circuitali, l'analisi approfondita dei parametri caratteristici sono solo alcuni degli argomenti trattati nel testo. I numerosi esempi pratici fanno di questo libro un utilissimo volume adatto anche sia a chi intende conoscere gli amplificatori operazionali senza per questo affrontare lunghe trattazioni teoriche, sia per gli studenti che hanno la necessità di conoscere a fondo questo affascinante ed utilissimo componente elettronico.

COD. FE-29 **€ 39,00**

LIBRI

www.ieshop.it

I prezzi indicati sono IVA compresa

Terza parte: *Giugno 2007*
Il controller T6963C

Quarta parte: *Luglio/Agosto 2007*
Demograph: la scheda di sviluppo

Quinta parte: *Settembre 2007*
TLCD: la libreria di gestione del controller grafico

Sesta parte: *Ottobre 2007*
Utilizzare il Demograph come album elettronico

LCD dalla A alla Z

In questo articolo verranno aggiunte altre funzionalità alla libreria ad alto livello, creata nella precedente puntata. Sarà così possibile disegnare linee e rettangoli ed evidenziare i testi con un semplice trucco. Infine, per mostrare le potenzialità della scheda di test realizzata, il Demograph verrà utilizzato come visualizzatore di immagini.

Disegnare linee

Nell'articolo precedente abbiamo creato una libreria a basso livello, chiamata tlcd, che ci permette di colloquiare in modo semplice con il controller T6963C del display grafico. Si tratta di una libreria a basso livello, poiché permette di inviare i comandi "primitivi" al controller: leggi o scrivi un byte dalla VRAM, abilita la grafica e/o il testo, ecc.
Per rendere il compito più semplice al programmatore, abbiamo anche creato una libreria ad alto livello dedicata alla scheda di test Demograph, progettata e realizzata nella Parte III di questa serie di articoli. Con questa libreria è sufficiente richiamare alcune semplici funzioni per gestire il display (oltre che le altre periferiche come la EEPROM esterna). Per esempio, basta richiamare la `demogr_init()` all'inizio del nostro programma per inizializzare correttamente il display ed utilizzare la `printxy()` per scrivere del testo in un punto ben preciso dell'LCD.
In una qualsiasi applicazione, però, il programmatore vuole utilizzare il display grafico, oltre che per scrivere testo, anche per disegnare alcune figure geometriche o delle immagini. Per esempio, delle linee.
È per questo che abbiamo pensato di aggiungere alla nostra libreria ad alto livello le funzioni che ci permettono di disegnare alcune semplici figure geometriche.

VRAM e pixel

Come si può facilmente immaginare, disegnare linee, rettangoli o altro vuol dire "accendere" o "spegnere" alcuni pixel della matrice del display, cioè modificare il valore di opportuni byte nella VRAM. È per questo che è necessario conoscere esattamente qual è la corrispondenza tra i pixel a video e i byte/bit in VRAM. Questa corrispondenza è evidenziata nella Figura 1 (essa è valida *soltanto* nel caso in cui il font scelto sia pari a 8x8 pixel).

Ricordiamo che con il termine VRAM (video-RAM) intendiamo la memoria RAM integrata nella sche-

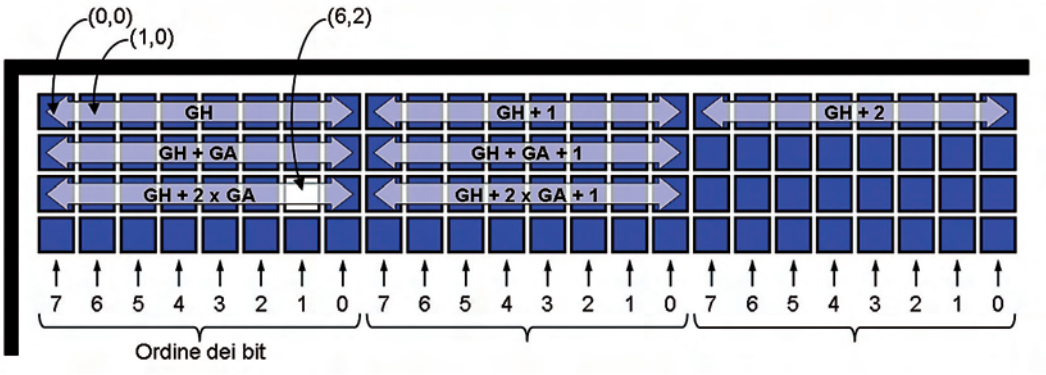


Figura 1
Corrispondenza tra pixel dell'LCD e byte della VRAM

Utilizzare il Demograph come album fotografico elettronico



di Giuseppe Modugno

da di controllo del display e interfacciata direttamente al controller T6963C, da non confondere assolutamente con la RAM del microcontrollore del Demograph.
Come si può vedere nella figura, il valore del byte all'indirizzo GH (GRAPHIC HOME) rappresenta lo stato dei primi 8 pixel orizzontali in alto a sinistra del display, quelli di coordinate (0,0)-(7,0): il bit più significativo di indice 7 corrisponde al pixel che si trova al vertice del display. Il byte all'indirizzo successivo (GH+1) rappresenta lo stato dei successivi 8 pixel orizzontali sul display, di coordinate (8,0)-(15,0). E così via, fino ad esaurimento dei pixel della prima riga.
Il byte che rappresenta i primi 8 pixel della seconda riga del display, di coordinate (0,1)-(7,1), è quello che si trova all'indirizzo GH+GA (GA è la GRAPHIC AREA). Come al solito, il byte all'indirizzo successivo (GH+GA+1) rappresenta i successivi 8 pixel e così via.
Per accendere il pixel di coordinate (6,2) evidenziato nella figura è necessario modificare il bit 1 del byte che si trova all'indirizzo GH+2xGA della VRAM.
Ricordiamo che sia GH che GA sono parametri che vengono impostati attraverso l'invio di opportuni comandi al controller grafico. La libreria a basso livello, sviluppata nel precedente articolo, si occupa di questa impostazione all'esecuzione della funzione di inizializzazione `tlcd_init()` (nel file tlcd.h, GH=0x0400 e GA=64). A sua volta, la funzione `tlcd_init()` è richiamata dalla `demogr_init()`.
Calcoliamo ora l'indirizzo del byte che corrisponde ad un generico punto del display di cui si conoscono le coordinate in pixel (x,y). Consideriamo inizialmente il caso semplice di y=0, cioè la prima riga di pixel del display. Se x è compreso tra 0 e 7, il byte relativo si trova all'indirizzo GH. Se x è compreso tra 8 e 15, l'indirizzo è GH+1. Non è difficile convincersi che l'indirizzo in generale è (GH+x/8), dove la divisione deve essere fatta con troncamento. Se y≠0, bisogna aggiungere a questo indirizzo tante volte GA. In definitiva, l'indirizzo del byte associato al pixel (x,y) può essere calcolato in linguaggio C nel seguente modo:

```
GH + GA*y + (x>>3)
```

La divisione per 8 con troncamento è ovviamente stata fatta in modo semplice shiftando a destra di 3 posizioni il valore di x.
Oltre all'indirizzo del byte in VRAM associato al pixel di coordinate (x,y), è necessario conoscere anche il numero del bit relativo. Anche in questo caso possiamo facilmente capire qual è la formula da utilizzare per induzione: se x=0, il bit è 7; se x=1, il bit è 6; se x=7, il bit è 0; se x=8, il bit è 7; ecc. In pratica serve conoscere soltanto i 3 bit meno significativi di x e fare il complemento a 7, cioè in linguaggio C:

```
7 – (x&0x07)
```

Ricordiamoci queste due formule appena calcolate perché saranno ampiamente utilizzate nel seguito.

Disegnare una linea verticale

Proviamo ora a disegnare una linea verticale sul display grafico.
È abbastanza semplice capire quali bit della VRAM

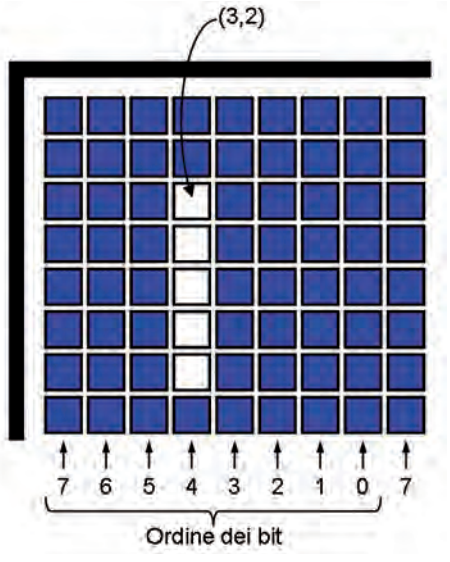


Figura 2
I pixel da modificare per una linea verticale a partire dal punto (3,2) di lunghezza 5

Listato 1

Funzione lcd_vline() [demograph.c]

```
void
lcd_vline( unsigned char x, unsigned char y, unsigned char h, unsigned char c )
{
  /* Disegna una linea verticale a partire dalla coordinata (x,y), altezza h
   * e colore c (COLOR_WHITE o COLOR_BLUE).
   */
  unsigned int addr = GRAPHIC_HOME + GRAPHIC_AREA*y + (x>>3);
  unsigned char bit = 7 - (x&0x07);

  while( h— ) {
      tlcd_addrptrset( addr );
      if( c==COLOR_WHITE )
        tlcd_bitset( bit );
      else
        tlcd_bitreset( bit );
        addr += GRAPHIC_AREA;
  }
}
```


bisogna modificare per disegnare una linea verticale, con un dato punto di partenza ed una data lunghezza. Per esempio, nel caso di punto iniziale di coordinate (3,2) e lunghezza 5, i pixel (quindi i bit) da modificare sono quelli evidenziati in Figura 2.
Si noti che dovremo modificare lo stato di tanti byte della VRAM quanti sono i pixel che compongono la linea. Inoltre, per ogni byte, dovremo andare a modificare sempre lo stesso bit (nell'esempio, il bit di indice 4).

La funzione `lcd_vline()` che permette di disegnare una linea verticale è mostrata nel listato 1.
La variabile locale addr è subito inizializzata con l'indirizzo VRAM del byte corrispondente al pixel di coordinate (x,y), primo punto della linea. La variabile locale bit rappresenta il numero del bit all'interno del byte da "accendere". Le formule utilizzate sono quelle che abbiamo ricavato precedentemente. A questo punto, il resto della funzione è molto semplice da capire. Essa non è altro che un ciclo fatto h volte (dove h sta per la parola inglese height, cioè altezza), la lunghezza della linea che vogliamo disegnare. All'interno del ciclo, viene impostato l'indirizzo VRAM su cui vogliamo operare e andiamo ad "accendere" oppure a "spegnere" il bit di indice `bit`. L'"accensione" o lo "spegnimento" dipende dal colore con cui vogliamo disegnare la linea, cioè dal parametro c che può assumere i valori `COLOR_WHITE` o `COLOR_BLUE` (si veda a tal proposito il file demograph.h). Per passare al byte del successivo pixel, cioè alla riga successiva, è sufficiente aggiungere GA alla variabile `addr`.

Disegnare una linea orizzontale

Sperando di non avervi fatto impazzire con le formule precedenti, passiamo ad un altro compito che presto scoprirete essere ancora più complicato. Si tratta di disegnare una linea orizzontale.
Proviamo a renderci conto, guardando la Figura 3, dei pixel da modificare per disegnare una linea che parte dal punto di coordinate (3,2) e ha una lunghezza pari a 14 pixel.
In generale, il primo byte dovrà essere solo parzialmente modificato nella parte meno significativa. Successivamente ci sarà un certo numero di byte da modificare completamente ("accendendo" tutti i bit nel caso di colore bianco). Infine ci sarà un ultimo byte da modificare solo parzialmente nella parte più significativa.
Essendo un procedimento più difficile da spiegare che da capire, faremo sempre riferimento all'esempio in Figura 3.
Ognuna delle 3 fasi può essere presente oppure no, a seconda delle coordinate e della lunghezza della linea che si vuole disegnare. Per esempio, se la linea è molto corta è probabile che sia necessario modificare parzialmente soltanto il primo e l'ultimo byte, quindi saltando la seconda fase.

La funzione `lcd_hline()` è di conseguenza divisa in 3 distinte parti che andremo a commentare singolarmente, iniziando dal frammento nel listato 2.

Come al solito, la variabile locale addr memorizza l'indirizzo VRAM del byte che vogliamo modificare. L'indirizzo del primo byte viene calcolato con la consueta formula vista precedentemente. Tale

con il patrocinio di
Ministero delle Comunicazioni
Comune di Faenza

mostra mercato

Fxpo® Elettronica

FAENZA 20·21 ottobre 2007

FIERE Viale Risorgimento, 1 ore 9/18

Elettronica per tutto, Elettronica per tutti!

RADIO EXPO'

MERCATINO delle RADIOCOMUNICAZIONI

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

Solo Sabato 20 ottobre



VALE COME RIDOTTO

NEWS ON LINE Vai nel sito www.blunautilus.it e iscriviti a NEWSLETTER: 10 giorni prima di un evento riceverai una e-mail di promemoria

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

Codice MIP 268077

Rif. FARELETTRONICA

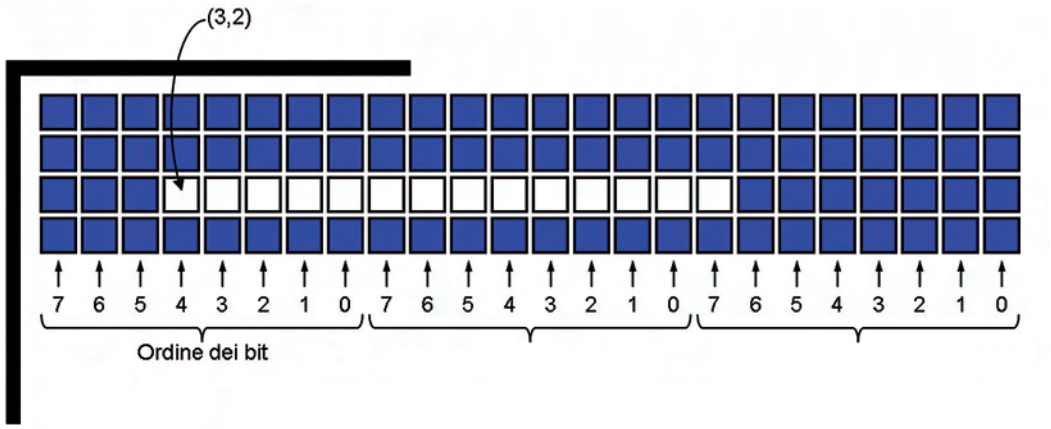


Figura 3
I pixel da modificare per una linea orizzontale a partire dal punto (3,2) di lunghezza 14

indirizzo viene subito impostato come corrente mediante la funzione `tlcd_addrptrset()`.
Il primo `if`, con tutte le istruzioni contenute all'interno, rappresenta la prima fase del disegno di una linea orizzontale. Si entra all'interno di questa fase solo se la variabile `mask`, pari a (`x&0x07`), è diversa da zero. In pratica, l'espressione (`x&0x07`) è pari ai soli 3 bit meno significativi di x che ci indicano se questo valore è allineato al byte. Se questi 3 bit sono *tutti* uguali a zero (quindi `mask` è nullo), il valore di x è allineato al byte (in pratica è un multiplo di 8) e il primo byte da disegnare non deve essere parzialmente, ma completamente, modificato. In altre parole, bisogna saltare la prima fase del disegno e passare direttamente alla seconda.
Nel nostro esempio, `mask` assumerà il valore 3, quindi la condizione dell'`if` è verificata.
Supponiamo di aver superato la condizione del primo `if`, cioè dobbiamo modificare il primo byte parzialmente, andando ad "accendere" o "spegnere" alcuni bit meno significativi che rappresentano i pixel della prima parte della linea. Calcoliamo subito il numero di questi pixel da modificare, memorizzando tale valore nella variabile locale `b`. Se `mask`, cioè i 3 bit meno significativi di `x`, è pari ad 1 (x=1, 9, 17, ecc.), il numero di pixel da manipolare sarà 7. Se è pari a 2 (x=2, 10, 18, ecc.), il numero di pixel da manipolare sarà 6. E così via. Non è difficile convincerci che la formula corretta per calcolare il numero di pixel del primo byte è `8-mask` (nell'esempio, i pixel del primo byte da modificare sono effettivamente 5, come si vede in Figura 3).

Conoscendo ora il numero dei bit meno significativi da modificare, ci serve creare una maschera di 8 bit (un byte) con tanti 1 a destra quanti sono questi bit. In pratica, `mask`=0b0000 0001 per `b`=1, `mask`=0b0000 0011 per `b`=2, ecc. È possibile calcolare "al volo" la maschera a partire dal numero dei bit mediante delle operazioni logiche sulla variabile `b`, ma abbiamo preferito utilizzare un array costante `maskbit[]` per rendere l'operazione più veloce. Nell'esempio, `mask` verrà posto uguale a 0b0001 1111.

A questo punto la strada potrebbe diramarsi, poiché non dobbiamo dimenticare alcuni casi particolari, cioè quando non tutti i bit meno significativi del nostro primo byte devono essere modificati. Questo può accadere se la lunghezza `w` della linea è molto piccola e per certi valori di `x`. Se la linea dell'esempio avesse avuto una lunghezza pari a 3 (anziché 14), la maschera dei bit da modificare sarebbe stata 0b0001 1100 e non 0b0111 1111, così come calcolato finora. È per questo che si trova il secondo `if` annidato al primo. Se la lunghezza della linea `w` è minore del numero dei bit meno significativi, vuol dire che si tratta di una linea "degenere" molto piccola e la maschera del primo byte non corrisponde esattamente al valore appena calcolato. Per ottenere la maschera corretta, viene calcolato prima il numero dei bit meno significativi da *non modificare* (istruzione `b-=w`; nell'esempio `b`=2), successivamente la maschera dei bit associata a questo numero (istruzione `mask2=maskbit[b];` nell'esempio `mask2`=0b0000 0011) e quindi la maschera definitiva azzerando i bit da non modificare (istruzione `mask&=~mask2;` nell'esempio `mask`=0b0001 1100). Come ultima istruzione, viene annullato il valore di `w`, visto che, nel caso "degenere", la linea termina sicuramente al primo byte. Non entrando in questo secondo `if`, al valore di `w` viene semplicemente sottratto il numero dei bit meno significativi del primo byte (nell'esempio iniziale `w`=9). In questo modo, alla fine della prima fase, `w` rappresenta il numero dei pixel ancora da disegnare.
A questo punto, con grande fatica, abbiamo la nostra maschera ad 8 bit dove i bit ad 1 rappresentano i pixel da "accendere" nel caso di colore bianco, oppure da "spegnere" nel caso di colore blu. È

Listato 2

Prima parte della funzione lcd_hline() [demograph.c]

```
void
lcd_hline( unsigned char x, unsigned char y, unsigned char w, unsigned char c )
{
  /* Disegna una linea orizzontale a partire dalla coordinata (x,y), larghezza w
   * e colore c (COLOR_WHITE o COLOR_BLUE).
   */
  unsigned int addr = GRAPHIC_HOME + GRAPHIC_AREA*y + (x>>3);
  unsigned char mask, mask2, b;

  tlcd_addrptrset(addr);
  if( mask=(x&0x07) ) {
    /* Il primo byte deve essere parzialmente scritto. */
    b = 8-mask;       /* Numero di pixel da riempire. */
    /* Calcola la maschera per modificare i bit meno significativi. */
    mask = maskbit[b];
    if( w<b ) {
      /* Non tutti i pixel a dx (bit meno significativi) del primo byte
       * devono essere riempiti... è una linea molto corta. */
      b -= w;
      mask2 = maskbit[b];
      mask &= ~mask2;
      w = 0;
    } else
      w -= b;

    /* Legge il byte da modificare dalla VRAM. */
    b = tlcd_dataread( TLCD_AUTOUNCH );
    if( c==COLOR_WHITE )
      b |= mask;
    else

      b &= ~mask;
    }
    /* Aggiorna il byte in VRAM. */
    tlcd_datawrite( b, TLCD_AUTOINCR );
  }
  ...
```

questo quello che viene fatto con le ultime istruzioni: viene letto dapprima il valore del byte dalla VRAM, applicata la maschera (positiva o negativa a seconda del colore) e scritto il nuovo valore così calcolato. Si noti che l'operazione di lettura viene fatta senza modificare il puntatore in VRAM (parametro `TLCD_AUTOUNCH`), poiché la successiva operazione di scrittura dovrà avvenire sempre *sulla stessa* locazione di memoria (il vecchio valore dovrà essere sovrascritto). L'operazione di scrittura, invece, avviene avanzando il puntatore in VRAM così da essere pronti alla modifica degli altri byte per il completamento del disegno.

Passiamo quindi alla seconda fase, per fortuna molto più semplice della precedente, mostrata nel nel listato 3.

Si tratta di modificare interamente un certo numero di byte consecutivi. Per farlo, ci serviamo della funzionalità di Autowrite del controller T6963C che permette di fare proprio questo, diminuendo al minimo i comandi da inviare al controller, quindi riducendo anche i tempi dell'operazione.

A questo punto ci serve solo sapere quanti sono i byte da modificare. Ricordiamoci che ora la variabile w indica la "lunghezza rimanente" della linea, cioè il numero dei pixel che devono essere ancora modificati. Ovviamente, il numero dei byte interi sarà pari alla divisione, con troncamento, di `w` per 8.

