www.elettronicain.it

€ 4,50

Elettronica Innovativa

Elettronica In

Mensile di elettronica applicata, attualità scientifica, novità tecnologiche.

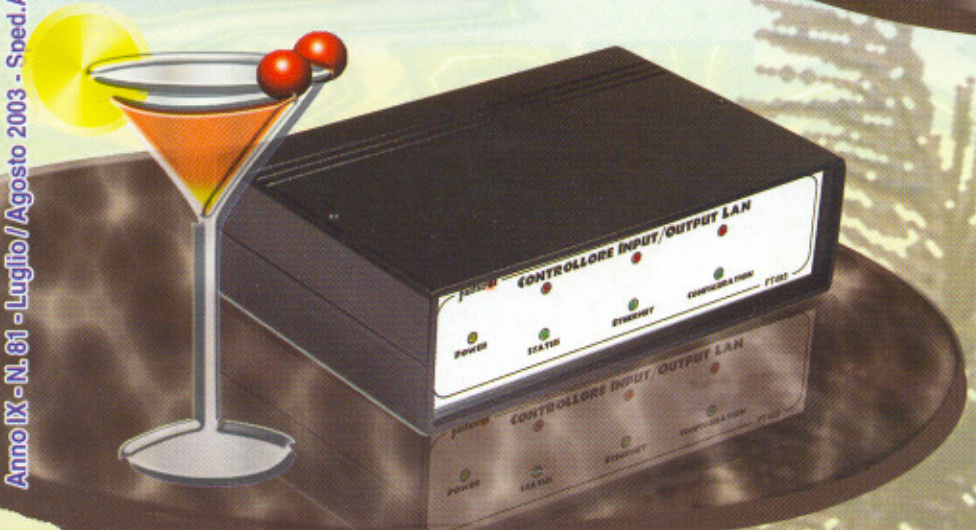
81

9 771124 854008 30081

- Indicatori di potenza per amplificatore Stereo
- Costruire e programmare i robot: Spider
- Costruzione rapida di PCB tramite fresatura
- Modulo avanzato Voice Extreme

Stimolatore per Elettroagopuntura

Controller Domotico gestibile via Internet



• ESCLUSIVO •
CORSO DI PROGRAMMAZIONE VOICE EXTREME

Anno IX - N. 81 - Luglio / Agosto 2003 - Sped.Abb.Post. 45% Art. 2 comma 20/B Legge 662/96 - Milano



Pag. 30

Pag. 44

13 **COSTRUZIONE RAPIDA DI PCB TRAMITE FRESATURA**

Macchina utensile a controllo numerico ideale per la realizzazione delle basette sulle quali vengono montati i circuiti elettronici. Mediante fresatura permette la creazione delle piste in rame; per la foratura delle basette può invece lavorare come trapano. Compatibile con i file in formato: HPGL, Excellon, Syeb&Meiere ISO G-code 2D/3D.

17 **MODULO AVANZATO VOICE EXTREME**

Un circuito in grado di parlare e di comprendere comandi vocali. Realizzato partendo da un microcontrollore ad 8 bit tipo RSC-364 che per funzionare necessita solamente di una memoria Flash da 2 MB, di un quarzo e di pochi altri componenti. Il tutto trova posto su un piccolo circuito stampato di circa 4 x 4 cm.

Sommario

ELETTRONICA IN
www.elettronicain.it

Rivista mensile, anno IX n. 81
LUGLIO / AGOSTO 2003

Direttore responsabile:
Arsenio Spadoni
(Arsenio.Spadoni@elettronicain.it)
Responsabile editoriale:
Carlo Vignati
(Carlo.Vignati@elettronicain.it)
Redazione:
Paolo Gaspari, Clara Landonio, Boris Landoni, Angelo Vignati, Andrea Silvello, Alessandro Landone, Marco Rossi, Davide Ferrario, Alessia Sfulcini, Andrea Colombo.
(Redazione@elettronicain.it)
Ufficio Pubblicità:
Monica Premoli (0331-577976).
DIREZIONE, REDAZIONE, PUBBLICITA':
VISPA s.n.c.
v.le Kennedy 98
20027 Rescaldina (MI)
telefono 0331-577976
telefax 0331-466686
Abbonamenti:
Annuo 10 numeri Euro 36,00
Estero 10 numeri Euro 78,00
Le richieste di abbonamento vanno inviate a: VISPA s.n.c., v.le Kennedy 98, 20027 Rescaldina (MI) tel. 0331-577976.
Distribuzione per l'Italia:
SO.DI.P. Angelo Patuzzi S.p.A.
via Bettola 18
20092 Cinisello B. (MI)
telefono 02-660301
telefax 02-66030320
Stampa:
ROTO 2000
Via Leonardo da Vinci, 18/20
20060 CASARILE (MI)
Elettronica In:
Rivista mensile registrata presso il Tribunale di Milano con il n. 245 il giorno 3-05-1995.
Una copia Euro 4,50, arretrati Euro 9,00
(effettuare versamento sul CCP n. 34206207 intestato a VISPA snc)
(C) 1995 ÷ 2002 VISPA s.n.c.
Spedizione in abbonamento postale 45% - Art.2 comma 20/b legge 662/96 Filiale di Milano.
Impaginazione e fotolito sono realizzati in DeskTop Publishing con programmi Quark XPress 4.1 e Adobe Photoshop 6.1 per Windows.
Tutti i diritti di riproduzione o di traduzione degli articoli pubblicati sono riservati a termine di Legge per tutti i Paesi.
I circuiti descritti su questa rivista possono essere realizzati solo per uso dilettantistico, ne è proibita la realizzazione a carattere commerciale ed industriale. L'invio di articoli implica da parte dell'autore l'accettazione, in caso di pubblicazione, dei compensi stabiliti dall'Editore. Manoscritti, disegni, foto ed altri materiali non verranno in nessun caso restituiti. L'utilizzazione degli schemi pubblicati non comporta alcuna responsabilità da parte della Società editrice.