Listato 3

Seconda parte della funzione lcd_hline() [demograph.c]

```
...
/* Scrive (w>>3) byte consecutivi in
/* VRAM. */
if( c==COLOR_WHITE )
  tlcd_autowrite_set( 0xFF, w>>3 );
else
  tlcd_autowrite_set( 0x00, w>>3 );
w &= 0x07;
...
```

Listato 4

Terza parte della funzione lcd_hline() [demograph.c]

```
...
if( w ) {
  /* L'ultimo byte deve essere
  /* parzialmente scritto. */

  /* Legge il byte da modificare dalla
  /* VRAM. */
  b = tlcd_dataread( TLCD_AUTOUNCH );

  mask = maskbit[8-w];
  if( c==COLOR_BLUE )
    b &= mask;
  else
    b |= ~mask;

  /* Aggiorna il byte in VRAM. */
  tlcd_datawrite( b, TLCD_AUTOINCR );
  }
```

Come al solito, questa divisione con troncamento viene effettuata, in linguaggio C, mediante shift a destra di 3 posizioni. Nel nostro esempio, il numero dei byte da modificare per intero è soltanto 1 (in alcuni casi, potrebbe anche essere 0, cioè la seconda fase non modificherà alcun byte in VRAM).
Se il colore della linea da disegnare è bianco, i byte dovranno essere impostati a 0xFF (tutti i pixel accesi), altrimenti dovranno essere impostati a 0x00 (tutti i pixel spenti).
L'ultima istruzione calcola i rimanenti pixel della linea ancora da disegnare. Questo numero sarà sicuramente compreso tra 0 e 7, altrimenti ci sarebbe un altro byte da modificare interamente. Esso è proprio pari ai 3 bit meno significativi di `w`, calcolati azzerando i 5 bit più significativi (istruzione `w&=0x07;`). Nel nostro esempio, rimane solo un pixel da disegnare per completare la linea.
Passiamo così all'ultima fase mostrata nel listato 4.
Naturalmente si entra in questa fase se ci sono ancora pixel da modificare, cioè se `w` non è nullo, altrimenti la linea è stata già completamente disegnata. Viene subito letto il valore del successivo byte (si ricordi che l'operazione di Autowrite incrementa automaticamente l'indirizzo VRAM corrente) e memorizzato nella variabile locale `b`. Dopo l'operazione di lettura, l'indirizzo non viene incrementato, poiché dovrà essere successivamente scritto con il nuovo valore.
Si calcola subito la maschera corrispondente al numero dei bit da *non* modificare (istruzione `mask=maskbit[8-w];` nel nostro esempio `mask=maskbit[7]`=0b0111 1111). Questa volta gli "zeri" della maschera rappresentano i bit da modificare e gli "uni" quelli da non modificare.
Se la linea è di colore blu, la maschera può essere utilizzata così com'è per azzerare i bit più significativi (istruzione b&=mask;). Se il colore scelto è bianco, la mascherà dovrà prima essere negata per poter "attivare" i bit più significativi (istruzione `b|=~mask;`). Il nuovo valore così ottenuto viene scritto in VRAM, sovrascrivendo il precedente.
E questo conclude finalmente il disegno di una generica linea orizzontale.

Ne valeva la pena?

Sono sicuro che qualcuno, considerandomi un temerario, si sta chiedendo il perché di tanta complessità. Non era più semplice accendere (o spegnere) tutti i pixel, uno dopo l'altro, richiamando la `tlcd_bitset()` (o la `tlcd_bitreset()`), facendo un semplice ciclo da eseguire `w` volte? Non abbiamo ucciso una mosca con un cannone?
A prima vista costui sembrerebbe aver ragione, ma, ragionandoci sopra, le cose non stanno proprio così. Considerando sempre la nostra linea orizzontale e il metodo alternativo proposto dal mio ipotetico interlocutore, calcoliamo il numero dei comandi da inviare al controller grafico. Sicuramente sarà necessario, per ogni pixel, inviare il comando di BIT SET/RESET. Per ogni byte, sarà necessario inviare il comando di SET ADDRESS POINTER. Nel nostro esempio di riferimento, saranno da inviare 3 comandi di SET ADDRESS POINTER e 14 comandi BIT SET/RESET per un totale di 17 comandi. Con il metodo proposto in questo articolo, invece, i comandi da inviare sono:

- SET ADDRESS POINTER, DATA READ, DATA WRITE per la prima fase;
- AUTOWRITE per la seconda fase
- DATA READ, DATA WRITE per la terza fase.

Per un totale di 6 comandi, con un risparmio di quasi il 60%!
Si tenga presente che, generalmente, le routine di gestione del display grafico sono *molto lente* e rap-

presentano un collo di bottiglia non trascurabile durante l'esecuzione del programma (non ci dimentichiamo che, prima di inviare un comando, è necessario *aspettare* che il controller si liberi dalle sue operazioni). Spesso, anche nei moderni computer, la grafica (videogiochi o film) rappresenta un fardello considerevole per la CPU principale o della scheda video.
Al contrario, le routine di calcolo che avvengono nel microcontrollore sono, in confronto, completamente trascurabili, diciamo a "costo quasi zero".
In definitiva, ottimizzare le routine di gestione del display grafico, riducendone il tempo di esecuzione, magari complicando soltanto le operazioni di calcolo della CPU, permette di creare programmi molto veloci e performanti.
Sperando di aver convinto anche i più perplessi, passiamo oltre.

Linee diagonali

Per semplicità, abbandoniamo l'idea di creare delle routine per il disegno di linee qualsiasi conoscendone i due estremi (x1,y1) e (x2,y2). Il compito, come potete immaginare, non è assolutamente semplice. La prima cosa da fare è decidere quali sono, all'interno della matrice del display, i pixel da disegnare. Procedendo in modo "classico", dovremmo scomodare funzioni di trigonometria che ci porterebbero facilmente a fare dei calcoli in virgola mobile, cosa assolutamente da evitare in un microcontrollore semplice come un PIC.
Per fortuna ci aiutano alcuni algoritmi inventati appositamente per risolvere questo tipo di problemi e che fanno uso soltanto di numeri interi, facilmente gestibili da un PIC. Il precursore di questi algoritmi è stato quello di Bresenham che risolse brillantemente il problema di pilotare un plotter per disegnare linee. Questo algoritmo, ulteriormente ottimizzato ed ampliato, è usato oggi nella maggior parte dei dispositivi elettronici che hanno un display grafico per la trasformazione di una linea ideale in un insieme di pixel da "accendere" (rasterizzazione della linea). Anche il disegno di un cerchio su una matrice di pixel viene effettuato mediante un algoritmo derivato da quello originale di Bresenham.
Per chi fosse interessato ad approfondire l'argomento, tra i riferimenti in fondo all'articolo, potete trovare qualche link utile.

Disegnare rettangoli

Concediamoci ora un momento di riposo, riponendo la nostra attenzione sui rettangoli. Seppure apparentemente più complicati, il compito di disegnare rettangoli diventa molto semplice se ci rendiamo conto che queste figure geometriche non sono altro che un insieme di linee.

Il bordo

Nel listato 5 è mostrata la funzione `lcd_rect()` che permette di disegnare un rettangolo (solo il bordo) a partire dalle coordinate (x,y) del vertice in alto a sinistra, la larghezza w, l'altezza h e il colore c.
La prima istruzione disegna la linea orizzontale superiore a partire dal punto di coordinate (x,y) e lunghezza w, pari alla larghezza del rettangolo.
La seconda istruzione disegna la linea orizzontale inferiore a partire dal punto di coordinate (x,y+h-1) e stessa lunghezza.
La terza istruzione disegna la linea verticale sinistra a partire dal punto di coordinate (x,y+1) e altezza (h-2).
La quarta istruzione, infine, disegna la linea verticale destra a partire dal punto di coordinate (x+w-1,y+1) e stessa altezza (h-2).
Anche in questa funzione abbiamo cercato di ottimizzare quello che era possibile, notando che i 4 vertici del rettangolo vengono disegnati già con le prime 2 istruzioni e che sarebbe stato inutile ridisegnarli con le ultime 2 istruzioni (è per questo che

Listato 5

Funzione per il disegno di rettangoli lcd_rect() [demograph.c]

```
void
lcd_rect( unsigned char x, unsigned char y, unsigned char w, unsigned char h, unsigned
char c )
{
  /* Disegna un rettangolo con il punto in alto a sinistra di coordinate (x,y),
   * larghezza w, altezza h e colore c.
   */
  lcd_hline( x, y, w, c );
  lcd_hline( x, y+h-1, w, c );
  lcd_vline( x, y+1, h-2, c );
  lcd_vline( x+w-1, y+1, h-2, c );
}
```

Listato 6

Funzione per il disegno di rettangoli lcd_rectfill() [demograph.c]

```
void
lcd_rectfill( unsigned char x, unsigned char y, unsigned char w, unsigned char h, unsigned
char c )
{
  /* Disegna un rettangolo ripieno con il punto in alto a sinistra di coordinate
   * (x,y), larghezza w, altezza h e colore c.
   */
  while( h— )
    lcd_hline( x, y++, w, c );
}
```

la lunghezza delle linee verticali è h-2 anziché h, quanto l'altezza di tutto il rettangolo).

Il contenuto

La funzione precedente permette solo di disegnare il bordo del rettangolo. Con la funzione `lcd_rectfill()`, mostrata nel listato 6, è possibile disegnare l'area di un rettangolo, il contenuto. Come al solito, si parte dal vertice in alto a sinistra di coordinate (x,y), larghezza w, altezza h e colore c.
L'area interna di un rettangolo non è altro che un insieme di linee orizzontali. La funzione `tlcd_rectfill()` non fa altro che disegnare, mediante il ciclo `while()`, h linee orizzontali, tutte della stessa lunghezza e tutte a partire dalla stessa coordinata orizzontale x. La coordinata verticale, a partire da y, viene incrementata ad ogni ciclo. In pratica, l'area del rettangolo viene riempita da sinistra verso destra e dall'alto verso il basso. Perché ho scelto di disegnare linee orizzontali anziché verticali? Ovviamente, sempre nell'ottica dell'ottimizzazione del codice e dei tempi di esecuzione. Provare per credere!
È bene sottolineare che questa funzione disegna *soltanto* l'area interna di un rettangolo, senza disegnare il bordo. Se vogliamo disegnare sia il bordo bianco che cancellare l'interno (quindi colorandolo di blu), sarà necessario richiamare le due funzioni, più o meno in questo modo:

```
lcd_rect( x, y, w, h, COLOR_WHITE );
lcd_rectfill( x+1, y+1, w-2, h-2,
              COLOR_BLUE );
```

Testo in reverse

Ormai i computer ci hanno abituato ad associare al testo visualizzato con i colori negati una selezione. Se normalmente il testo è nero su sfondo bianco, il testo selezionato sarà bianco su sfondo nero. La selezione indica una porzione "attiva" del testo su cui effettuare delle operazioni.
Anche sul nostro piccolo display grafico potrebbe essere utile visualizzare il testo in reverse, per esempio per indicare la voce correntemente selezionata di un menu, cosa frequente in diversi prodotti elettronici come i cellulari.
Nella Parte III dedicata al controller T6963C, abbiamo visto come sia possibile modificare il testo visualizzato tramite degli "attributi", tra cui c'è anche il reverse. Purtroppo, però, se vogliamo utilizzare la visualizzazione del testo tramite attributi, dovremo rinunciare alla possibilità di utilizzare la grafica. Cosa ce ne facciamo di un display grafico senza grafica?
È per questo che abbiamo pensato di trovare un escamotage a questa limitazione, cercando un modo alternativo alla selezione del testo. La soluzione è quella di abbandonare l'idea degli "attributi" del testo, impostando una visualizzazione *contemporanea* della grafica e del testo in modalità XOR (si veda a tal proposito la Parte III). In pratica, se la parte testo dice che un pixel deve essere acceso e la parte grafica dice che quel pixel deve essere spento, complessivamente quel pixel sarà acceso (risultato di 1 XOR 0). Di contro, se sia la parte testo che la parte grafica sono per l'accensione dello stesso pixel, complessivamente quel pixel sarà spento (risultato di 1 XOR 1).
Proviamo ad immaginare cosa succede se disegniamo, mediante la funzione `lcd_rectfill()` creata precedentemente, un rettangolo bianco in coincidenza con una stringa di testo. Otterremo proprio l'effetto di negare il colore dello sfondo (che da blu diventa bianco) e il colore del testo (che da bianco diventa blu). Per togliere l'effetto reverse, è sufficiente richiamare nuovamente la stessa `lcd_rectfill()`, cambiando il colore da `COLOR_WHITE` a `COLOR_BLUE`.

Blitting

Passiamo ora a qualcosa di più evoluto rispetto alle semplici linee e rettangoli, cioè alle immagini: come fare per disegnarne una sul display grafico basato sul nostro amico T6963C?
A tale scopo, abbiamo creato una funzione chia-

Sistemi di Videosorveglianza

Vivi la tua casa in tranquillità.

WIRELESS



PREZZO IMBATTIBILE

€ 185,-

CAMSETW11N

rotazione: 320°

Tutti i prezzi si intendono IVA compresa.

Set A/V 2,4GHz con trasmettitore e ricevitore con display LCD

Funziona sulla banda dei 2,4GHz, comprende un ricevitore A/V a 3 canali con monitor TFT LCD da 2,5" e una telecamera CMOS a colori con audio. La telecamera è orientabile manualmente (320° sull'asse orizzontale) ed è dotata di sensore PIR che attiva un segnale acustico sul ricevitore nel caso rilevi il passaggio di persone o animali domestici. Dispone, inoltre, di 8 LED infrarossi che consentono riprese anche in condizioni di buio assoluto fino ad una distanza massima di 5 metri. Sia la telecamera che il ricevitore possono essere alimentati mediante batterie (anche ricaricabili) oppure tramite gli appositi adattatori di rete forniti in dotazione. Adatto per essere utilizzato in abitazioni private, uffici e piccole imprese. Ideale per avere sempre sotto controllo bambini ed anziani. Eccezionale rapporto prezzo/prestazioni.

Ricevitore con monitor incorporato

Display: 2.5" TFT LCD; risoluzione: 480 x 284 pixels; consumo di corrente: ~ 800mA; dimensioni: 70 x 183 x 24 mm; alimentazione: mediante adattatore di rete (incluso), 4 batterie alcaline tipo AAA (non incluse), o pacco batterie ricaricabili (non incluso).

Telecamera con trasmettitore

Sistema TV: PAL; ottica: f=4.3mm, F1.8 fuoco fisso; 8 LED IR: risoluzione: 360 linee TV; sensore: colore CMOS 1/3"; consumo corrente: ~ 120mA; portata: 60 metri circa; dimensioni: 58 x 185 x 97 mm; alimentazione: mediante adattatore di rete (incluso), 4 batterie alcaline tipo AAA (non incluse), o pacco batterie ricaricabili (non incluso).

Disponibile separatamente telecamera supplementare (cod. CAMW6) al prezzo di euro 70,00.

Set A/V con trasmettitore a tenuta stagna

CP292

€ 86,-



Set composto da una telecamera a colori con microfono incorporato e trasmettitore A/V a 2,4GHz. La telecamera può essere collocata all'esterno in quanto utilizza un contenitore a tenuta stagna. Il set di videosorveglianza comprende anche il ricevitore e tutti gli accessori. L'illuminatore IR a 30 LED, che entra automaticamente in funzione in presenza di scarsa luminosità, consente riprese al buio ad una distanza di oltre 10 metri.

Telecamera con trasmettitore: elemento sensibile: CMOS 1/3" PAL; pixel totali: 628 x 582; sensibilità: 3 Lux/F1.2 (0 Lux IR ON); risoluzione orizzontale: 380 linee TV; tensione di alimentazione: +8Vdc; potenza RF: 10mW; assorbimento: 80mA (250mA IR ON); dimensioni: 50 x 60 x 45mm; portata indicativa: 50-100m.

Ricevitore: frequenza di funzionamento: 2400~2483 MHz (CH1= 2,414GHz CH2=2,432GHz CH3=2,450GHz CH4=2,468GHz); impedenza di antenna: 50 ohm; uscite video: 2; uscite video: 1 Vpp/75 ohm; uscita audio: 200mVpp 10 Kohm; tensione di alimentazione: 12Vdc; consumo: 2W; connettore antenna: SMA; dimensioni: 96 x 79 x 30mm.

Set A/V con telecamera CMOS

€ 69,-



CP293

Sistema di videosorveglianza senza fili operante sulla banda dei 2,4 GHz composto da una piccola telecamera CMOS a colori con audio e trasmettitore A/V e da un ricevitore a quattro canali con selettore a slitta. Il set comprende sia l'adattatore di rete per la telecamera che per il ricevitore.

Telecamera con trasmettitore: Elemento sensibile: CMOS 1/3" OMNIVISION PAL; ottica: f=3,6mm F2.0; sensibilità: 3 Lux/F1.2; risoluzione orizzontale: 380 linee TV; tensione di alimentazione: +8Vdc; potenza RF: 10mW; assorbimento: 80mA; dimensioni staffa inclusa: 35 x 57 x 35mm; portata indicativa: 30 -50 metri.

Ricevitore: Numero canali: 4; frequenza di funzionamento: 2400~2483 MHz; impedenza di antenna: 50 ohm; 2 uscite video: 1 Vpp/75 ohm; 2 uscite audio: 2 Vpp (max); tensione di alimentazione: 12Vdc; assorbimento: 130mA; connettore antenna: SMA; dimensioni: 105 x 85 x 30mm.

Set con telecamera CMOS PIN-HOLE e ricevitore

CP294

€ 70,-



Sistema di videosorveglianza senza fili operante sulla banda dei 2,4 GHz composto da una piccola telecamera CMOS a colori con audio e trasmettitore A/V e da un ricevitore a quattro canali con selettore a slitta. Il set comprende sia l'adattatore di rete per la telecamera che per il ricevitore.

Telecamera con trasmettitore A/V: elemento sensibile: CMOS 1/3" OMNIVISION PAL; ottica: f=4,3 mm F2.0; sensibilità: 3 Lux/ F1.2; risoluzione orizzontale: 380 linee TV; tensione di alimentazione: +8Vdc; potenza RF: 10mW; assorbimento: 80mA; dimensioni: 20 x 20 x 20 mm; portata indicativa: 30-50 metri.

Ricevitore: Numero canali: 4; frequenza di funzionamento: 2400~2483 MHz; impedenza di antenna: 50 ohm; 2 uscite video: 1 Vpp/75 ohm; 2 uscite audio: 2 Vpp (max); tensione di alimentazione: 12Vdc; assorbimento: 130mA; connettore antenna: SMA; dimensioni: 105 x 85 x 30mm.

Set con 4 telecamere CMOS con IR



Sistema composto da 4 telecamere CMOS con trasmettitore A/V a 2,4 GHz e da un ricevitore a 4 canali con telecomando ad infrarossi. Le telecamere possono essere utilizzate all'esterno in quanto dotate di contenitore a tenuta stagna. L'illuminatore IR a 30 LED, che entra automaticamente in funzione in presenza di scarsa luminosità, consente riprese al buio ad una distanza di oltre 10 metri.

Telecamera con trasmettitore A/V: elemento sensibile: CMOS 1/3" OMNIVISION PAL; ottica: f=3,6mm F2.0; apertura angolare: 92°; sensibilità: 3 Lux / F1.2; risoluzione orizzontale: 380 linee TV; tensione di alimentazione: +12Vdc; potenza RF: 10 mW; assorbimento: 110 mA (130 mA con illuminatore); dimensioni staffa inclusa: 55 x 130 x 55mm; portata indicativa: 30-50 metri.

Ricevitore: numero canali: 4; frequenza di funzionamento: 2400~2483 MH; 2 uscite video: 1 Vpp/75 ohm; 2 uscite audio: 2 Vpp (max); tensione di alimentazione: 12Vdc; assorbimento: 130mA; connettore antenna: SMA; dimensioni: 120 x 100 x 30mm.

CP295

€ 240,-

Set con 4 telecamere CMOS e ricevitore 4 uscite

CP326



€ 260,-

Sistema di videosorveglianza senza fili composto da 4 telecamere CMOS con trasmettitore A/V a 2,4 GHz e da un ricevitore a 4 canali. Quest'ultimo dispone di quattro uscite A/V separate (1 x telecamera) e di un'uscita (AUTO) con funzione switcher. Le 4 uscite A/V separate consentono di gestire le telecamere singolarmente permettendo la visualizzazione, su altrettanti monitor, delle immagini riprese. L'uscita switcher permette di visualizzare in sequenza i 4 canali (tempo di commutazione 5 secondi). Le telecamere possono essere utilizzate all'esterno in quanto dotate di contenitore a tenuta stagna. L'illuminatore IR a 12 LED, che entra automaticamente in funzione in presenza di scarsa luminosità, consente riprese al buio ad una distanza di circa 8m.

Ultraminiatura

Camera Pen a 2,4GHz



FR225

€ 290,-

Sistema via radio a 2,4 GHz composto da un ricevitore, da una microtelecamera a colori e da un microtrasmettitore audio/video inseriti all'interno di una vera penna. Possibilità di scegliere tra 4 differenti canali. Ricevitore completo di alimentatore da rete. La confezione comprende i seguenti componenti:

Wireless Pen Camera: Wireless Pen Camera; 15 batterie LR 44; cilindretto metallico da usare con adattatore per batterie da 9 Volt; cavo adattatore per batterie da 9V.

Ricevitore Audio/Video: Ricevitore AV; alimentatore da rete; cavo RCA audio/video.