25 **INDICATORE DI POTENZA PER AMPLIFICATORE STEREO**

Misuratore a led utilizzabile per visualizzare la potenza di uscita di un amplificatore. Nella configurazione stereo la scala di ogni canale è composta da 12 led e indica potenze comprese tra 0,001W e 100W. Nella configurazione mono è possibile collegare due indicatori in serie per complessivi 24 led con scala della potenze compresa tra 0,001W e 400W.

30 **STIMOLATORE PER ELETTROAGOPUNTURA BIFASTICA**

Strumento progettato appositamente per l'applicazione della terapia medica dell'elettroagopuntura. Prevede la regolazione dell'intensità della corrente utilizzata per il trattamento e la possibilità di applicare, mediante un opportuno manipolo, l'elettroagopuntura alla propria persona. Inoltre, combinando il manipolo ed un elettrodo, il trattamento può essere applicato anche a terzi. Completo di batterie ricaricabili che ne consentono l'utilizzo in qualsiasi luogo e di un circuito che provvede alla ricarica delle stesse tramite un alimentatore esterno a 12 Volt.

44 **CONTROLLER DOMOTICO GESTIBILE VIA INTERNET**

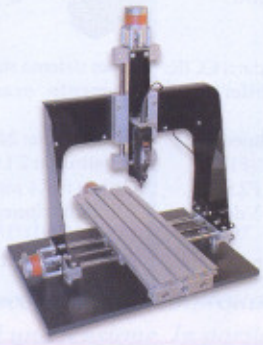
Dispositivo per la gestione di ingressi e uscite digitali o analogiche che può essere controllato a distanza tramite Internet. Ideale per applicazioni di controllo remoto di un'abitazione. Munito di un'uscita di allarme e di una porta I²CBus utilizzabile per espandere il sistema con nuovi ingressi o uscite, sia digitali che analogiche.

58 **COSTRUIRE E PROGRAMMARE I ROBOT: SPIDER**

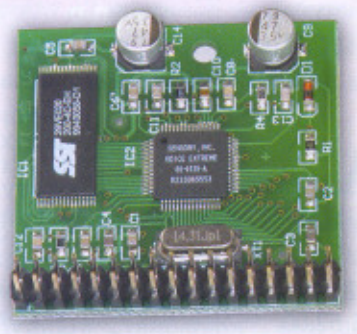
Terminiamo questo mese la descrizione del terzo robot: Spider, il ragno a sei zampe. Analizziamo insieme tutti i segreti del software che permettono a questo exapod di muoversi e di riconoscere gli ostacoli.

73 **CORSO DI PROGRAMMAZIONE VOICE EXTREME**

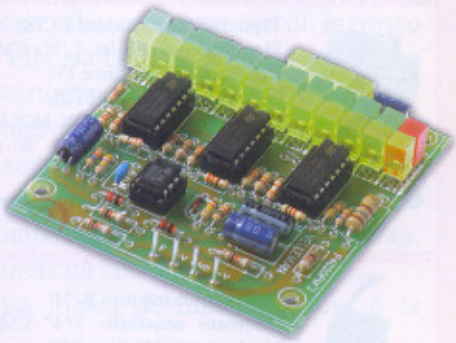
Corso di utilizzo e programmazione dell'integrato Voice Extreme della Sensory. Questo chip, un particolare microcontrollore ad 8 bit, è in grado di parlare e di comprendere comandi vocali. Impareremo a programmare il VE-IC realizzando applicazioni che utilizzano la voce come mezzo di controllo per apparecchiature o sistemi di sicurezza.
Decima e ultima puntata.