Microtelecamera TX/RX A/V a 2,4GHz



FR163

€ 210,-

Microscopica telecamera CMOS a colori (18 x 34 x 20mm) con incorporato microtrasmettitore video a 2430 MHz e microfono ad alta sensibilità. Potenza di trasmissione 10 mW; Risoluzione telecamera 380 linee TV; ottica 1/3" f=5,6mm; Apertura angolare: 60°; Alimentazione da 5 a 12 Vdc; Assorbimento: 80 mA. La telecamera viene fornita con un portabatterie stilo e un ricevitore a 2430 MHz (dimensioni: 150 x 88 x 44mm) completo di alimentatore da rete e cavi di collegamento.

Set RTX A/V 2,4 GHZ

AVMOD15



€ 78,-

Sistema wireless operante sulla banda dei 2,4 GHz composto da un trasmettitore e da un ricevitore Audio/Video. L'unità TX permette la trasmissione a distanza di immagini e suoni provenienti da un ricevitore satellitare, da un lettore DVD, da un videoregistratore o da un impianto stereo, verso un televisore collegato all'unità RX posizionato in un altra stanza. Il sistema dispone anche di un ripetitore per telecomando IR che consente di controllare a distanza il funzionamento del dispositivo remoto, ad esempio per cambiare i canali del ricevitore satellitare, per inviare dei comandi al lettore DVD o per sintonizzare l'impianto stereo sull'emittente radiofonica preferita. Il set comprende l'unità trasmittente, quella ricevente, i due alimentatori da rete ed il ripetitore di telecomando ad infrarossi. Alimentazione: 9 VDC / 300mA (2 adattatori AC/DC inclusi).

Baby Monitor A/V

CAMSETW4



€ 141,-

Sistema per ambienti domestici composto da un trasmettitore radio completo di telecamera con pan/tilt e microfono e da un ricevitore con altoparlante incorporato ed uscita video da collegare a qualsiasi monitor, TV, ecc. Si installa facilmente in qualsiasi ambiente e può operare sia con alimentatore da rete che a batteria. Alimentazione: 7,5Vdc/ 500mA (alimentatore compreso) o a batterie: 2 x AA. Portata: circa 50m; Frequenza: 2,4GHz.

Trasmettitore con telecamera: Sensore: 1/3" CMOS colori; risoluzione orizzontale: 330 linee TV; luminosità: 0 Lux; Canali radio: 3; dimensioni: 75 x 33 x 122mm.

Ricevitore: Canali radio: 3; Uscita video: 1Vp-p / 75 ohm (RCA); Uscita audio: 1 Vp-p / 600 ohm; Risoluzione orizzontale: 480 x 234; Dimensioni: 75 x 33 x 130mm.

Set 2,4GHz con telecamera e monitor B/N



Sistema senza fili per impiego domestico composto da una telecamera con microfono incorporato e trasmettitore audio/video a 2,4 GHz e da un monitor in bianco/nero da 5,5" completo di ricevitore. Portata massima del sistema 25/100m, telecamera con illuminatore ad infrarossi per una visione al buio fino a 3 metri di distanza.

Monitor con ricevitore: Alimentazione DC: 13.5V/1200mA (adattatore incluso); Sistema video: CCIR; 4 CH radio; Risoluzione video: 250 (V) /300 (H) linee TV.

Telecamera con trasmettitore: Alimentazione DC: 12V/300 mA (adattatore incluso); Sistema video: CCIR; Sensore 1/4" CMOS; Risoluzione 240 Linee TV; Sensibilità 2 Lux (0,1Lux con IR ON); Microfono incorporato.

CAMSETW2 Set ciompleto(Telecamera+monitor) - Euro 120,00

CAMW2 (Telecamera wireless supplementare) - Euro 70,00

€ 120,-

FUTURA ELETTRONICA

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

Codice MIP **268083**

mata `lcd_bitblt()`. Il nome potrebbe essere noto a qualcuno di voi e richiama le funzioni grafiche usate nei primi personal computer e ancora oggi presenti (Windows ha una sua funzione API GDI che si chiama proprio `bitblt()`). Essa sta per Bit BLock Transfer, cioè trasferimento di blocchi di bit. Naturalmente il trasferimento avviene tipicamente dalla memoria di sistema alla memoria video.
La nostra funzione è mostrata nel listato 7.
Essa è molto semplice e si limita ad utilizzare la funzione Autowrite per trasferire il contenuto dalla Flash del microcontrollore alla VRAM del display.
Alla funzione è necessario passare il pulsore all'array di costante di byte (quindi memorizzaato in Flash) che contiene l'immagine, le coordinate (x,y) del vertice in alto a sinistra, la larghezza dell'immagine da trasferire e l'altezza, tutti valori espressi in pixel. Per semplificare questa operazione, che sarebbe in realtà molto più complessa, facciamo l'ipotesi che la coordinata x sia allineata al byte, cioè sia un multiplo intero di 8. Con questa semplificazione, non dobbiamo fare altro che copiare, byte dopo byte, il valore da una memoria all'altra mediante Autowrite, stando attenti soltanto all'ultimo byte che potrebbe essere solo parzialmente modificato.
Se l'ultimo byte è parziale, bisogna leggere il valore corrente dalla VRAM e, mediante maschera di bit, sostituire i bit più significativi con quelli dell'immagine da disegnare. La sostituzione viene fatta azzerando i bit da modificare mediante operazione di AND con una maschera opportuna e, successivamente, mediante operazione di OR con il byte dell'immagine.
Se la coordinata x non fosse multiplo di 8, il compito della funzione si complicherebbe notevolmente, poiché i byte nella Flash non corrisponderebbero esattamente ai byte in VRAM, ma sarebbero "a cavallo" di due byte.
Per gestire questa situazione, bisognerebbe lavorare molto con maschere e shift logici, cosa che, almeno per ora, ci risparmiamo.

Listato 7

Funzione per il disegno di immagini lcd_bitblt() [demograph.c]

```
void
lcd_bitblt( const unsigned char *img, unsigned char x, unsigned char y, unsigned char w,
unsigned char h )
{
  /* Copia la bitmap img[] nella VRAM, a partire dal punto in alto a sinistra
   * di coordinate (x,y). La larghezza della bitmap è x, l'altezza è y.
   * Per semplicità, i parametri hanno le seguenti limitazioni:
   * - x deve essere multiplo di 8 (allineato al byte della VRAM)
  */
  unsigned int addr = GRAPHIC_HOME + GRAPHIC_AREA*y + (x>>3);
  unsigned char b = w>>3;      /* Numero di byte interi da modificare. */
  unsigned char d, mask;

  while( h— ) {
    tlcd_addrptrset( addr );
    tlcd_autowrite( img, b );
    img += b;
    if( w&0x07 ) {
      /* L'ultimo byte deve essere parzialmente modificato. */
      d = tlcd_dataread( TLCD_AUTOUNCH );
      mask = maskbit[8-(w&0x07)];
      d &= mask;
      d |= *img++&~mask;
      tlcd_datawrite( d, TLCD_AUTOINCR );
    }
    /* Passa al rigo successivo. */
    addr += GRAPHIC_AREA;
    }
}
```

Applicazione di test

Per dimostrarvi la validità delle funzioni precedenti, abbiamo sviluppato un firmware per il Demograph che mostra il funzionamento di quanto abbiamo detto finora. Potete scaricare l'intero progetto MikroC (sorgente ed eseguibile) dalla pagina dedicata a questo articolo sul sito di Fare Elettronica. Il file da scaricare è test_rect.zip.

Il funzionamento è molto semplice: ogni tasto del Demograph è associato ad un test diverso. In totale ci sono 4 test che è possibile visualizzare.

Il primo test disegna la classica griglia del gioco del tris, mediante `lcd_hline()` e `lcd_vline()`.

Il secondo test disegna una scacchiera, utilizzando le `lcd_rect()` e `lcd_rectfill()`.

Il terzo test scrive le voci di un ipotetico menu, evidenziando la seconda opzione mediante `lcd_rectfill()` e abilitazione di testo e grafica in modalità XOR.

Il quarto test, infine, utilizza la `lcd_bitblt()` per disegnare il logo della rivista Fare Elettronica, memorizzata all'interno di un array costante di byte.

Come al solito, programmate il file .hex con uno dei metodi che preferite, stando attenti ai bit di configurazione.

Album fotografico

Chi ha realizzato la scheda di sviluppo Demograph proposta, non deve dimenticare che non ha nelle mani un circuito che permette *soltanto* l'utilizzo di un display grafico, ma una scheda completa di EEPROM esterna al microcontrollore, RS232, porta I^2C verso l'esterno, ecc. Per dimostrare la validità di questa scheda, che, seppur molto economica, può fare concorrenza alle altre ben più blasonate in commercio, sviluppiamo una simpatica applicazione: un album fotografico.

In pratica, mediante la connessione RS232 con un computer, memorizziamo nella EEPROM esterna al microcontrollore una serie di immagini che verranno visualizzate, una dopo l'altra, sul display grafico. Il vantaggio di memorizzare le immagini nella EEPROM risiede nel fatto che, essendo una memoria non volatile, alla riaccensione della scheda verrano visualizzate nuovamente le immagine senza la necessità del collegamento al computer: in pratica, diventa a tutti gli effetti un semplice visualizzatore di immagini stand-alone.

Quante immagini possiamo memorizzare nella EEPROM? Ovviamente dipende dalla dimensione della memoria e dallo spazio occupato da una singola immagine.

A quest'ultima domanda possiamo immediatamente rispondere: 128x64=8192 bit=1024 byte=1KB, nel caso di display 128x64. La dimensione della EEPROM usata (24LC512) è di 512Kbit=64KB, cosa che ci permette di immagazzinare fino a 64 immagini.

Non dimentichiamo che anche il microcontrollore ha una sua piccola EEPROM interna che può essere utilizzata per memorizzare alcune impostazioni: durata di visualizzazione di una immagine prima di passare alla successiva, numero di immagini da visualizzare, eventuale transizione da una immagine all'altra, modalità di visualizzazione (seriale o casuale), ecc.

Scaricamento delle immagini

Per l'invio delle immagini utilizziamo ovviamente la connessione RS232, già presente sul Demograph. In una versione avanzata della scheda, si potrebbe pensare di sostituire la vetusta RS232 con la nuova interfaccia USB.

Naturalmente dovremo creare un apposito software sul computer che si occuperà di caricare l'immagine e inviarla, secondo un opportuno protocollo, al Demograph. Dal canto suo, il Demograph dovrà



Codice MIP 268085

Listato 8

Funzione per il disegno di immagini da EEPROM esterna lcd_eebitblt() [demograph.c]

```
/* Scrive 64 righe, leggendole dalla
EEPROM esterna. */
for( row=0; row<64; row++ ) {
  /* Legge la riga dalla EEPROM esterna.
*/
  i2c_eepromread( eeaddr, buf, 16 );
  eeaddr += 16;
  /* Scrive la riga in VRAM. */
  tlcd_addrptrset( lcdaddr );
  tlcd_autowrite_ram( buf, 16 );
  lcdaddr += GRAPHIC_AREA;
}
```

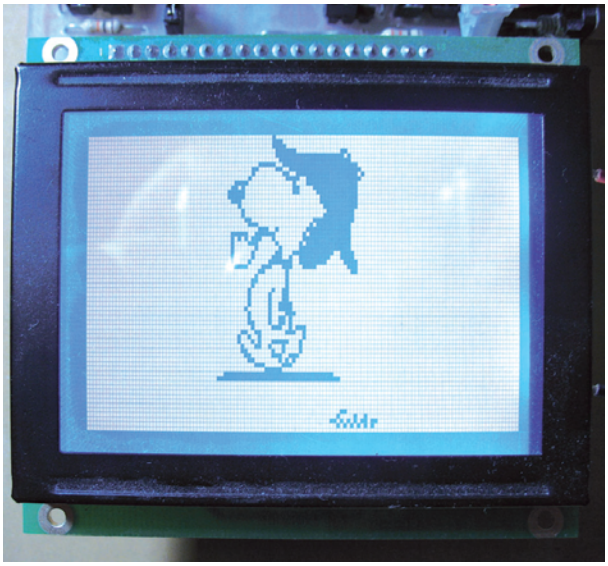


Figura 5
Display del Demograph in modalità album fotografico

ricevere l'immagine dalla RS232 e memorizzarla nella EEPROM esterna. In una versione avanzata della scheda di sviluppo, si potrebbe pensare di sostituire la EEPROM con una memory card (MMC, SD o simili). Per evitare di conservare l'intera immagine nella RAM del microcontrollore (il PIC16F877A ha solo 368 byte di memoria) e successivamente inviarla alla EEPROM, abbiamo pensato di inviare una riga per volta. Con un display da 128x64, si tratta di inviare 128/8=16 byte, valore molto più semplice da gestire con una piccola RAM. In pratica, il computer invia un comando SET IMAGE e il Demograph risponde subito con un ACK. Successivamente il computer invia 18 byte: numero di immagine (da 1 a 64), numero di riga (da 1 a 64) e i 16 byte che rappresentano la riga dell'immagine da inviare. Il Demograph, dopo aver salvato questi dati nella EEPROM, risponde con un ACK e il computer può inviare la riga successiva. Per rendere più robusta la comunicazione RS232 ed evitare di bloccare il programma in attesa di byte che non arriveranno mai, è stato implementato un meccanismo a timeout mediante il Timer0 a 8 bit. Ad ogni byte ricevuto, il timer viene azzerato. Se non viene ricevuto alcun byte prima che il registro associato al timer raggiunga il suo valore massimo, il comando viene ignorato.



Figura 4
Screenshot del software per gestire l'album fotografico

Visualizzazione delle immagini

La visualizzazione delle immagini memorizzate nella EEPROM sembrerebbe molto semplice utilizzando la funzione `lcd_bitblt()`. Purtroppo questa funzione accetta soltanto un array costante di byte, cioè un array che risiede nella Flash del PIC. La nostra immagine, invece, risiede nella EEPROM esterna al microcontrollore. Non sarebbe neanche fattibile copiare temporaneamente tutta l'immagine dalla EEPROM esterna nella RAM e da lì copiarla nella VRAM, per la solita limitazione sulla dimensione della RAM del PIC.

Anche in questo caso dobbiamo procedere per righe (si veda il listato 8).

Ogni riga viene letta dalla EEPROM esterna con la funzione `i2c_eepromread()` (presente nel file demograph.c) e copiata nell'array `buf[]`. Successivamente viene trasferita nella VRAM del display mediante Autowrite richiamando la funzione `tlcd_autowrite_ram()`, identica alla `tlcd_autowrite()` sviluppata nel precedente articolo a parte il fatto che legge i byte da trasferire dalla RAM piuttosto che dalla Flash.

La pausa tra il disegno di una immagine e l'altra è gestita tramite il Timer1 a 16 bit, abilitando l'interrupt relativo. Nella routine dell'interrupt (ISR), il registro del timer viene caricato in modo da far "scattare" nuovamente l'interrupt dopo circa 0.1 secondi (100ms). Mediante la variabile `tmrimg`, decrementata di una unità nell'ISR, è possibile creare dei tempi variabili multipli di 100ms. Questo intervallo di tempo, cioè quante volte deve

scattare l'ISR, è memorizzato nella EEPROM *interna* del PIC e letto all'accensione. È possibile variare questo valore mediante un apposito comando via RS232.
Nella EEPROM interna è memorizzato anche il numero totale delle immagini che devono essere visualizzate. Anche questa informazione può essere variata via RS232 mediante apposito comando.
Per i dettagli, ovviamente fate riferimento al sorgente che potete scaricare, insieme all'eseguibile per la programmazione del PIC, dal sito di Fare Elettronica.

Il software

Il software per computer che comunica con il Demograph è stato sviluppato in Visual Basic 6.0. Potete scaricare il sorgente e il pacchetto di installazione di questo programma sempre dal sito di Fare Elettronica alla pagina relativa a questo articolo.
All'esecuzione compare una finestra mediante la quale è possibile:

- Scegliere la porta seriale a cui è collegato il Demograph;
- Caricare una immagine da file (tasto *Apri immagine*);
- Inviare l'immagine caricata al Demograph (tasto *Scrivi immagine*);
- Impostare il timeout tra una immagine e l'altra (tasto *Imposta tempo*);
- Impostare il numero di immagini da visualizzare (tasto *Imposta n. immagini*).

Ovviamente è possibile aprire file che contengono immagini di dimensioni pari a 128x64 pixel. Essendo il display a due colori, anche l'immagine dovrebbe essere in bianco e nero. Se così non fosse, tutti i pixel neri vengono considerati pixel da accendere sul display del Demograph, tutti gli altri vengono considerati pixel da spegnere. Cliccando sul tasto *Scrivi immagine*, sarà richiesto il numero dell'immagine (da 1 a 64) da scrivere.
Con i tasti *Imposta tempo* e *Imposta n. immagini* è possibile impostare l'intervallo di tempo tra una immagine e l'altra (da 1 a 100s) e il numero totale delle immagini (da 1 a 64) da visualizzare.
In Figura 4 è presente la finestra del programma dopo aver caricato una immagine pronta per essere trasferita nel Demograph, in Figura 5 è mostrato il nostro album fotografico in azione!
Impostando un timeout tra una immagine e l'altra molto piccolo e delle immagini provenienti da un filmato, è possibile creare anche semplici effetti di animazione.

Conclusioni

Con questo termina la serie di articoli sui display a cristalli liquidi. Abbiamo percorso velocemente un po' tutto quello che c'è da sapere su questi gioielli della tecnologia, a partire dalla loro storia per finire agli esempi più pratici.
Ovviamente non pensiamo di aver trattato esaurientemente tutti gli aspetti, ma di aver dato almeno un punto di partenza agli appassionati che vorranno continuare lo studio e gli esperimenti.
Ci potrebbero essere numerose possibilità di approfondimento. Per esempio è possibile studiare gli altri tipi di controller grafici come i KS0107 e KS0108 della Samsung, anch'essi molto diffusi. Un altro "simpatico" controller grafico è quello utilizzato nei display LCD dei telefoni cellulari Nokia 3310 e 3330, ormai andati in disuso, ma che possono essere riutilizzati almeno per i nostri esperimenti elettronici.

Il passo naturale successivo è quello di pensare ai display LCD a colori: CSTN o TFT. Come già sappiamo, i primi sono molto più semplici da gestire e possono essere trovati con un controller integrato. I secondi devono essere pilotati in modo molto più complicato.
L'altra possibilità è quella di modificare il Demograph, sostituendo la porta RS232 con una USB e la EEPROM esterna con una memory card.
Per qualsiasi domanda, dubbio, critica ecc., potete contattarmi tramite la redazione.
Desidero ringraziare per la preziosa collaborazione tecnica il mio amico Pino che, tra l'altro, ha realizzato il prototipo del Demograph, e Mariangela che ha reso comprensibili molte mie frasi sconnesse.

Riferimenti utili

Articolo originale di Bresenham (in inglese):
http://www.research.ibm.com/journal/sj/041/ibmsjIVRIC.pdf
Descrizione dell'algoritmo di Bresenham su Wikipedia:
http://it.wikipedia.org/wiki/Algoritmo_della_linea_di_Bresenham

More Info Please!

Inserisci il Codice **268074** alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

Terza parte: *Settembre 2007*
Introduzione allo Spin

Quarta parte: *Ottobre 2007*
Esempi di programmazione in Spin

Quinta parte: *Novembre 2007*
Gestione dei gamepad e dell'audio

Sesta parte: *Dicembre 2007*
Esempio di utilizzo dell'assembler

Videogames by Example

In questa puntata verrà mostrato come creare con Hydra immagini e grafica da visualizzare su TV utilizzando lo Spin. In particolare verrà spiegato come utilizzare i driver video e grafico per ottenere in maniera estremamente semplice immagini ed animazioni complesse.

La generazione del video

Una delle (tante) caratteristiche che distingue il Propeller da altri microcontrollori è la sua capacità di generare segnali video senza bisogno di utilizzare alcun hardware aggiuntivo. Proprio questa caratteristica lo rende adatto a realizzare videogame o comunque sistemi multimediali. La generazione del segnale video e delle relative immagini ed animazioni costituisce il nucleo di qualsiasi videogame, e per questo motivo si è scelto di iniziare dagli esempi relativi a questa possibilità come particolare esempio di utilizzo del linguaggio Spin. A partire dagli esempi mostrati infatti non dovrebbe risultare difficile costruire semplici videogame, efficaci "demo", interfacce grafiche o presentazioni multimediali di vario genere.

Per potere comprendere a pieno gli esempi, e per poterli eventualmente modificare a piacimento, è necessaria una introduzione al modo in cui il Propeller genera i segnali video, ed al funzionamento dei driver TV e grafico che saranno utilizzati nei programmi.

Come accennato nelle scorse puntate, il Propeller non possiede un'unità grafica vera e propria (come una GPU o un RAMDAC), ma genera il video utilizzando delle semplicissime periferiche chiamate Video Streaming Unit (VSU), presenti in ogni COG, che possono produrre dei segnali ad alta frequenza ed ampia banda, ma che devono essere controllate da un firmware apposito per generare un segnale video vero e proprio (composto da sincronismi, burst di colore, segnale di luminanza e crominanza). Si potrebbe dire quindi che il Propeller impiega una unità grafica "virtuale", prevalentemente software, e quindi completamente ridefinibile dall'utente.

Ciascuna VSU è in pratica un sofisticato registro a scorrimento, che prende in ingresso i dati su una sequenza di "livelli video" da trasmettere e li emette da uno dei pin di I/O alla velocità selezionata (che dipende dallo standard televisivo scelto). Ciascuno dei "livelli video" in questione può essere una combinazione arbitraria di un livello di luminosità, di uno di crominanza (che viene opportunamente modulato in fase e quadratura) e della portante colore. Dalla combinazione di questi elementi e dalla loro corretta temporizzazione è possibile generare tutti gli elementi di un segnale video standard (ad esempio PAL o NTSC).

Queste ultime operazioni vengono gestite dal driver video software.

Questo funzionamento potrebbero sembrare un po' complesso e poco pratico da utilizzare, per fortuna però normalmente non è necessario preoccuparsi di questi dettagli, perché è sufficiente utilizzare il driver, che mette a disposizione un'interfaccia molto più semplice ed immediata. L'unico dettaglio che è utile conoscere è il modo in cui vengono rappresentati i pixel ed i colori per le VSU. I dati video sono inviati sequenzialmente alle VSU come due long (32 bit): la prima rappresenta i dati della bitmap (cioè i pixel ed i rispettivi colori della tavolozza), la seconda specifica invece quali sono i colori indicati nella prima word. Il formato della prima long è molto semplice: nella modalità ad 1 bit di colore (due colori possibili), ciascun bit rappresenta il colore di un pixel. In altri termini i 32 bit specificano una "striscia" di 32 pixel consecutivi che possono avere i colori indicati come 0 o 1 (uno dei due colori viene usato come sfondo, l'altro per il disegno). Nella modalità a 2 bit di colore (4 colori), la long specifica il colore di 16 pixel consecutivi, dal momento che per ciascun pixel vengono utiliz-

Esempi di programmazione in Spin



di Antonio Di Stefano

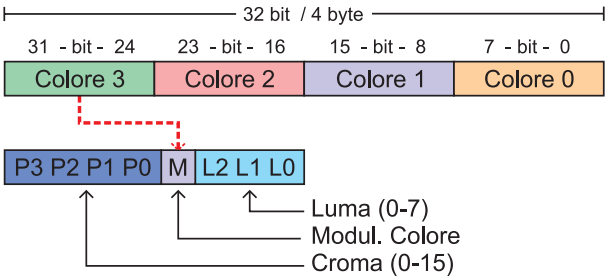


Figura 1
Codifica dei 32 bit di colore delle VSU

zati due bit, che indicano i colori di indice da 0 a 3. La long relativa ai colori ha il formato indicato in Figura 1: essa specifica i 4 colori corrispondenti agli indici da 0 a 3 (o solo i primi due nel caso si utilizzi grafica ad un bit). Ciascun colore è specificato utilizzando 8 bit: i 3 più bassi specificano la luminosità, i 4 più alti la crominanza (cioè la particolare tonalità di colore), mentre il bit 3 attiva o meno la portante colore. I diversi colori possono essere ottenuti come indicato in Figura 2. Va notato che la luminosità può variare tra circa 2 e 6 in generale, perché l'attivazione della portante colore causa un'oscillazione di +-1, che potrebbe portare a valori vicini a 0, interpretati come sincronismo. Spesso tuttavia è possibile utilizzare il valore 0 assieme alla portante colore per ottenere colori addizionali e "super saturi". Questa descrizione potrebbe trarre in inganno: i colori effettivamente utilizzabili in un'immagine non sono solo 4, ma molti di più! Per ogni gruppo di 16 pixel infatti è possibile specificare 4 colori diversi ed indipendenti dagli altri. Questo meccanismo, anche se apparentemente scomodo, permette di ottenere immagini dotate di molti colori e sfumature.

Maggiori dettagli sulla programmazione delle VSU possono essere trovati sul manuale di Hydra, queste informazioni sono comunque sufficienti per comprendere ed utilizzare i driver video e grafico.

Come già detto la generazione dei segnali video veri e propri avviene a più livelli: le VSU generano il segnale "fisico", il driver TV genera il segnale video organizzando opportunamente i dati per le VSU ed i tempi, ed il driver grafico infine crea e fornisce i dati della bitmap al driver TV, gestendo funzioni più sofisticate per la creazione di primitive e funzioni grafiche (punti, linee, archi, sprite, double buffer etc...). L'utente è libero di interfacciarsi con il livello che ritiene più opportuno: può scrivere il proprio driver video, può creare e fornire i dati relativi alla bitmap al driver TV, o può utilizzare direttamente il driver grafico per creare immagini con funzioni ad alto livello. Ovviamente i livelli più bassi offrono prestazioni migliori in termini di velocità, ma risultano più laboriosi da utilizzare, e viceversa. Di seguito verranno descritti i due driver e le funzioni di cui dispongono.

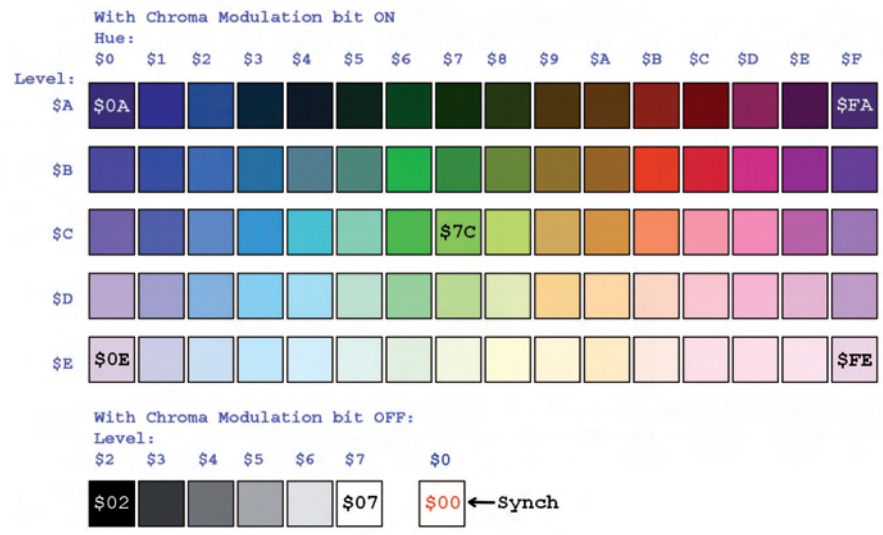


Figura 2
Codici dei colori

Il driver TV

Il driver TV è costituito dal file "TV_DRV_010.spin" che deve essere richiamato nella propria applicazione come oggetto. Si tratta di un driver di livello molto basso, che essenzialmente si occupa di inviare alla VSU i dati di una bitmap letti dalla RAM condivisa con le temporizzazioni

appropriate in modo da generare un segnale video PAL o NTSC. Il diver svolge soltanto questa funzione (quella di "costruire" il segnale video a partire dalla bitmap), ma dispone di moltissimi parametri che consentono di personalizzarne li funzionamento. Questi parametri devono essere tutti specificati prima di avviare il driver stesso, quindi i programmi che richiamano il driver devono includere la dichiarazione e l'inizializzazione di alcuni dati. Una parte dei parametri sono specificati come costanti, come riportati nel listato 1.

Queste costanti si riferiscono alle temporizzazioni PAL ed NTSC (che non vanno modificati nel tempo in genere), al numero di parametri passati come variabili, ed all'indirizzo in cui si vuole posizionare la tavola dei colori nella memoria del COG che esegue il driver. I parametri espressi come variabili sono invece costituiti da 14 long che contengono informazioni quali la larghezza dello schermo, gli indirizzi della memoria in cui risiede la bitmap e la tabella dei colori, o le coordinate di riferimento sullo schermo, più alcuni altri valori un po' più criptici. Questi parametri sono mostrati e brevemente commentati di seguito, ma si rimanda al manuale o al sorgente del driver per una descrizione più dettagliata. Anche in questo caso infatti si usano dei valori tipici, che sono quelli mostrati nei listati. In diversi casi può essere utile modificare questi parametri durante l'esecuzione. Questo spiega perché sono stati dichiarati come variabili e non costanti. (Listato 2)

Una cosa interessante, che si può notare dalle dichiarazioni, è che il driver video funziona secondo una modalità a "tile" (a tasselli) e non a bitmap. Questo particolare può risultare estremamente comodo in molti casi, ma complica un po' il suo utilizzo. Prima di analizzare il modo in cui devono essere disposti i dati relativi alla grafica in memoria, è il caso di accennare brevemente alla differenza tra grafica bitmap e a tile. Il primo caso è il più

Listato 1

```
CON

  fntsc             = 3_579_545          'NTSC color frequency
  lntsc             = 3640               'NTSC color cycles per line * 16
  sntsc             = 624                'NTSC color cycles per sync * 16

  fpal              = 4_433_618          'PAL color frequency
  lpal              = 4540               'PAL color cycles per line * 16
  spal              = 848                'PAL color cycles per sync * 16

  paramcount        = 14
  colortable        = $180               'start of colortable inside cog
```

Listato 2

```
VAR                                    'TV parameters - 14 contiguous longs

 long          tv_status               '0/1/2 = off/invisible/visible
 long          tv_enable               '0/non-0 = off/on
 long          tv_pins                 '%pppmmmm = pin group, pin group mode
 long          tv_mode                 '%ccip = chroma, interlace, ntsc/pal
 long          tv_screen               'pointer to screen (words)
 long          tv_colors               'pointer to colors (longs)
 long          tv_ht                   'horizontal tiles
 long          tv_vt                   'vertical tiles
 long          tv_hx                   'horizontal tile expansion
 long          tv_vx                   'vertical tile expansion
 long          tv_ho                   'horizontal offset
 long          tv_vo                   'vertical offset
 long          tv_broadcast            'broadcast frequency (Hz)
 long          tv_auralcog             'aural fm cog
```

Da 0 a 30 anni! Baby, Kids, Tweens, Teens, College Students

Scopri, attraverso **32 interventi**, come raggiungere e conquistare tutti i segmenti di giovani consumatori!

Youth MARKETING FORUM '07

Iscrivendoti entro il **28 settembre** risparmi **200 euro!!**

WHAT'S COOL?

MARKETING PER BAMBINI E RAGAZZI

21 e 22 novembre 2007 - Milano, Hotel Melià

Con il contributo dei più importanti **protagonisti** dello **Youth Marketing italiano!!**

Codice MIP 268091

- 36 qualificati Relatori
- 32 interventi
- 4 giornate di approfondimento
- 2 workshop tematici

Tra gli altri, **SPECIFICI APPROFONDIMENTI** sui seguenti temi:

- Come costruire un LEGAME EMOZIONALE con il segmento KIDS
- Approfondire il rapporto tra BAMBINI e BRAND
- Come la MULTICULTURALITA' e il MULTILINGUISMO diventano elementi chiave di successo di una strategia di marketing e comunicazione rivolta al target kids
- Analisi dei consumi dei GIOVANI STRANIERI in Italia
- Nuove esperienze di DIGITAL e MOBILE TECHNOLOGY da offrire ai giovani
- Come sfruttare le nuove opportunità offerte da WEB PARTECIPATIVO e VIRTUAL WORLDS
- NUOVE SEGMENTAZIONI per lo Youth Marketing e nuovi ambiti di relazione per acquisire e fidelizzare
- Come attrarli e coinvolgerli sul PUNTO VENDITA: analisi, tecniche di comunicazione e strategie di SHOPPER MARKETING

... e in più partecipa a 2 Workshop innovativi:

Martedì 20 novembre

ADOTTARE IL MARKETING ETNICO: METODI, STRATEGIE, OPPORTUNITA' E CASI

Venerdì 23 novembre

MARGAYTING: CAPIRE E DIALOGARE CON IL SEGMENTO GAY

Inviare la scheda compilata a: **Istituto Internazionale di Ricerca - Via Forcella, 3 - 20144 Milano** www.iir-italy.it/youth
Tel. 0283847627 - Fax 0283847262 - Email: youth@iir-italy.it

❒ Sì, inviatemi il programma dettagliato del convegno: **YOUTH MARKETING FORUM 07**

Cod. FED3262

NOME:_____ COGNOME:_____

FUNZIONE:_____

SOCIETÀ:_____

TEL.:_____ FAX:_____ E-MAIL:_____

INDIRIZZO:_____

CITTÀ:_____ CAP:_____ PROV:_____

I dati saranno trattati dall'Istituto Internazionale di Ricerca secondo quanto previsto dal D.Lgs. 196/03 sulla tutela della Privacy.

SPONSOR

Affinion

A-Tono

StudentiMediaGroup

MEDIA PARTNER

AREA PRESS · CAMPUS · dmc · fare elettronica · infocommercio.it · MyMarketing.Net · nuova DISTRIBUZIONE · not · P5 · WIRELESS

INFORMAZIONI E ISCRIZIONI:
TEL. 02.83847627 FAX 02.83847262
EMAIL: YOUTH@IIR-ITALY.IT
WWW.IIR-ITALY.IT/YOUTH

Istituto Internazionale di Ricerca
1987-2007 • 20 anni in Italia

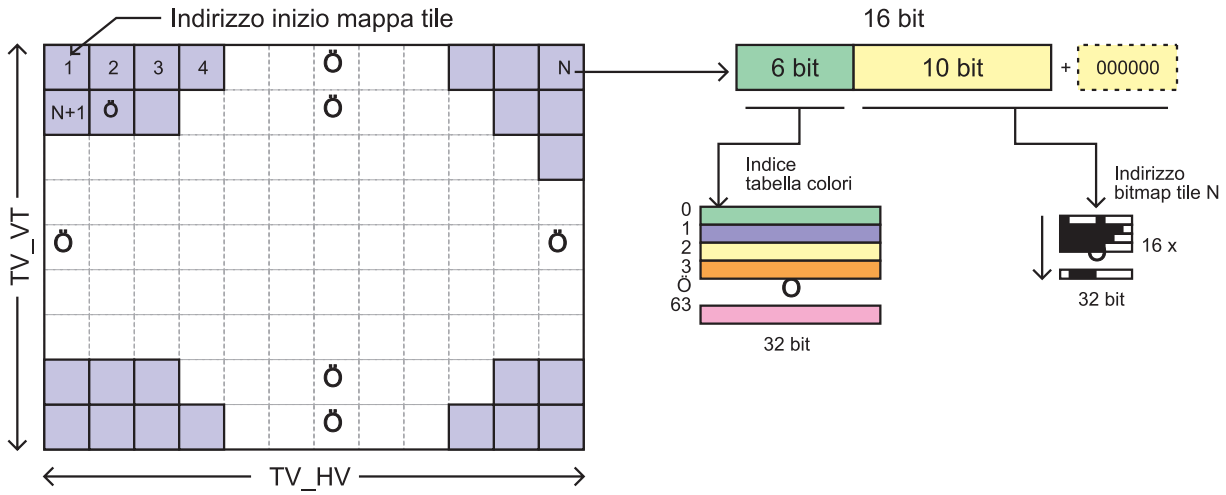


Figura 3
Organizzazione della memoria per il driver TV

intuitivo: in un'area di memoria contigua viene memorizzata l'immagine che si desidera visualizzare, specificando di seguito il colore di ogni pixel. Esiste quindi una corrispondenza diretta tra quello che si vede sullo schermo ed il contenuto della memoria. La grafica a tile è invece leggermente più complessa. Essa è stata (ed è ancora) molto utilizzata nel campo dei videogame e dei piccoli computer in quanto permette di risparmiare memoria e di ottenere una maggiore flessibilità e semplicità nella costruzione dell'immagine per determinate applicazioni. In pratica l'immagine da visualizzare è descritta non come una matrice di pixel, ma come una matrice di tasselli. I tasselli sono dei piccoli "pezzi" di immagine, composti da NxM pixel (16x16 in questo caso), che messi assieme costruiscono un'immagine più complessa. Si può avere un'idea immediata di questa tecnica ripensando all'aspetto di vecchi videogame quali Super Mario o altri platform: i livelli erano, per quanto vari, sempre composti da blocchetti elementari, disposti come in una griglia, che si ripetevano fino a costruire uno scenario complesso e vario. Anche la "modalità" testo usata sui computer può essere considerata un esempio di grafica a tile: in questo caso i tasselli sono proprio i caratteri. Quando si utilizza una modalità a tile, l'immagine viene specificata dalla cosiddetta "mappa dei tile", che non è altro che un'area di memoria dove è specificato quale tile deve essere visualizzato in ciascuna posizione della griglia NxM. Se ad esempio l'immagine ha dimensioni 256x192 pixel, e si utilizza una modalità bitmap, sarà necessario specificare il colore di tutti i 256x192 pixel. Se si utilizza una modalità a tile, con dimensioni dei tile di 16x16, occorrerà specificare soltanto quali tile usare per le 16x12 posizioni. Quindi occorreranno solo 192 valori invece di 49152! Ovviamente in quest'ultimo caso occorrerà descrivere in un'altra area di memoria l'aspetto dei singoli tile, come se fossero delle piccole bitmap da 16x16 pixel. Di solito i vari tile sono descritti in un'area della memoria separata. Dal momento che di solito i tile definiti sono relativamente pochi, si ottiene un notevole risparmio di memoria rispetto ad una modalità bitmap. Non solo, la grafica a tile presenta anche un ulteriore vantaggio: per modificare completamente un "quadro" è sufficiente cambiare pochi valori nella mappa dei tile, e non modificare molti pixel, ottenendo quindi una maggiore semplicità di gestione e velocità di esecuzione (si pensi ad esempio al gioco Arkanoid: ogni volta che la pallina urta un mattone, per farlo scomparire è sufficiente cambiare un solo valore nella mappa, e non ridisegnare l'intera area!). In questo modo si possono ottenere anche diversi "effetti speciali" molto interessanti.
Tornado al driver TV, esso impiega tile di dimensioni 16x16 pixel e prevede che venga specificata una mappa dei tile, e le bitmap dei tile. La mappa è costituita da M colonne (parametro TV_HV) e da N righe (TV_VT) disposte in memoria in maniera contigua e sequenziale (le singole righe sono specificate una dietro l'altra). Ciascun elemento della mappa è una word da 16 bit che contiene nei 6 bit più alti l'indice di uno dei 64 possibili indici della tabella dei colori, nei 10 più bassi il puntatore al descrittore del tile. La tabella dei colori, che si trova all'indirizzo specificato dal parametro TV_COLORS, è formata da 64 long, ciascuna delle quali specifica 4 colori secondo il formato descritto nel paragrafo precedente. Il tile può quindi essere disegnato con 4 colori indicati nella long specificata. I 10 bit del puntatore invece vengono estesi a 16 bit dal driver, aggiungendo 6 bit bassi posti a 0 (si ottengono così indirizzi multipli di 64, cioè la lunghezza della descrizione di un tile). Dalla locazione risultante inizia la descrizione del tile vero e proprio. Essa è formata da 16 long contigue, ciascuna delle quali

Listato 3

```
' *** Costanti ***
CON
 _clkmode = xtal2 + pll8x              ' clock ext. + PLL 8x
 _xinfreq = 10_000_000                 ' freq. ext. = 10MHz
 _stack = ($3000 + $3000 + 64) >> 2   ' spazio per stack
 ' numero parametric driver TV
 PARAMCOUNT         = 14
 ONSCREEN_BUFFER    = $5000            ' buffer video
 ' dimensioni tilemap e schermo
 X_TILES             = 16
 Y_TILES             = 12
 SCREEN_WIDTH       = 256
 SCREEN_HEIGHT      = 192
' *** Variabili ***
VAR
 long  tv_status      '0/1/2 = off/visible/invisible
 long  tv_enable      '0/? = off/on
 long  tv_pins        '%ppmmm = pins
 long  tv_mode        '%ccinp = chroma,interlace,ntsc/pal,swap
 long  tv_screen      'pointer to screen (words)
 long  tv_colors      'pointer to colors (longs)
 long  tv_hc          'horizontal cells
 long  tv_vc          'vertical cells
 long  tv_hx          'horizontal cell expansion
 long  tv_vx          'vertical cell expansion
 long  tv_ho          'horizontal offset
 long  tv_vo          'vertical offset
 long  tv_broadcast   'broadcast frequency (Hz)
 long  tv_auralcog    'aural fm cog
 word  screen[X_TILES * Y_TILES] ' mappa dei tile
 long  colors[64]                 ' tavola dei colori
' *** Oggetti ***
OBJ
 tv    : "tv_drv_010.spin"     ' richiama il driver TV
' *** Funzione ***
PUB start | i, dx, dy, x, y
  'avvia driver tv
  longmove(@tv_status, @tvparams, paramcount)
  tv_screen := @screen
  tv_colors := @colors
  tv.start(@tv_status)
  'init colori
  repeat i from 0 to 64
    colors[i] := $00001010 * (i+4) & $F + $FB060C02
  'init mappa tile
  repeat dx from 0 to tv_hc - 1
    repeat dy from 0 to tv_vc - 1
      screen[dy * tv_hc + dx] := onscreen_buffer >> 6 + dy +
                                 dx * tv_vc + ((dy & $3F) << 10)

' *** Dati binari ***
```

```
DAT
tvparams     long  0            'status
              long  1             'enable
              long  %011_0000    'pins
              long  %0000         'mode
              long  0             'screen
              long  0             'colors
              long  X_TILES      'hc
              long  Y_TILES      'vc
              long  10            'hx timing stretch
              long  1             'vx
              long  0             'ho
              long  0             'vo
              long  55_250_000   'broadcast: ch. 2 VHF
              long  0             'auralcog
```


specifica il colore dei 16 pixel di una riga del tile (vengono usati due bit per pixel, in modo da potere individuare i 4 colori). Questa organizzazione è riassunta in Figura 3.
Per richiamare il driver nelle proprie applicazioni è necessario dichiarare alcune costanti e variabili, come spiegato prima, richiamare il driver come oggetti esterno, ed avviarlo. Tutto questo è mostrato nel listato 3, che visualizza una piccola linea colorata in alto a sinistra.

Nella sezione CON sono state dichiarate alcune costanti di sistema per la scelta a l'impostazione del clock e dello stack, e altre relative al driver TV. La sezione VAR comprende le variabili relative al driver TV viste prima e due array (Screen e Colors) che costituiscono in pratica la mappa dei tile e la tavola dei colori. Nella sezione OBJ viene richiamato l'oggetto TV, che è appunto il driver. La funzione pubblica Start è quella che viene eseguita all'avvio. Si può notare anche che le variabili relative al driver viste in precedenza non sono state inizializzate. Per farlo, all'inizio della funzione Start vengono copiati i 14 valori elencati nella sezione DAT nell'area di memoria dove risiedono le variabili in questione, usando la funzione Longmove. Questa funzione riceve in ingresso il puntatore di destinazione (tv_status, che coincide con la prima variabile da inizializzare), quello sorgente (che è in pratica l'etichetta da cui iniziano i dati in questione nella sezione DAT), ed il numero di long da copiare, in questo caso 14. Questo metodo di inizializzare la variabili risulta molto comodo quando si hanno delle lunghe sequenze di dati (in questo caso si sarebbe potuta utilizzare una inizializzazione classica in sede di dichiarazione). Successivamente vengono assegnati alle variabili del driver relativi alla mappa dei tile ed alla tabella dei colori i rispettivi puntatori alle aree di memoria individuate (tramite le due variabili). Il driver è quindi avviato utilizzando la sua funzione Start, e passando come parametro il puntatore alla variabile d'inizio dei parametri. Segue l'inizializzazione della tabella dei colori, in cui viene utilizzata un'espressione che genera una sfumatura tra vari colori (una sorta di "arcobaleno"). L'inizializzazione della mappa dei tile è fatta in modo da puntare ad una locazione chiamata Onscreen_buffer, che rappresenta proprio l'inizio della zona di memoria in cui è presente la descrizione dei tile. Se si scrive una long in questa locazione si definisce in pratica il contenuto dei primi 16 pixel del primo tile (in alto a sinistra). Scrivendo una long all'indirizzo Onscreen_buffer+4 si definisce la seconda riga di pixel del primo tile e così via. A Onscreen_buffer+16 si definirà la rima riga del secondo tile in alto a sinistra.

Il driver grafico

Da quanto detto a proposito del driver TV appare chiaro che esso permette di creare immagini basate su tile in maniera relativamente semplice e veloce, ma risulta piuttosto scomodo da usare quando si vogliono disegnare elementi grafici che non sono facilmente descrivibili tramite tile, quali ad esempio linee, punti, archi o poligoni.
Per semplificare questo compito è possibile ricorrere al driver grafico, costituito dal file GRAPHICS_DRV_010.spin. Il driver grafico fornisce un'interfaccia ad un livello un po' più alto al driver TV, con funzioni prevalentemente orientate alla grafica di tipo bitmap piuttosto che a tile. Il driver grafico contiene diverse funzioni avanzate che permettono di disegnare immagini molto complesse e di realizzare animazioni in maniera estremamente semplice. Purtroppo però la gestione di funzioni così avanzate rende il driver grafico leggermente lento rispetto a quello video, e non particolarmente adatto a gestire animazioni di scene complesse, come quelle che possono aversi in un videogame. Il driver risulta comunque molto potente e versatile,

DVR, la tecnologia digitale per la tua sicurezza

Vasta gamma di videoregistratori digitali per qualsiasi esigenza, dalla casa, al punto vendita, alla piccola o grande azienda. Da 4 a 16 canali, differenti sistemi di compressione, con interfaccia LAN e video web server, con trasferimento dati USB o back-up su DVD: scegli il modello che meglio si adatta alle tue esigenze.

I DVR vengono forniti senza Hard Disk.

FR329W - DVR / MULTIPLEXER A 4 CANALI MPEG-4/JPEG CON VIDEO WEB SERVER

€ 310,00

PREZZO IMBATTIBILE



Videoregistratore digitale real-time a quattro canali con frame rate di 100 IPS e interfaccia Ethernet (10-100 Base-T). Formato di compressione MPEG-4(CIF), MJPEG (frame), funzione Motion Detection avanzata per tutti i canali con sensibilità regolabile, possibilità di registrare, riprodurre e visualizzare contemporaneamente le immagini. Visualizzazione rel-time da remoto tramite connessione Internet (massimo 5 utenti) con possibilità di effettuare registrazioni e back-up. Grazie all'elevato livello di compressione il DVR è in grado di registrare per 170 giorni consecutivamente (HDD da 500 GB, 4CH, CIF alta qualità, 15 IPS). Standard video NTSC e PAL • Livello ingressi video 1 Vpp 75 Ohm (BNC) • 4 Uscite loop, un'uscita monitor principale e una secondaria, livello 1 Vpp 75 ohm (BNC) • Regolazione qualità immagine • Hard Disk supportato IDE, ATA 66, capacità oltre 400 GB • Refresh video 120 IPS (NTSC) / 100 IPS (PAL) • un ingresso e un'uscita audio (RCA) • Registrazione pre-allarme • Compressione Motion JPEG per trasmissione via web • Interfaccia Web compatibile IE browser e software AP • Invio delle immagini in caso di allarme tramite Email o via FTP • Protocolli supportati TCP/IP, PPPoE, DHCP, DDNS • 4 ingressi e un'uscita d'allarme • Zoom 2 X digitale (solo in modalità real-time) • Alimentazione 19 VDC tramite adattatore di rete 100 ~ 240 Vac incluso • Assorbimento < 42 W • Dimensioni 343x223x59 mm • Sistema di ripristino di tutte le funzioni dopo un black-out.

FR261 - DVR 4 CANALI con CONNESSIONE LAN/INTERNET e PORTA USB

NEW PRICE **€ 398,00**



DVR 4 canali (1Vpp, 75 Ohm, BNC) MJPEG dotato di Hard Disk estraibile (escluso) e di uscita USB 2.0 per lo scarico veloce delle immagini senza rimozione dell'HDD (software di gestione incluso). Le immagini riprese dalle quattro telecamere possono essere registrate su HDD (capacità max. 120 GB) e controllate da remoto tramite Internet. Risoluzione 720x576 pixel (320 x 136 singolo canale, 640 x 272 modalità quad in registrazione); 100 fps singolo canale, 25 fps modalità quad; 1 canale audio; Ethernet 10/100BaseT; alimentazione 12 Vdc (adattatore di rete incluso); dimensioni 360x290x65 mm; peso 5,5 Kg.

FR344 -DVR/MULTIPLEXER 4 CH MPEG4 con PORTA USB e VIDEO WEB SERVER

new **€ 480,00**



DVR 4 canali con frame rate di 100 IPS (CIF) dotato di cassetto HDD estraibile e d'interfaccia USB 2.0 per back-up. L'apparecchio supporta Hard Disk (IDE, UDMA 66) con capacità superiore a 400 GB. Formato di compressione MPEG4/MJPEG; refresh video 100 IPS (PAL); funzione Motion Detection avanzata, possibilità di registrare, riprodurre e visualizzare contemporaneamente le immagini. Visualizzazione real-time e controllo da remoto tramite connessione internet con possibilità di effettuare registrazioni e back-up; interfaccia network Ethernet (10/100 Base-T). Interfaccia Web compatibile IE browser e software AP. Invio immagini in caso di allarme via e-mail o FTP; protocolli supportati TCP/IP, PPPoE, DHCP e funzione DDNS. Controllo PTZ; 4 ingressi e 2 uscite audio; 4 ingressi e un'uscita d'allarme. Completo di adattatore di rete, HDD non incluso.

FR337 -DVR 4 INGRESSI MPEG4/WEB/CDRW

€ 610,00



DVR a 4 canali dotato di masterizzatore CD R/W per effettuare il backup delle registrazioni su supporto CD, e di un'interfaccia USB per backup su unità flash USB 1.1 / 2.0. La registrazione delle immagini avviene su HDD interno (IDE, ATA66), non incluso, con capacità di oltre 400 GB. Frame rate 25 IPS PAL (100 img/sec, CIF); compressione MJPEG per trasmissione su web; interfaccia Web compatibile IE browser e software AP; invio immagini in caso di allarme via e-mail o FTP; protocolli supportati TCP/IP, PPPoE, DHCP e DDNS (Dynamic DNS); protocollo PELCO-D; video motion per tutti i canali; 4 ingressi e un'uscita d'allarme. Due ingressi audio, menu multilingua, funzione di registrazione a distanza, registrazione nascosta, ripristino automatico del sistema a seguito della mancanza di alimentazione. Il DVR consente di eseguire contemporaneamente le funzioni di visualizzazione in tempo reale, registrazione, riproduzione, backup e rete. Completo di adattatore di rete.

FR334 - DVR 8 CANALI MPEG4/GPRS/ETHERNET/ CD-RW - CONNESSIONE PER RAID

€ 1.250,00



La notevole capacità di cui dispone il DVR (possibilià di installare fino a tre HDD da 500 GB ciascuno oppure due HDD e un RAID esterno, non inclusi) e la compressione video MPEG4 (paragonabile allo standard divX) consentono di registrare ininterrottamente per molti mesi. Il Video Web Server di cui dispone permette la visualizzazione delle immagini mediante browser tramite qualsiasi PC collegato a Internet. Le immagini possono essere visualizzate da remoto anche mediante telefono cellulare dotato di connessione GPRS. Masterizzatore CD-RW per backup delle registrazioni su supporto CD; salvataggio dati su memoria Flash USB; controllo telecamera con funzione PTZ. Adattatore di rete e telecomando IR inclusi.

DVR 16 CANALI MPEG4/GPRS/ETHERNET/CD-RW - CONNESSIONE PER RAID

Stesse caratteristiche del modello FR334 ma con 16 canali anzichè 8.

FR335 - € 1.750,00

CCTV ProLine

€ 560,00



FR319 - DVR/MULTIPLEXER 9 INGRESSI

Versione a 9 canali con cassetto Hard Disk estraibile. Integra in un unico apparecchio un DVR e un multiplexer full-duplex a 9 canali. Quattro differenti modalità di visualizzazione: 1 canale (a pieno schermo), 4 canali (modalità quad), 7 e 9 canali. Funzionalità duplex: registrazione e live multischermo contemporanei, di ricerca rapida delle registrazioni su Data/Ora e su evento d'allarme. Compressione Wavelet.

DVR/MULTIPLEXER 9 INGRESSI CON WEB-SERVER

Stesse caratteristiche del modello FR319 ma con l'aggiunta di una interfaccia Ethernet che rende possibile la visualizzazione delle immagini da remoto tramite una connessione Internet.

FR319W - € 670,00

FR322 DVR/MULTIPLEXER 16 CH PORTA USB, WEB SERVER e GPRS

€ 1.450,00



DVR a 16 canali nel quale è possibile installare fino a due HHD con capacità di oltre 400 GB ciascuno. Garantisce moltissime ore di registrazione con una buona qualità dell'immagine grazie alla compressione MPEG4. Integra in un unico apparecchio un DVR e un multiplexer full-duplex a 16 canali. Dispone di Video Web Server con possibilità di visualizzare le immagini da remoto anche mediante telefono cellulare dotato di connessione GPRS. È dotato di una pratica interfaccia USB per lo scarico dei filmati su PC. Completo di adattatore di rete e di telecomando IR per gestione DVR e controllo telecamera con

DVR/MULTIPLEXER 16 INGRESSI con PORTA USB, WEB SERVER, GPRS e DVD

Stesse caratteristiche del modello FR322 ma con l'aggiunta di un masterizzatore DVD-RW che permette di effettuare il backup delle registrazioni su DVD.

FR322D - € 1.670,00

FR336 - DISK ARRAY BOX per TRE HARD DISK con FUNZIONE HUB

€ 280,00



Unità di espansione di capacità di memoria per DVR dotati d'interfaccia Disk Array.
Dispone di funzione HUB che consente di collegare in cascata più unità FR336. Al suo interno si possono alloggiare fino a tre Hard Disk da 3,5" (di qualsiasi marca) caratteristica che - insieme alla funzione HUB - permette di espandere, in modo praticamente illimitato, la memoria dei DVR predisposti per unità disco esterna e di prolungare notevolmente l'autonomia di registrazione. Funzione HUB; interfaccia Host Port: 1 x IDE (cavo SCSI M. incluso); interfaccia Device Port 3 x IDE; supporto modalità JBOD; velocità di trasferimento HDD Ultra DMA/100-66/33; interfaccia HDD ATA-6 (T13/1410D REV3) DMA66; alimentazione 19 Vdc (adattatore di rete incluso); consumo 2,3 A; dimensioni in mm 432 (L) x 305 (P) x 60 (H); peso 6 kg.

DVR4USB -SCHEDA USB a 4 CANALI per ACQUISIZIONE A/V

€ 115,00



new

Trasforma il tuo PC in un vero e proprio DVR con l'ausilio di questa scheda da collegare alla porta USB! Quattro ingressi video e due audio, possibilità di visualizzare le immagini riprese in modalità QUAD o a pieno schermo anche da remoto via LAN o via Internet. Un potente software consente di ottenere tutte le funzioni tipiche di un DVR.

ACCESSORI per DVR

- **BOXHHD** - Cassetto estraibile per DVR FR319 e FR233. **€ 22,50**
- **DVRCARTR** - Cassetto estraibile per DVR FR190 e FR233. **€ 32,00**
- **DVR-CARTR1** - Cassetto estraibile per DVR FR318, FR320, FR320W, FR327. **€ 16,00**

Tutti i prezzi sono da intendersi IVA inclusa.

Codice MIP 268095

FUTURA ELETTRONICA idee in elettronica

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

Disponibili presso i nostri Rivenditori o nel nostro punto vendita di Gallarate (VA). Caratteristiche tecniche e vendita on-line direttamente sul sito **www.futuranet.it**

soprattutto per chi inizia. Dal momento che si ha a disposizione il file sorgente, è possibile modificarlo per adattarlo alle proprie esigenze o per ottimizzarne le prestazioni.
Una possibilità interessante offerta dal driver grafico è quella di utilizzare la tecnica del "double buffer" per rendere più fluide le animazioni. Questa tecnica consiste nell'impiegare due buffer video (per utilizzare una terminologia già impiegata a proposito del driver TV, si tratta della memoria riservata alle bitmap dei tile): uno è utilizzato dal driver TV per leggere e visualizzare a schermo i dati dell'immagine, l'altro invece è impiegato per costruire l'immagine e non é mai visualizzato. Una volta terminata la costruzione dell'immagine, il buffer secondario viene copiato per intero e molto velocemente in quello primario, in questo modo verrà visualizzata a video immediatamente l'immagine completa, e non la si vedrà costruire a poco a poco (questo darebbe un effetto di sfarfallamento o "fliker"). Un'altra particolarità del driver grafico è che in molte funzioni sono utilizzate le coordinate polari invece che le più comuni cartesiane. Questo rende a volte scomodo l'introduzione dei dati dell'immagine, ma in compenso facilita e rende più veloci gli effetti di rotazione degli oggetti e la realizzazione figure a simmetria circolare.
Le funzioni messe a disposizione dal driver sono elencate in Tabella 1. Per una più accurata descrizione si rimanda al manuale di Hydra o al sorgente del driver. Così come il driver TV, anche il driver grafico deve essere avviato con la funzione Start, ed eventualmente configurato con la funzione Setup prima di essere utilizzato. Dal momento che il driver grafico si interfaccia con il driver TV sarà necessario richiamare ed avviare sempre anche quest'ultimo.
Nel listato 4 è mostrato un esempio di utilizzo del driver grafico, in un programma che visualizza un pixel che si sposta sullo schermo da sinistra a destra, utilizzando la tecnica del doppio buffer.
Il codice risulta abbastanza simile a quello mostrato in precedenza, le uniche differenze sono la dichiarazione del buffer off-line tra le variabili, l'inclusione e l'avvio del driver grafico, le operazioni di disegno sul buffer off-line e la copiatura su quello on-line.
Nella funzione Setup del driver grafico è stato specificato il numero di tile orizzontali e verticali, l'offset delle coordinate e l'indirizzo del buffer off-line.
Il driver grafico utilizza un sistema di coordinate che prevede che il punto (0,0) sia nell'angolo in basso a sinistra dello schermo. E' possibile comunque specificare uno scostamento e quindi spostare l'origine, in modo da avere ad esempio il punto (0,0) corrispondente al centro dello schermo e coordinate che vanno da –altezza/larghezza a +altezza/larghezza, come fatto nel programma.
Questo semplice programma può essere modificato fino ad ottenere degli effetti più interessanti e complessi. Ad esempio si può sostituire la funzione Pixel con Box, Tri, Arc o Text ad esempio, in modo da disegnare rettangoli, triangoli, cerchi o testo invece che il semplice pixel, e prevedere qualche funzione più sofisticata per gestire l'aggiornamento della posizione (ad esempio rimbalzo, caduta, animazione particellare, etc.).
A proposito della funzione Text va sottolineato che si tratta di una funzione molto potente e versatile, che permette di visualizzare a schermo un testo in una specifica posizione, usando caratteri di dimensioni spaziatura e giustificazione a scelta. Il prototipo della funzione è:

```
PUB Text(x, y, str_ptr)
```

In cui x ed y sono le coordinate dello schermo dove verrà visualizzato il testo, mentre str_ptr è un puntatore ad una stringa espressa nel formato classico del C (sequenza di caratteri terminati con uno 0). Le caratteristiche del testo possono essere impostate con la funzione Testmode, il cui prototipo è il seguente:

```
PUB Textmode(x_scale, y_scale,
spacing, just)
```

In cui x_scale ed y_scale sono le dimensioni orizzontali e verticali dei caratteri, spacing è la spaziatura (6=normale) e just è un codice che definisce la giustificazione orizzontale e verticale (si veda il sorgente).
Per provare queste funzioni nel-

Funzione	Descrizione
Start / Stop	Avvia / arresta driver
Setup	Imposta caratteristiche grafica
Clear	Cancella buffer (off-line)
Copy	Copia il buffer off-line in quello on-line
Color / Width / Colorwidth	Impostano colore e dimensione della penna
Plot	Disegna un pixel
Line	Disegna una linea
Arc	Disegna un arco, settore o cerchio
Vec	Disegna un poligono
Text / Textarc	Disegna un testo lineare o curvo
Box	Disegna un rettangolo
Quad	Disegna un quadrilatero
Tri	Disegna un triangolo

Tabella 1
Principali funzioni del driver grafico

Listato 4

```
' *** Costanti ***
CON
 _clkmode = xtal2 + pll4x              ' clock ext. + PLL 4x
 _xinfreq = 10_000_000                 ' freq. ext. = 10MHz
 _stack = ($3000 + $3000 + 64) >> 2   ' spazio per stack
 ' numero parametric driver TV
 PARAMCOUNT         = 14
 OFFSCREEN_BUFFER   = $2000            ' buffer video off
 ONSCREEN_BUFFER    = $5000            ' buffer video on
 ' dimensioni tilemap e schermo
 X_TILES             = 16
 Y_TILES             = 12
 SCREEN_WIDTH       = 256
 SCREEN_HEIGHT      = 192
' *** Variabili ***
VAR
 long  tv_status      '0/1/2 = off/visible/invisible
 long  tv_enable      '0/? = off/on
 long  tv_pins        '%ppmmm = pins
 long  tv_mode        '%ccinp = chroma,interlace,ntsc/pal,swap
 long  tv_screen      'pointer to screen (words)
 long  tv_colors      'pointer to colors (longs)
 long  tv_hc          'horizontal cells
 long  tv_vc          'vertical cells
 long  tv_hx          'horizontal cell expansion
 long  tv_vx          'vertical cell expansion
 long  tv_ho          'horizontal offset
 long  tv_vo          'vertical offset
 long  tv_broadcast   'broadcast frequency (Hz)
 long  tv_auralcog    'aural fm cog
 word  screen[X_TILES * Y_TILES] ' mappa dei tile
 long  colors[64]                ' tavola dei colori
' *** Oggetti ***
OBJ
 tv     : "tv_drv_010.spin"         ' driver TV
 gr     : "graphics_drv_010.spin"   ' driver grafico
' *** Funzione ***
PUB start | i, dx, dy, x, y
 'start tv
 longmove(@tv_status, @tvparams, paramcount)
 tv_screen := @screen
 tv_colors := @colors
 tv.start(@tv_status)
 'init colori
 repeat i from 0 to 64
   colors[i] := $00001010 * (i+4) & $F + $FB060C02
 'init tile
 repeat dx from 0 to tv_hc - 1
   repeat dy from 0 to tv_vc - 1
     screen[dy * tv_hc + dx] := onscreen_buffer >> 6 +
                   dy + dx * tv_vc + ((dy & $3F) << 10)
 ' avvia driver grafico
 gr.start
```

```
gr.setup(X_TILES, Y_TILES, SCREEN_WIDTH/2, SCREEN_HEIGHT/2, offscreen_buffer)
' loop principale
repeat while TRUE

  ' —— disegno —
  ' cancella buffer-off
   gr.clear

  ' penna colore 1, larghezza minima
  gr.colorwidth(1,0)

  ' disegna pixel
  gr.plot(x, 0)
  ' sposta coord. pixel
  if (++x > SCREEN_WIDTH/2)
     x := -SCREEN_WIDTH/2
  ' copia da buffer-off a buffer-on
  gr.copy(onscreen_buffer)
  ' —— fine disegno ——
' fine loop principale
' *** Dati binari ***
DAT
tvparams     long  0            'status
             long  1             'enable
             long  %011_0000    'pins
             long  %0000         'mode
             long  0             'screen
             long  0             'colors
             long  X_TILES      'hc
             long  Y_TILES      'vc
             long  10            'hx timing stretch
             long  1             'vx
             long  0             'ho
             long  0             'vo
             long  55_250_000   'broadcast: ch. 2 VHF
             long  0             'auralcog
```

l'esempio mostrato prima basta aggiungere le due righe seguenti appena sotto l'istruzione che disegna il punto:

```
gr.textmode(1,1,6,5)
gr.text(0,10,@stringa)
```

e dichiarare la stringa nella sezione DAT come segue:

```
stringa      byte  "Ciao!", 0
```

Per una completa panoramica delle funzioni della libreria grafica, e per rendersi conto delle loro potenzialità e delle modalità di utilizzo è possibile eseguire o consultare il file "Graphics_demo_010.spin" fornito come esempio.

Conclusioni

Le funzioni viste ed i programmi mostrati rappresentano gli elementi base per realizzare immagini ed animazioni a video, e per iniziare a costruire delle semplici demo. Per iniziare a realizzare dei videogame veri e propri però è necessario aggiungere altri due elementi che verranno approfonditi nella prossima puntata: le periferiche di input (controller, mouse e tastiera) ed il suono. Nella prossima puntata verranno presentati i driver relativi a queste funzioni e descritto il codice di un primo semplice videogame.

More Info Please!
Inserisci il Codice **268088** alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

www.rccitaly.com

Prodotti e sistemi per il controllo e la comunicazione industriale

Modulo di telemetria via SMS Infinite Scom-100



- Motore Wavecom GSM quad-band
- 4 Ingressi Digitali
- 4 Uscite Digitali a Relè (250Vac-6A)
- 2 Ingressi Analogici (10 Bit, 0-10V, 4-20mA)
- Alimentazione 12 o 24 Vdc
- Installazione guida-din
- Set Esteso SMS-ASCII per comando e configurazione remoti
- Moduli aggiuntivi (max 8) con 4 ingressi e 2 uscite digitali
- Moduli aggiuntivi (max 4) con 4 ingressi analogici
- Ingressi e uscite completamente parametrizzabili

Modem GSM Erco & Gener GenPro14e



- Motore Wavecom GSM quad-band
- Supporta dati-voce-SMS
- 1 Ingresso microfono
- 1 Uscita Altoparlante
- 1 Porta Seriale RS232
- 3 Ingressi digitali Optoisolati
- 1 Uscita digitale Open-Collector
- Alimentazione 5-32 Vdc
- Dimensioni 73 x 54 x 25 mm
- Programmabile con set comandi AT estesi e flash-Ram "On Board"
- Disponibile in versione GPRS con Stack TCP/IP integrato (GenPro24e) e GPS a 16 canali integrato (GenLoc31e dual band)

Codice MIP 268099

Access point Sparklan OR-100



- Contenitore IP66 per installazioni Outdoor (-30 +60°C)
- Conforme 802.11b/g (2,4GHz – 11/54Mbit/s)
- Supporta VoIP e QoS
- Supporta cifratura WEP e WPA
- Alimentazione PoE consumo 7W
- Configurabile come AP, Repeater, Client
- Funzionalità punto-punto, punto-multipunto, WDS (wireless distribution system)
- Connettore N-type per antenna esterna

RCC Remote Command & Communication

Distribuiti e supportati da:

Via G. di Vittorio 19 - 20097 San Donato Milanese (MI)
Tel: +39 335 8784 738, Fax: +39 02 5187 6194
www.rccitaly.com - info@rccitaly.com

Disponibili anche presso i nostri partner a valore aggiunto

www.automa.it www.gestinfo.it www.mg-eng.com www.sensorline.biz

Come gestire una tastiera utilizzando

Proponiamo un sofisticato metodo per la gestione di una tastiera con un microcontrollore. A differenza della tradizionale e complessa connessione a matrice, questa tecnica richiede un solo ingresso analogico.

L'implementazione di una tastiera a matrice è spesso un'operazione complessa e richiede le seguenti condizioni:

- La disponibilità e l'utilizzo di molte porte di I/O del micro;
- La corrispondenza incrociata di eventi;
- Un codice eseguibile non tra i più semplici.

In più, se la richiesta di tasti aggiuntivi è una condizione necessaria, la procedura diventa veramente problematica, specialmente se essi superano le 20 unità.
Con il metodo proposto, anche l'uso di una tastiera costituita da numerosi tasti non comporta alcun problema, e il codice di programma da realizzare è molto semplice e facilmente espandibile e implementabile. Unica condizione richiesta: la presenza di una porta di ingresso analogico (ADC) nel microcontrollore.



Figura 1
Una classica tastiera a matrice

La tradizionale tastiera a matrice

Sebbene l'adozione dello schema tradizionale non comporti problemi di sorta, il suo utilizzo a volte potrebbe risultare un po' scomodo.
In primo luogo infatti esso implica l'utilizzo di numerose porte di I/O, come si vede nello schema di cui alla figura 2.
Il numero di pin richiesti è infatti pari al numero di righe più il numero di colonne: nell'esempio i 16 tasti sono organizzati attraverso una matrice di 4 righe e 4 colonne, per cui il numero di ingressi digitali richiesto è pari ad 8. La sua lettura, diciamolo sinceramente, non risulta immediatamente comprensibile.
In secondo luogo sono richieste parecchie resistenze di *pull-down*, che assicurino il potenziale di massa alle porte non coinvolte da alcun segnale.
In terzo luogo, il codice che si deve implementare, non utilizzando le librerie già pronte, è davvero complicato, e potrebbe dar luogo a qualche *bug* se esso non è attentamente valutato, testato e controllato.
In quarto luogo, se il numero dei pulsanti da gestire comincia a crescere, le porte da utilizzare aumenterebbero a dismisura, sottraendole ad altre periferiche utili al prototipo.
Insomma una serie di piccoli problemi che, se sommati, potrebbero, se non far del tutto desistere, scoraggiare il principiante.
Il Mikrobasic risolve brillantemente il problema, prevedendo, tra le sue numerose librerie, anche quella dedicata alle tastiere a matrice, e precisamente alla *Keypad* 4x4. Le procedure e le funzioni previste dal compilatore sono:

- Keypad_init;
- Keypad_read;
- Keypad_released.

Non stiamo a spiegare il loro funzionamento in questa sede ma rimandiamo il lettore all'apposito manuale. Sebbene tale libreria sia molto sofisticata, essa gestisce solamente le matrici 4x4 nonché le sottomatrici 4x1, 4x2, 4x3 occupando la stessa quantità di memoria programma.

un solo ingresso del PIC



di Giovanni Di Maria

Il metodo proposto

Al fine di risparmiare molte porte di I/O del micro, e quindi di riutilizzarle per altre periferiche ed utilizzatori, si può adottare il metodo illustrato in queste pagine. Con esso è possibile implementare una tastiera (non a matrice) composta semplicemente da tanti pulsanti normalmente aperti.
Il metodo si basa sull'impiego di un multi partitore resistivo, che fornisce ai vari nodi le diverse tensioni. Sarà poi l'ingresso ADC del micro a "leggere" la tensione attivata dal relativo pulsante per intraprendere, finalmente, la decisione del caso. Tutto questo avviene grazie al software integrato.
Indubbiamente il metodo presenta molti vantaggi e pochi svantaggi, riassunti nella tabella 1. Sarà poi l'utilizzatore a vagliarli complessivamente e decidere se continuare ad usare la "classica" tastiera a matrice ovvero optare per questa realizzazione ben più semplice e priva di problemi di sorta.

Il partitore resistivo

Il partitore resistivo è in pratica un divisore di tensione, formato da due o più resistenze collegate in serie, ai cui capi risulta una tensione diversa e ben precisa. La quantità di corrente che circola nel circuito è uguale in tutti i componenti, poiché non vi è divisione della stessa.

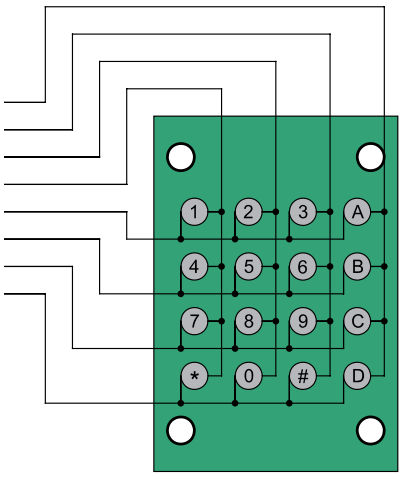


Figura 3
Le connessioni della tradizionale tastiera a matrice

Per il calcolo della tensione ai vari nodi del partitore si usa, ovviamente, la legge di Ohm. Per la comprensione del circuito intraprenderemo adesso una serie di osservazioni, utilizzando due circuiti di partizione, comprendenti rispettivamente due e tre resistenze di divisione.
Nei circuiti in esame, e naturalmente nel prototipo finale, il valore delle resistenze è identico, pertanto i calcoli sono ampiamente semplificati.

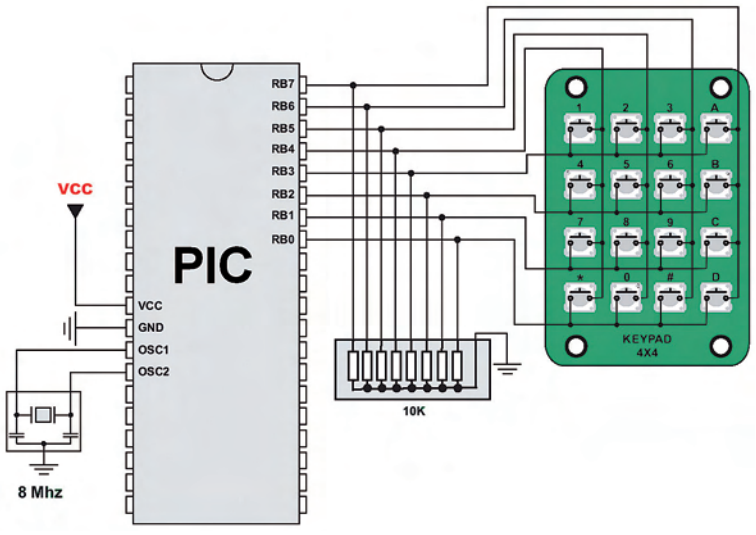


Figura 2
Lo schema tradizionale del collegamento Pic / Tastiera

Partitore con due resistenze

In questo tipo di circuito, la corrente attraversa due resistenze. Esaminiamo lo schema della figura 4. L'alimentazione è di 5 Volt. Le resistenze sono tutte da 1K, quindi tutte del medesimo valore. Dal circuito possono venir prelevate due differenti tensioni: una dal punto A e una dal punto B. Dal primo si ottiene una tensione di 5 Volt (corrispondenti in pratica alla tensione di alimentazione), mentre dal secondo si ottiene una tensione dimezzata, ossia di 2,5 Volt.
Per calcolare a priori i diversi punti di tensione si utilizza, come detto prima,

VANTAGGI	SVANTAGGI
È necessaria solamente una porta di I/O	Il micro deve prevedere un ingresso ADC
Il software da scrivere è molto semplice	È consigliabile utilizzare resistenze precise (1% tolleranza). Vedi proseguo dell'articolo.
La risposta alla pressione del tasto è molto veloce	
Possibilità di prevedere molti tasti (anche più di 30)	
Non è necessario utilizzare una tastiera a matrice	
Collegamenti e cablaggi molto semplici	
Possibilità di processare più tasti contemporaneamente	

Tabella 1
Vantaggi e svantaggi nell'utilizzo del presente metodo

la legge di Ohm. Si inizia naturalmente con il calcolo della corrente assorbita, secondo la formula:

$$I=\frac{V}{R}=\frac{5}{(1000+1000)}=0,0025A=2,5mA$$

da cui:

$$V=R\cdot I=V_{R2}=R2\cdot I=1000\Omega\cdot 0,0025A=2,5V$$

(punto B)

$$V=R\cdot I=V_{R2+R1}=(R2+R1)\cdot I=2000\Omega\cdot 0,0025A=5V$$

(punto A).

Ricordiamo ai lettori che queste formule sono esatte solo nel caso in cui a valle del partitore non viene inserito alcun carico esterno. Un altro formula che permette di calcolare lo scarto di tensione tra un nodo e l'altro, prevede la divisione della tensione di alimentazione per il numero di resistori. Si utilizza solamente se le resistenze hanno lo stesso valore.

Partitore con tre resistenze

Anche se il circuito è più complesso del precedente, il procedimento di calcolo è identico. Lo schema relativo è raffigurato in figura 5. Dal circuito possono venir prelevate tre differenti tensioni: una dal punto A, una dal punto B e l'altra dal punto C. Dal primo si ottiene una tensione di 5 Volt (corrispondenti in pratica alla tensione di alimentazione), dal secondo si ottiene una tensione a 3,3 Volt e dal terzo una tensione pari a 1,6 Volt. Eseguiamo i calcoli relativi per ricavare la corrente assorbita dal circuito:

$$I=\frac{V}{R}=\frac{5}{(1000+1000+1000)}=0,0016A=1,6mA$$

da cui:

$$V=R\cdot I=V_{R3}=R3\cdot I=1000\Omega\cdot 0,00166A=1,66V$$

(punto C)

$$V=R\cdot I=V_{R3+R2}=(R3+R2)\cdot I=2000\Omega\cdot 0,00166A=3,3V$$

(punto B)

$$V=R\cdot I=V_{R3+R2+R1}=(R3+R2+R1)\cdot I=3000\Omega\cdot 0,00166A=5V$$

(punto A).

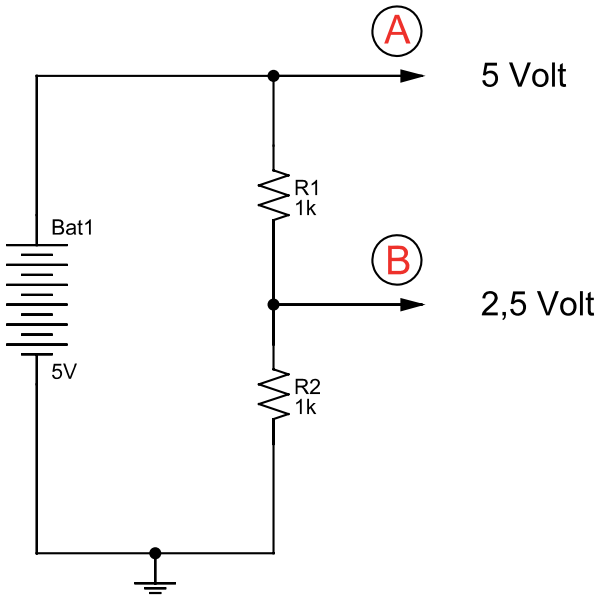


Figura 4
Il partitore con due resistenze

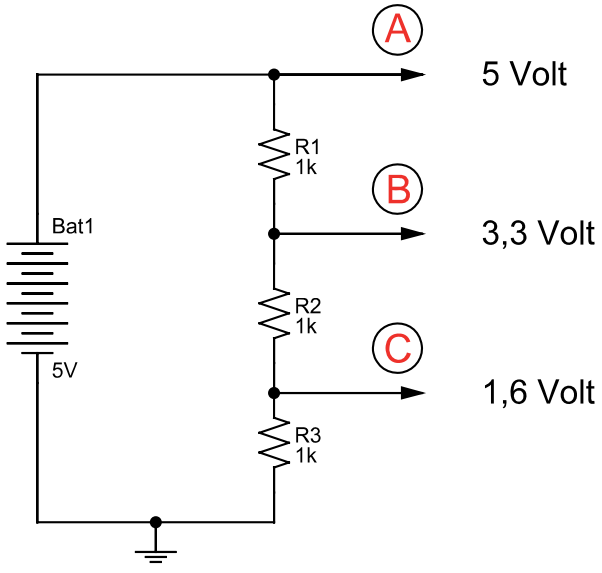


Figura 5
Il partitore con tre resistenze

Schema elettrico del prototipo

Esaminiamo adesso lo schema elettrico del prototipo. Esso è visibile in figura 6. Il sistema è alimentato da una batteria o alimentatore capace di erogare una tensione "esatta" di 5 Volt. Si consiglia di rispettare al massimo questo valore, al fine di ottenere una tensione precisa anche dal partitore. Se nessun pulsante è premuto, nessuna tensione giunge al piedino RA0 del Pic, pertanto tale ingresso (di tipo analogico) è posto a massa grazie alla resistenza di pull-down R10, dal valore molto elevato di 470K, comunque sufficiente per espletare al meglio il suo lavoro. Si consiglia di non abbassare tale valore per non "ingannare" l'uscita del partitore. Premendo invece un qualsiasi pulsante, il piedino RA0 sarà sottoposto alla tensione del relativo nodo del partitore.

In questo caso il suo valore risulterà pari a quello dei calcoli prima effettuati, meno una piccolissima percentuale dovuta al resistore R10. Il software noterà subito tale innalzamento di tensione e, confrontandola con altre di riferimento, sarà in grado di determinare quale tasto è stato premuto. Se ovviamente la tensione risultante al pin RA0 è uguale a 0, vorrà dire che nessun pulsante è stato premuto.

L'impedenza di ingresso del piedino RA0 è molto elevata, non influisce pertanto alle variazioni di tensione. Il display a 7 segmenti ha l'unico scopo di visualizzare il numero del tasto premuto. Naturalmente esso non è un componente obbligatorio, e il programmatore può adottare la configurazione che meglio rispecchia le sue esigenze.

Obiettivo del prototipo

Lo schema elettrico, visibile in figura 6, costituisce naturalmente un prototipo sperimentale ma perfettamente funzionante.

È la base di partenza per qualsiasi progetto che preveda l'utilizzo di una tastiera o, in generale, di molti tasti per l'input delle informazioni. Sono previsti, in questo caso, 9 pulsanti normalmente aperti. Se nessun pulsante è premuto, il display a 7 segmenti rimane spento.

La pressione di uno dei tasti determina invece la formazione del relativo numero sul display. Come si evince dalla tabella 2, ogni tasto coinvolto a pressione genera un preciso valore, derivante dall'azione del partitore. Se nessun tasto è premuto, la porta analogica acquisirà una tensione nulla. La differenza di tensione ad ogni gradino è di circa 0,555 Volt, corrispondente al risultato delle divisione (5 Volt / 9 tasti = 0,555). Ricordiamo che la resistenza R10, influisce leggermente sui risultati, in quanto si pone in parallelo al ramo inferiore del partitore.

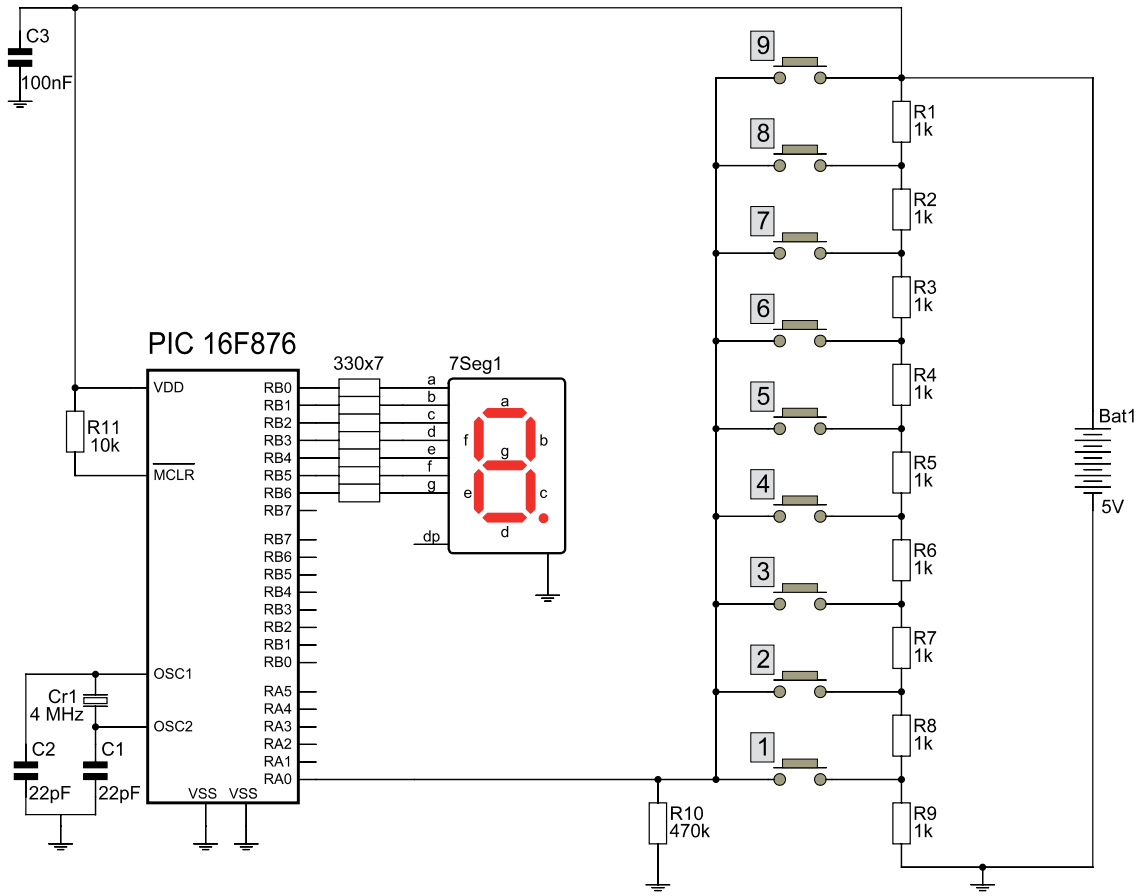


Figura 6
Schema elettrico del prototipo

Tasto premuto	Valore al piedino RA0 (Volt)	Valore corrispondente digitale a 10 bit
Nessun tasto	0	0
1	0,555	114
2	1,107	226
3	1,660	340
4	2,212	453
5	2,764	566
6	3,319	679
7	3,876	793
8	4,436	908
9	5	1023

Tabella 2
Tensione al piedino analogico RA0 a seconda del tasto premuto

Il listato Basic

Il listato Basic, compilabile con il Mikrobasic, ma adattabile con poche modifiche a qualsiasi altro linguaggio e compilatore, è molto semplice e ricco di commenti ed è riportato nel listato 1. Dopo la definizione delle variabili, si passa alla definizione delle porte di ingresso e di uscita. Il programma prosegue pertanto eseguendo un test del display, codificato all'interno del ciclo FOR/NEXT. La verifica, eseguita 5 volte, illumina, uno per volta, i sette segmenti del display. Finalmente ha inizio il ciclo infinito, all'interno del quale viene rilevata la eventuale presenza di tensione uscente dal partitore. Attraverso un serie di istruzioni IF, viene rilevato il livello di tensione (e quindi il relativo tasto premuto), con la conseguente illuminazione del display a 7 segmenti. Da notare che se nessun tasto viene premuto, il display resta spento (PORTB=0). Per i valori di codificazione di cifra e di livello tensione, occorre riferirsi alle tabelle 2, 3 e 4.

La tolleranza dei resistori

L'ideale naturalmente sarebbe utilizzare delle resistenze per il partitore con tolleranza dell'1%, in

Valore da visualizzare	Valore della PORT-B
0	63
1	6
2	91
3	79
4	102
5	109
6	125
7	7
8	127
9	111

Tabella 3
Valori del display a 7 segmenti e corrispondente codifica del Pic

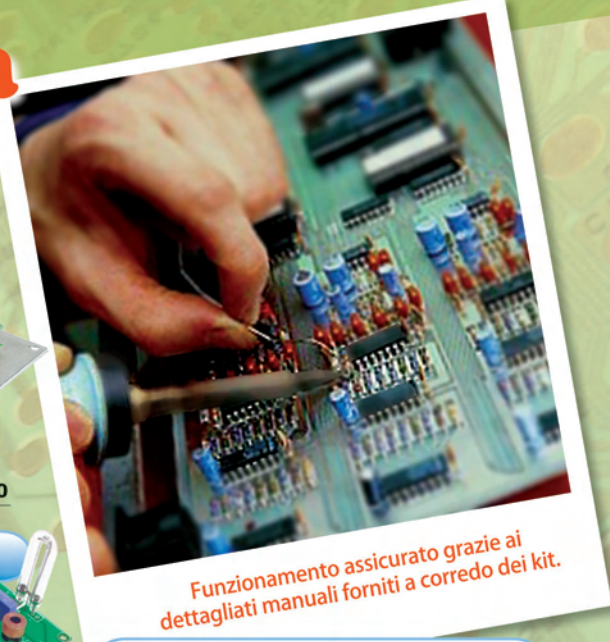
Listato 1

```
rem Gestione TASTI
rem con input ANALOGICO
rem 9 Tasti e un solo ingresso!!!
rem by Giovanni Di Maria
program tastiera
dim k as byte  'Variabile per i cicli
dim lettura as word 'Contiene i bit della
                    'PORTA.0
ADCON1=$80  'PORTA per input analogico
TRISA=1     'PORT-A.0 come input
trisb=0     'PORTB in Output
portb=0     'Spegne il Display
delay_ms(100)  'Piccola pausa iniziale
rem   Inizia il Test del Display
rem   ripetendolo 5 volte
rem   accendendo un segmento alla volta
for k=1 to 5
    portb=1   'Segmento A
    delay_ms(150)
    portb=2   'Segmento B
    delay_ms(150)
    portb=4   'Segmento C
    delay_ms(150)
    portb=8   'Segmento D
    delay_ms(150)
    portb=16  'Segmento E
    delay_ms(150)
    portb=32  'Segmento F
    delay_ms(150)
    portb=64  'Segmento G
    delay_ms(150)
next k
portb=0   'Spegne il display
rem Inizia il ciclo infinito
while true
    lettura=adc_read(0)  'LEGGE IL VALORE
                         'ANALOGICO DA RA.0
                         'a 10 bit
                         'Ha un range da 0
                         'a 1023
    delay_ms(100)   'Pausa antistress dell'ADC
                    'e antirimbalzo
    rem Se nessun tasto è premuto
    if (lettura>=0) and (lettura<=57) then
       portb=0 'Spegne il display
    end if
    rem Se si preme il tasto 1
    if (lettura>=58) and (lettura<=170) then
       portb=6 'Visualizza numero 1
    end if
    rem Se si preme il tasto 2
    if (lettura>=171) and (lettura<=283) then
       portb=91 'Visualizza numero 2
    end if
    rem Se si preme il tasto 3
    if (lettura>=284) and (lettura<=396) then
       portb=79 'Visualizza numero 3
    end if
    rem Se si preme il tasto 4
    if (lettura>=397) and (lettura<=509) then
       portb=102 'Visualizza numero 4
    end if
    rem Se si preme il tasto 5
    if (lettura>=510) and (lettura<=623) then
       portb=109 'Visualizza numero 5
    end if
    rem Se si preme il tasto 6
    if (lettura>=624) and (lettura<=737) then
       portb=125 'Visualizza numero 6
    end if
    rem Se si preme il tasto 7
    if (lettura>=738) and (lettura<=849) then
       portb=7 'Visualizza numero 7
    end if
    rem Se si preme il tasto 8
    if (lettura>=850) and (lettura<=966) then
       portb=127 'Visualizza numero 8
    end if
    rem Se si preme il tasto 9
    if (lettura>=967) and (lettura<=1023) then
       portb=111 'Visualizza numero 9
    end if
wend

end.
```

TOP Kit by Futura

Diventa anche tu un **esperto di elettronica** con le nostre **scatole di montaggio!**



Funzionamento assicurato grazie ai dettagliati manuali forniti a corredo dei kit.

Barriera ad infrarossi TX / RX



€ 12,50

Invisibile barriera con portata fino a 4 metri in grado di rilevare il passaggio di persone; utilizzabile per controllare varchi e passaggi obbligati. Per interno. Avvisatore luminoso (LED) e acustico con buzzer da 85 dB. Alimentazione: 1 batteria 9Vdc per elemento; dimensioni trasmettitore 55 x 40 mm, ricevitore 100 x 50 mm.

MK120 **€ 12,50**

Programmatore PIC con textool



Si collega alla porta seriale del PC e consente di programmare la maggior parte dei microcontrollori PIC Microchip, anche quelli con memoria di tipo flash. Software di programmazione PICprog2006 e cavo seriale inclusi. Alimentazione 15 Vdc/300 mA (adattatore di rete non incluso). Dimensioni 132 x 65 x 20 mm.

K8076 **€ 35,00**

VM134 (Versione montata) **€ 59,00**

Modulo amplificatore BF 200 W



Modulo di potenza mono completo di circuito rettificatore. Eroga una potenza di 100Wrms su 4 ohm o 70Wrms su 8 ohm (200W musicali su 4 ohm). Indicato per casse acustiche amplificate, sistemi sub-woofer, ecc. Trasformatore di alimentazione consigliato: cod. 12030 (2x30V, 120VA); dissipatore non incluso.

K8060 **€ 21,00**

VM100
(Versione montata, con dissipatore) **€ 52,00**

Sistema di controllo IR a 2 canali



Composto da un piccolo trasmettitore palmare IR a due canali con elegante contenitore e da un compatto ricevitore (45 x 50 x 15 mm) ad autoapprendimento dotato di due uscite a relè (max 1 A) con funzionamento ad impulso o bistabile, alimentato a 12 Vdc. Portata massima 15 metri.

MK162 (Trasmettitore) **€ 14,00**

MK161 (Ricevitore) **€ 17,00**

Codice MIP **268105**

Chiave a tastiera



Permette di attivare/disattivare un allarme o di aprire un'elettroserratura mediante un codice a 4 cifre. Più di 3000 combinazioni possibili. Uscita a relé (5 A/220 V), alimentazione 9 ÷ 15 Vdc (8 ÷ 12 Vac).

K6400 **€ 28,50**

Stroboscopio

Lampada ad alta luminosità



Crea i tuoi giochi di luce per discoteca! Frequenza di lampeggio regolabile da 2 a 20 Hz mediante potenziometro. Alimentazione 220V/240 Vac, dimensioni 87 x 65 mm.

K2601 **€ 19,00**

Ascolto ambientale 433 MHz



Sistema RF composto da un sensibilissimo microtrasmettitore (FT98TX) in grado di captare anche i rumori più tenui, e da un ricevitore UHF portatile (FT98RX) con potenza d'uscita audio di 0,5 W. Ciascuna unità viene alimentata tramite una batteria a 9 V (non inclusa) ed è completa di contenitore plastico. Auricolari inclusi.

FT98TX (Microtrasmettitore) **€ 25,00**

FT98RX (Ricevitore) **€ 27,00**

Programmatore USB per PIC

€ 65,00



Versione hobbistica del programmatore ICD2 di Microchip. Permette di verificare il funzionamento del software in fase di sviluppo direttamente sulla scheda a microprocessore, operando congiuntamente all'ambiente IDE MPLAB ed è in grado di fornire alimentazione alla scheda target (5 V, 10-40 mA max). Collegamento al PC tramite interfaccia USB (2.0); alimentazione mediante USB; gestione target a 3 V solo con alimentazione esterna; minima tensione target ~ 2 V; funzionamento con tutti i PIC e dsPIC gestiti da ICD2 originale; firmware aggiornabile da MPLAB IDE; dimensioni 100 x 35 x 15 mm.

FT676M (Versione montata) **€ 65,00**

Controllo remoto mediante cellulare GSM

Cellulare escluso



Si abbina a qualsiasi telefono cellulare dotato di display a vista e consente di attivare o disattivare da remoto qualsiasi apparecchiatura elettrica o elettronica con una semplice chiamata. Modalità di funzionamento ON/OFF o attivazione temporizzata (0,5 s ÷1 h); uscite a relé (NO e NC) da 3 A; alimentazione 12 Vdc.

MK160 **€ 13,50**

Finale BF a MOSFET 100 / 150 W

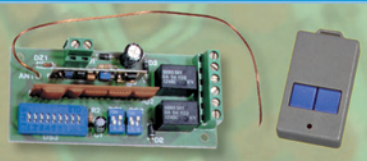


Modulo HI-FI in grado di erogare una potenza di 100 Wrms su 8 ohm o di 150 Wrms su 4 ohm con una tensione di alimentazione duale di 50 Vdc per ramo. Banda passante 5÷80 kHz; distorsione <1%; dimensioni 100 x 75 x 43 mm. Per un funzionamento continuo alla massima potenza, è necessario fissare il modulo ad un dissipatore di dimensioni adeguate.

FT15K **€ 30,00**

FT15M (Versione montata) **€ 40,00**

Radiocomando 2 canali



Sistema radio 433 MHz composto da un ricevitore con codifica Motorola e un trasmettitore bicanale fornito già montato. Funzionamento astabile o bistabile. Alimentazione 12 Vdc.

FT26K-433 **€ 35,00**

FT26M-433 (Versione montata) **€ 40,00**

TX2CSAW **€ 22,00**

Tutti i prezzi si intendono IVA inclusa.

FUTURA ELETTRONICA idee in 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.futurashop.it*

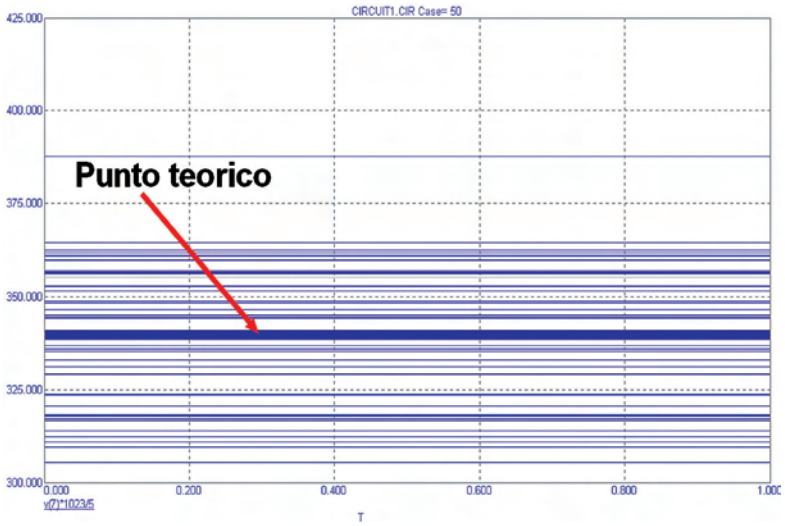


Figura 7
Simulazione Montecarlo relativa alla pressione del tasto n° 3

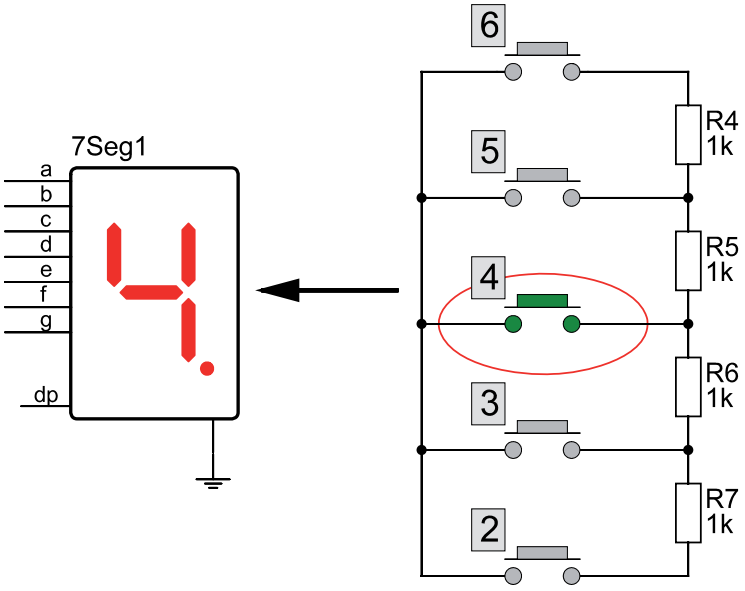


Figura 8
Se si preme il tasto n° 4 il display lo visualizza

Tasto premuto	Valore digitale esatto a 10 bit	Limiti minimi e massimi da considerare
Nessuno	0	0 – 57
1	114	58 – 170
2	226	171 – 283
3	340	284 – 396
4	453	397 – 509
5	566	510 – 623
6	679	624 – 737
7	793	738 – 849
8	908	850 – 966
9	1023	967 – 1023

Tabella 4
Valori limiti per la rilevazione del tasto premuto (tolleranze digitali)

modo da agevolare la precisione di calcolo delle tensioni ottenute. Tuttavia il software, per via degli scaglionamenti previsti dalle clausole IF/ENDIF, rende elastica questa proprietà, prevedendo una sorta di "tolleranza software", con cui esso risponde bene anche alle variazioni relative di ogni componente, pressoché inevitabili. Una correzione digitale insomma che permette di ottenere la corretta risposta da parte del software, anche con valori leggermente diversi delle resistenze.
Intraprendendo una simulazione Montecarlo, si scopre che con tale soluzione, vanno senz'altro bene dei resistori con una tolleranza addirittura del 35%.

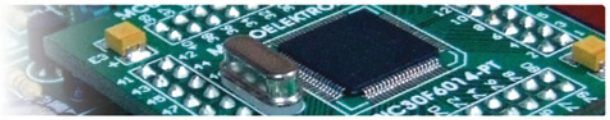
Possibile espansione del sistema e conclusioni

Bene, con questo sistema si risparmiano realmente tante porte del micro, che sono sempre limitate ma preziose. Ma soprattutto, implementando tale tecnica, si possono aumentare il numero dei tasti anche a 40, 50 o più, a patto stavolta di utilizzare delle resistenze precise ed una tensione di riferimento e di alimentazione ancora più precisi. Il tutto utilizzando solamente UNA PORTA del Pic! Una possibile espansione del software, peraltro facilmente realizzabile, potrebbe essere quella di prevedere la pressione "simultanea" dei tasti. In questo caso, grazie al principio della sovrapposizione degli effetti, si possono ottenere diverse tensioni, pigiando contemporaneamente più pulsanti. Ricordate comunque che l'alimentazione del sistema deve essere sempre perfettamente livellata e pari a 5 Volt esatti.

More Info Please!
Inserisci il Codice **268100** alla pagina **www.farelettronica.com/mip**
oppure utilizza il **modulo** a pagina **3**

mikroElektronika

DEVELOPMENT TOOLS | COMPILERS | BOOKS





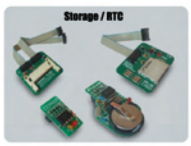
CAN-1 Board - Interface CAN via MCP2551.

CANSPI Board - Make CAN network with SPI interface.

RS485 Board - Connect devices into RS-485 network

Serial Ethernet - Make ethernet network with SPI Interface (ENC28J60).

IrDA2 Board - Irda2 serve as wireless RS232 communication between two MCU's.



CF Board - Easy way to use Compact flash in your design.

MMC/SD Board - Easy way to use MMC and SD cards in your design.

EEPROM Board - Serial EEPROM board via I2C interface.

RTC Board - PCF8583 RTC with battery backup.



ADC Board - 12-bit analog-to-digital converter (ADC) with 4 inputs.

DAC Board - 12-bit digital-to-analog converter (DAC) with SPI.

Keypad 4x4 Board - Add keypad to your application.

Accel. Board - Accel. is an electronic device that measures acceleration forces .

PICFlash

with mikroICD support



PICFlash programmer – an ultra fast USB 2.0 programmer for the PIC microcontrollers. Continuing its tradition as one of the fastest PIC programmer on the market, a new PICFlash with mikroICD now supports more PIC MCUs giving developer a wider choice of PIC MCU for further prototype development.
mikroICD debugger enables you to execute mikroC / mikroPascal / mikroBasic programs on the host PIC microcontroller and view variable values, Special Function Registers (SFR), memory and EEPROM while the program is running.

- All of our products are shipped in special **protective boxes**.
- On-line secure ordering provides fast and safe way of buying our products.

EasyPIC4 Development Board

Complete Hardware and Software solution with on-board USB 2.0 programmer and mikroICD



Following tradition of its predecessor EasyPIC3 as one of the best PIC development systems on the market, EasyPIC4 has more new features for the same price. The system supports 8-, 14, 18, 20, 28 and 40 pin PIC microcontrollers (it comes with a PIC16F877A). **USB 2.0** on-board programmer with **mikroICD** (In-Circuit Debugger) enables very efficient debugging and faster prototype development. Examples in **C**, **BASIC** and **Pascal** language are provided with the board.

LV24-33 Development Board

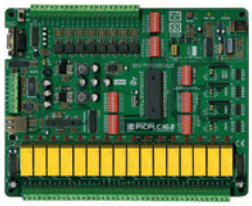
Complete Hardware and Software solution with on-board USB 2.0 programmer and mikroICD



System supports 64, 80 and 100 pins **PIC24F/24H/dsPIC33F** microcontrollers (it comes with PIC24FJ96GA010 - PIC24 16-bit Microcontroller, 96 KB Flash Memory, 8 KB RAM in 100 Pin Package). Examples in BASIC, PASCAL and C are included with(in) the system. You can choose between USB and External Power supply. **LV 24-33** has many features that make your development easy. **USB 2.0** on-board programmer with **mikroICD** (In-Circuit Debugger) enables very efficient debugging and faster prototype development.

PICPLC16B Development Board

Complete Hardware and Software solution with on-board USB 2.0 programmer and mikroICD



PICPLC16B is a system designed for controlling industrial systems and machines. 16 inputs with **optocouplers** and 16 **relays** (up to 10A) can satisfy many industrial needs. The ultra fast **mikroICD** (in-circuit Debugger) enables very efficient debugging and faster prototype development. Features : **RS485**, **RS232**, **Serial Ethernet**, **USB 2.0** on-board programmer and **mikroICD** (In-Circuit Debugger) on-board.

mikroElektronika Compilers

Pascal, Basic and C Compilers for various microcontrollers



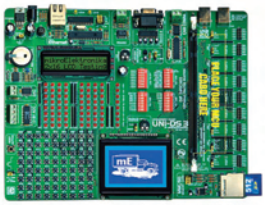
Supporting an impressive range of microcontrollers, an easy-to-use IDE, hundreds of ready-to-use functions and many integrated tools makes MikroElektronika compilers one of the best choices on the market today. Besides mikroICD, mikroElektronika compilers offer a statistical module, simulator, bitmap generator for graphic displays, 7-segment display conversion tool, ASCII table, HTML code export, communication tools for SD/MMC, UDP (Ethernet) and USB , EEPROM editor, programming mode management, etc.

Each compiler has many routines and examples such as EEPROM, FLASH and MMC, reading/writing SD and CF cards, writing character and graphics on LCDs, manipulation of push-buttons, 4x4 keyboard and PS/2 keyboard input, generation of signals and sounds, character string manipulation, mathematical calculations, I2C, SPI, RS232, CAN, USB, RS485 and OneWire communications, Manchester coding management, logical and numerical conversion, PWM signals, interrupts, etc. The CD-ROM contains many already-written and tested programs to use with our development boards.

mikroElektronika manufactures competitive development systems. We deliver our products across the globe and our satisfied customers are the best guarantee of our first-rate service. The company is an official consultant on the PIC microcontrollers and the third party partner of Microchip company. We are also an official consultant and third party partner of Cypress Semiconductors since 2002 and official consultant of Philips Electronics company as well. All our products are RoHS compilant.

Uni-DS 3 Development Board

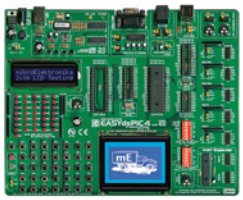
Complete Hardware and Software solution with on-board USB 2.0 programmer



The system supports PIC, AVR, 8051, ARM and PSoC microcontrollers with a large number of peripherals.In order to continue working with different chip in the same development environment, you just need to swich a card. **UNI-DS3** has many features that make your development easy. You can choose between USB or External Power supply. Each MCU card has its own USB 2.0 programmer!

EasydsPIC4 Development Board

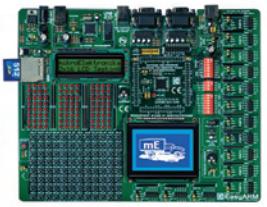
Complete Hardware and Software solution with on-board USB 2.0 programmer and mikroICD



The system supports **18**, **28** and **40** pin microcontrollers (it comes with dsPIC30F4013 general purpose microcontroller with internal 12-bit ADC). **EasydsPIC4** has many features that make your development easy. Many of these already made examples in C, BASIC and PASCAL language guarantee successful use of the system. Ultra fast **USB 2.0** on-board programmer and **mikroICD** (In-circuit Debugger) enables very efficient debugging and faster prototype developing.

EasyARM Development Board

Complete Hardware and Software solution with on-board USB 2.0 programmer



EasyARM board comes with Philips LPC2214 microcontroller. Each jumper, element and pin is clearly marked on the board. It is possible to test most of industrial needs on the system: temperature controllers, counters, timers etc. EasyARM has many features making your development easy. One of them is on-board USB 2.0 programmer with automatic switch between 'run' and 'programming' mode. Examples in **C** language are provided with the board.

EasyAVR4 Development Board

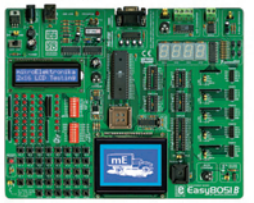
with on-board USB 2.0 programmer



The system supports **8, 20, 28 and 40 pin** microcontrollers (it comes with ATMEGA16). Each jumper, element and pin is clearly marked on the board. It is possible to test most of industrial needs on the system: temperature controllers, counters, timers etc. **EasyAVR4** is an easy-to-use Atmel AVR development system. Ultra fast **USB 2.0** on-board programmer enables very efficient and faster prototype developing. Examples in **BASIC** and **Pascal** language are provided with the board.

Easy8051B Development Board

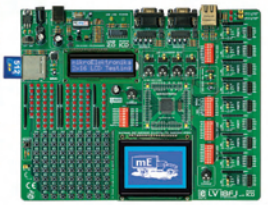
with on-board USB 2.0 programmer



System is compatible with **14, 16, 20, 28 and 40 pin** microcontrollers (it comes with AT89S8253). Also there are **PLCC44 and PLCC32** sockets for 32 and 44 pin microcontrollers. **USB 2.0 Programmer** is supplied from the system and the programming can be done without taking the microcontroller out.

LV 18FJ Development Board

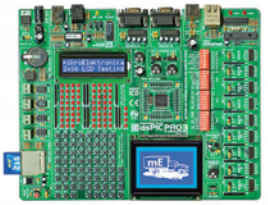
Complete Hardware and Software solution with on-board USB 2.0 programmer and mikroICD



System supports 64, 80 and 100 pin **PIC18FxxJxx** microcontrollers (it comes with PIC18F87J60 - PIC18 Microcontroller with an integrated 10Mbps Ethernet communications peripheral, 80 Pin Package). **LV 18FJ** is easy to use Microchip PIC18FxxJxx development system. **USB 2.0** on-board programmer with **mikroICD** (In-Circuit Debugger) enables very efficient debugging and faster prototype development. Examples in **C**, **BASIC** and **Pascal** language are provided with the board.

dsPICPRO 3 Development Board

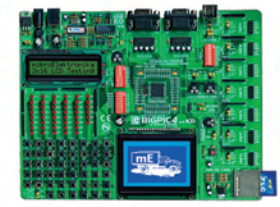
Complete Hardware and Software solution with on-board USB 2.0 programmer and mikroICD



The system supports dsPIC microcontrollers in 64 and 80 pins packages. It is delivered with dsPIC30F6014A microcontroller. **dsPICPRO3** development system is a full-featured development board for the Microchip dsPIC MCU. dsPICPRO3 board allows microcontroller to be interfaced with external circuits and a broad range of peripheral devices. This development board has an on-board USB 2.0 programmer and integrated connectors for MMC/SD memory cards, 2 x RS232 port, RS485, CAN, on-board ENC28J60 Ethernet Controller, DAC etc...

BIGPIC4 Development Board

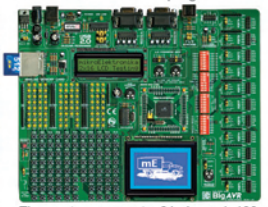
Complete Hardware and Software solution with on-board USB 2.0 programmer and mikroICD



Following tradition of its predecessor BIGPIC3 as one of the best 80-pin PIC development systems on the market, **BIGPIC4** continues the tradition with more new features for the same price. System supports the latest (64) and 80-pin PIC microcontrollers (it is delivered with PIC18F8520). Many of these already made examples in **C**, **BASIC** and **Pascal** language guarantee successful use of the system. Ultra fast on-board programmer and **mikroICD** (In-circuit Debugger) enables very efficient debugging and faster prototype developing.

BIGAVR Development Board

with on-board USB 2.0 programmer



The system supports 64-pin and 100-pin AVR microcontrollers (it is delivered with ATMEGA128 working at 10MHz). Many already made examples guarantee successful use of the system. **BIGAVR** is easy to use Atmel AVR development system. BIGAVR has many features that makes your development easy. You can choose between USB or External Power supply. BIGAVR also supports Character LCD as well as Graphic LCD.

EasyPSoC3 Development Board

with on-board USB 2.0 programmer



The system supports **8, 20, 28** and **48 pin** microcontrollers (it comes with CY8C27843). Each jumper, element and pin is clearly marked on the board. **EasyPSoC3** is an easy-to-use PSoC development system. On-board **USB 2.0** programmer provides fast and easy in-system programming.

http://www.mikroe.com/en/distributors/

Find your distributor: UK, USA, Germany, Japan, France, Greece, Turkey, Italy, Slovenia, Croatia, Macedonia, Pakistan, Malaysia, Austria, Taiwan, Lebanon, Syria, Egypt, Portugal, India.

Please visit our website for more info http://www.mikroe.com

SOFTWARE AND HARDWARE SOLUTIONS FOR EMBEDDED WORLD

Codice MIP **268107**

Sesta parte: *Settembre 2007*
Gli inseguitori di tensione

Settima parte: *Ottobre 2007*
Il preamplificatore equalizzatore

Ottava parte: *Novembre 2007*
Il preamplificatore microfonico

Nona parte: *Dicembre 2007*
Filtri per Hi-Fi: progetto dei filtri antiscratch e antirumble

Imparare a progettare

Gli equalizzatori ambientali sono preamplificatori in grado di apportare notevoli correzioni alla banda audio. Ecco come progettarne uno.

Generalità

Nell'ascolto di un brano musicale si avverte talvolta o una carenza dei bassi, o degli alti o delle medie frequenze della banda audio. Ciò può essere determinato da vari fattori come, per esempio, la non perfetta idoneità dell'acustica ambientale, o la non corretta disposizione della casse acustiche o anche soltanto l'usura da cui può essere affetto il supporto in riproduzione.
Per ovviare a questi inconvenienti si può ricorrere agli equalizzatori ambientali che altro non sono che preamplificatori in grado di modificare la risposta su più frequenze all'interno della banda fisiologica (20 Hz ÷ 20 kHz). Queste apparecchiature, che si dispongono fra lo stadio di preamplificazione e lo stadio finale, se ben progettate sono senz'altro in grado di eliminare o quantomeno correggere i problemi di cui si è detto.
Sull'inserzione in una catena hi-fi di questi dispositivi c'è sempre stata una sorta di polemica fra coloro che li ritenevano indispensabili e coloro che, da puristi, ritenevano, al contrario, che un brano musicale dovesse essere ascoltato così come era stato originariamente trasferito su un qualsiasi supporto in grado di consentirne la riproduzione.
È probabile che le due scuole di pensiero siano meno distanti di quanto non ritengano i rispettivi adepti. Se infatti è opportuno ascoltare una qualsiasi musica così come è stata prodotta, per esempio da un'orchestra, è altrettanto opportuno che, ove si riscontrino le condizioni a cui prima si è accennato, si possa avere la possibilità di una correzione che, in fondo, non fa che migliorare la qualità dell'ascolto.

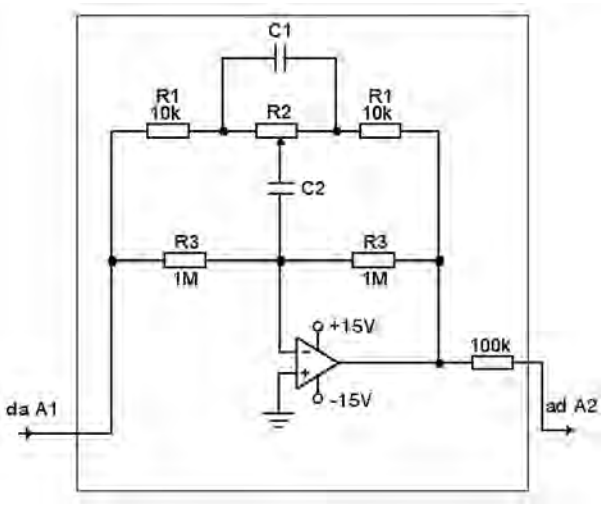


Figura 1
Modulo da realizzarsi in dieci esemplari per realizzare un equalizzatore ambientale a dieci bande di frequenza

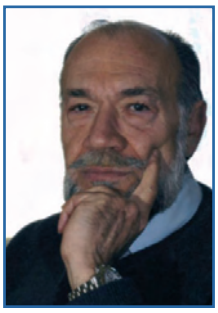
E tutto ciò a prescindere da una considerazione che ha valenza quasi universale facendo riferimento alla sensibilità dell'ascoltatore, sensibilità che è sempre del tutto soggettiva.

Il circuito

Il circuito di base è riportato nella figura 1. Per realizzare l'equalizzatore a dieci bande qui proposto sono sufficienti dieci moduli eguali a quello della figura.

Si perviene così al circuito completo esposto nella figura 2 dove si notano gli amplificatori operazionali A_1 posto in ingresso e A_2 posto in uscita, comuni a tutti e dieci i canali.
Questi amplificatori sono necessari per svincolare le impedenze di uscita del preamplificatore e di ingresso dello stadio finale da qualsiasi problema. L'ingresso dell'equalizzatore presenta così un'alta impedenza (circa 100 kΩ) al fine di non caricare lo stadio precedente, mentre presenta altresì una bassa impedenza di uscita (pochi ohm) in modo da comportarsi come ottimo generatore per lo stadio finale.

L'equalizzatore ambientale



di Nico Grilloni

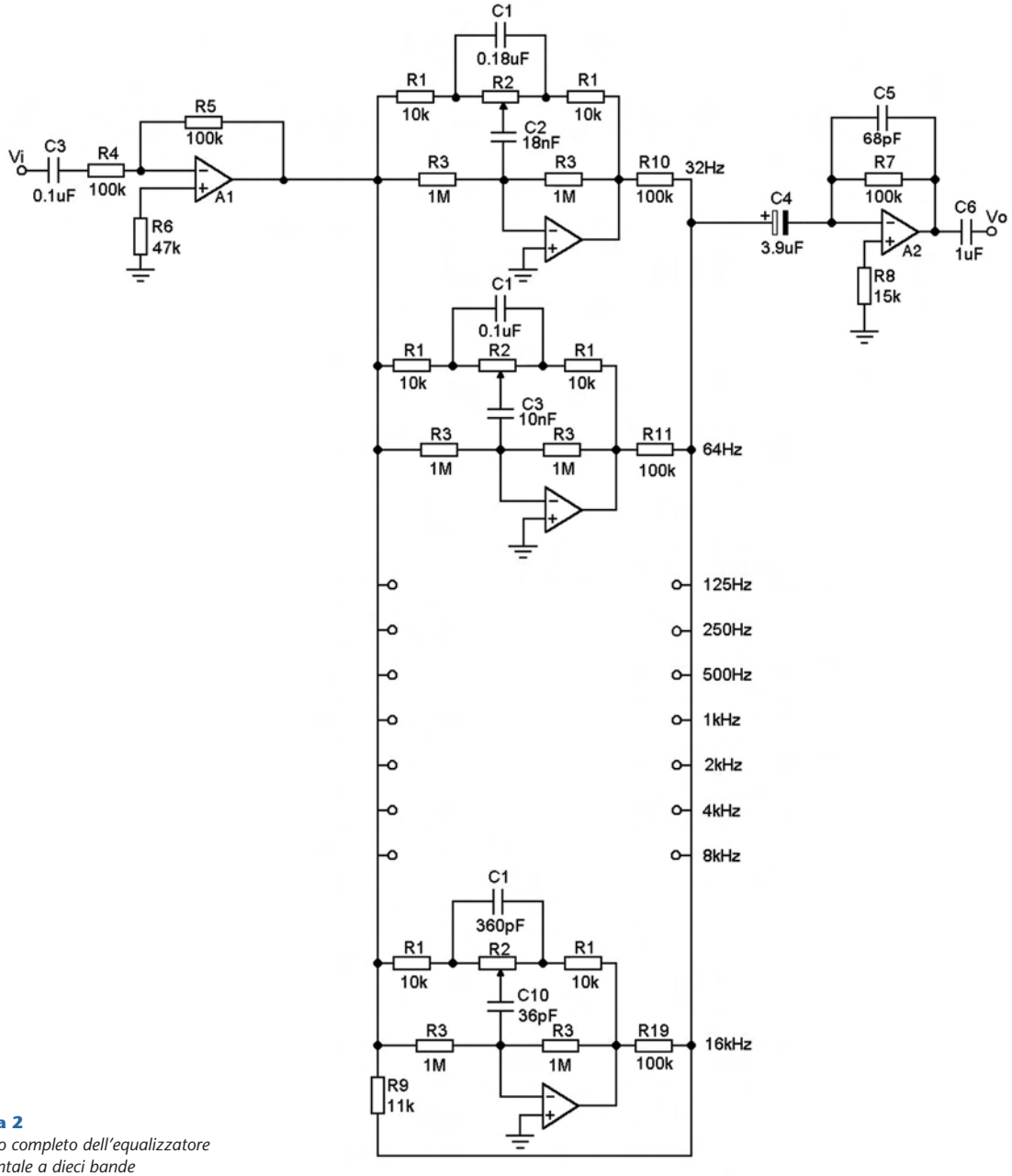


Figura 2
Circuito completo dell'equalizzatore ambientale a dieci bande

Si tenga presente che i valori delle resistenze R_1, R_3 e dei potenziometri R_2, nonché della resistenza da 100 kΩ, sono eguali per tutti i moduli, mentre variano i valori delle capacità C_1 e C_2 secondo quanto riportato nella tabella seguente.

Dalle figure 1 e 2 si evince che il circuito è in fondo costituito da filtri passa-banda che alla frequenza f_o di centrobanda, in funzione della posizione del cursore dei potenziometri R_2, forniscono un'attenuazione e un'esaltazione massima di 12 dB.

Le frequenze di centrobanda distano fra loro di un'ottava, il che significa che si va dalla frequenza

F_o (Hz)	C_1	C_2
32	0,18 µF	0,018 µF
64	0,1 µF	0,01 µF
125	0,047 µF	0,0047 µF
250	0,022 µF	0,0022 µF
500	0,012 µF	0,0012 µF
1k	0,0056 µF	560 pF
2k	0,0027 µF	270 pF
4k	0,0015 µF	150 pF
8k	680 pF	68 pF
16k	360 pF	36 pF

Tabella 1
In funzione della frequenza f_o sono riportati i valori da attribuire alle capacità C_1 e C_2

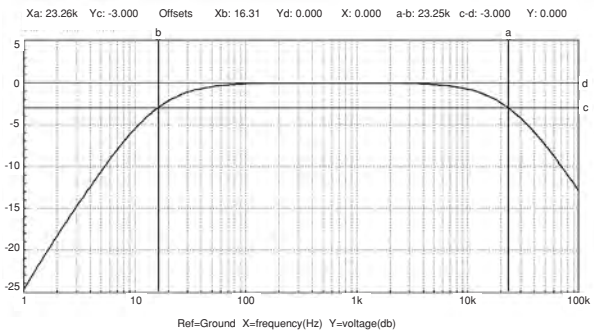


Figura 3
Risposta in frequenza dell'equalizzatore di cui alla figura 2 con i cursori dei potenziometri posti a metà corsa

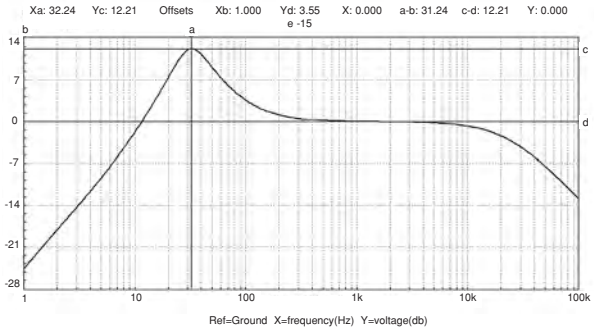


Figura 4
Dalla curva della risposta in frequenza si vede che il guadagno dei vari moduli – così come l'attenuazione – è di 12 dB

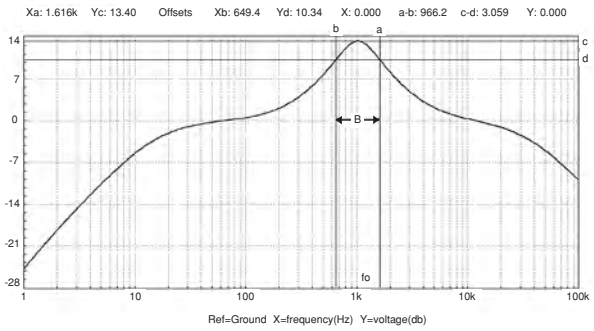


Figura 5
Il fattore Q di ciascun filtro ha valore unitario. Ciò significa che l'ampiezza della banda passante B è pressoché eguale alla frequenza di centrobanda f_o

f_o = 32 Hz alla frequenza f_o = 18 kHz raddoppiando di volta in volta la frequenza precedente.
Si tenga presente che è comunque possibile modificare la frequenza di centro banda di uno o più filtri. Per far ciò è sufficiente rifarsi alle espressioni:
$C_2 = (5{,}513 \times 10^7) / f_o$
$C_1 = 10\ C_2$
Per f_o = 1 kHz, per esempio, si ha:
$C_2 = (5{,}513 \times 10^7)/1000 = 551{,}3$ pF
Questo valore si approssimerà al valore normalizzato di 560 pF.
$C_1 = 10 \times 560$ pF = 5600 pF = 0,0056 µF
La figura 3 mostra la risposta in frequenza dell'equalizzatore quando i cursori dei dieci potenziometri sono posti a metà corsa. Il guadagno in banda passante è unitario come mostra la posizione del marker orizzontale *d* che è su 0 dB (livello del segnale di ingresso), mentre la banda passante si estende a – 3 dB da 16,31 Hz – posizione del marker verticale *b* – a 23,26 kHz – posizione del marker verticale *a*. Volendo modificare la frequenza di taglio superiore si dovrà modificare il valore della capacità del condensatore C_5 tenendo presente che all'aumentare della capacità si ha una frequenza di taglio via via minore (e viceversa).
Che il guadagno – così come l'attenuazione – sia di 12 db si vede dalla risposta in frequenza dei cui alla figura 4 ricavata in simulazione per il modulo a frequenza di centrobanda f_o = 32 Hz. Per f_o eguale a circa 32 Hz (*Xa* = 32,24 Hz) il marker orizzontale *c* dista di 12,21 dB dal marker *d* posto a 0 dB (livello del segnale di ingresso).
Si legge infatti: (*c* – *d*) = 12,21 dB.
Il fattore *Q* dei vari filtri passabanda che realizzano questo equalizzatore ambientale ha valore pressoché unitario. A questo proposito si consideri, per esempio, la risposta in frequenza del modulo relativo alla frequenza di centrobanda di 1 kHz riportata nella figura 5. Si può constatare che a – 3 dB – posizione reciproca dei marker *c* e *d* – la banda passante *B* è di 966,2 Hz.
Essendo comunque:
$Q = f_o / B$

per *Q* si ha il valore:
$Q = 1000 / 966{,}2 = 1{,}04$

I potenziometri di vari moduli hanno tutti il valore di 500 kΩ.
Non resta infine che da augurare buon lavoro ai lettori che vorranno realizzare questo ottimo equalizzatore.

More Info Please!
Inserisci il Codice **268108** alla pagina
www.farelettronica.com/mip
oppure utilizza il **modulo** a pagina **3**

MARCHEFIERE
ERF ENTE REGIONALE PER LE MANIFESTAZIONI FIERISTICHE

Quartiere Fieristico CIVITANOVA MARCHE (MC)

8-9 dicembre 2007

25ª 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

"4° 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

Codice MIP 268111

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*