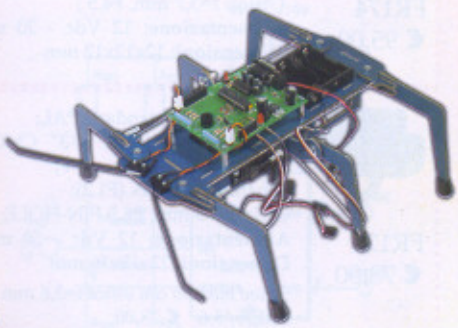
13



17



25



58



73

Editoriale

Come ogni anno il numero estivo di *Elettronica In* è più interessante e ricco che mai. Il progetto che ha richiesto il maggiore impegno, e del quale andiamo più fieri, è sicuramente il **Controller Domotico gestibile via Internet**; un apparato che mette a disposizione una serie di ingressi e uscite, sia digitali che analogiche, ideale per la realizzazione di un controllo domotico di un'abitazione o di un ufficio. Sul fronte **Voice Extreme** segnaliamo il progetto del **Modulo Avanzato**; un circuito che nello spazio di circa 4x4 cm permette di realizzare sistemi in grado di parlare e comprendere comandi vocali.
Nel fascicolo di giugno abbiamo presentato un amplificatore stereo da 600W; in questo numero proponiamo la realizzazione di un **VU-Meter** (utilizzabile sia nella versione mono che in quella stereo) adatto a questo amplificatore.
Dopo una serie di elettrostimolatori presentati nei mesi scorsi, e in considerazione dell'interesse per questi argomenti, proponiamo questo mese il progetto di uno **Stimolatore per l'Elettroagopuntura Bifasica**, ideale per risolvere alcuni problemi fisici o per eliminare alcuni vizi (per esempio il fumo...) che minacciano la nostra salute.
Questo mese termina, dopo sette puntate, la descrizione dei robot programmabili; in quest'ultimo appuntamento analizziamo tutti i segreti del **software di Spider.** In ogni caso, di robotica continueremo sicuramente ad occuparci nei prossimi fascicoli. Un altro corso che termina questo mese, dopo dieci puntate, è quello di **Programmazione del sistema Voice Extreme.**
Prima di lasciarvi alla lettura di questo numero, vogliamo segnalare che è disponibile il **catalogo 2° Semestre 2003** della ditta **Futura Elettronica** la quale produce le scatole di montaggio dei progetti pubblicati: per richiederlo è sufficiente utilizzare il coupon che trovate in terza di copertina.



Arsenio Spadoni

[elencoInserzionisti]

BIAS
Blu Nautilus
CPM Elettronica
Elle Erre
Eurocom Pro
Fiera di Gonzaga
Fiera di Novegro
Futura Elettronica
Idea Elettronica
Millenium
RM Elettronica
Tommesani
www.pianetaelettronica.it

LETTERE

Parola ai lettori

Window Comparator

Ho la necessità di controllare il livello di una tensione. In particolare, devo verificare che questa sia compresa all'interno di un determinato intervallo, specificabile dall'esterno. Mi potete aiutare o comunque dare dei consigli?

Luca Gadioli - Brescia

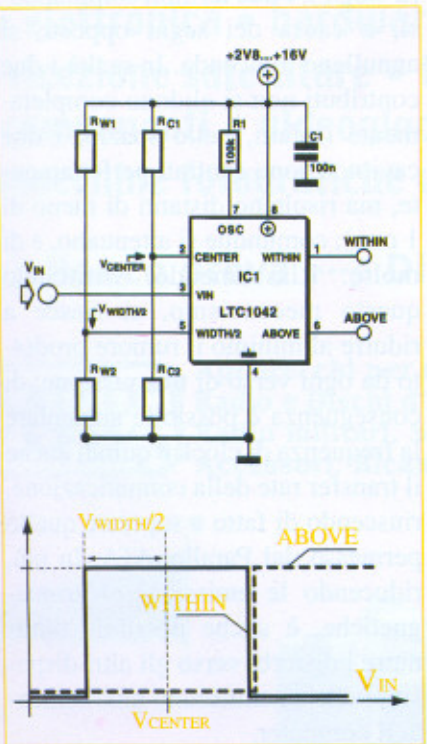
Il nostro consiglio è di utilizzare l'integrato LTC1042 prodotto dalla Linear Technology. Questo è infatti un "Window Comparator", cioè un chip progettato apposta per verificare che una tensione di ingresso VIN sia compresa all'interno di un certo range. L'intervallo viene definito impostando un livello di tensione centrale (VCENTER) e l'ampiezza del range (in realtà viene impostata la metà dell'ampiezza, indicata con VWIDTH/2).

La logica di funzionamento è la seguente: fintanto che VIN risulta all'interno della finestra specificata, l'uscita WITHIN è alta; altrimenti l'uscita è bassa.

In più, è presente anche una seconda uscita (ABOVE) che lavora invece secondo questa logica: ABOVE è alta quando VIN supera il limite massimo della finestra impostata.

Una caratteristica particolare è che, scambiando i piedini VIN e CENTER, l'uscita WITHIN continua a lavorare secondo la logica precedente, mentre la seconda uscita (ABOVE) non indica più che il segnale di ingresso ha superato il limite massimo della finestra, ma indica che è sceso al di sotto del limite minimo. Negli schemi che abbiamo pubblicato trovi una possibile applicazione: le resistenze RW1/RW2 e RC1/RC2, come è abbastanza semplice intuire, sono utilizzate come partitore di tensione per selezionare i livelli di VCENTER e VWIDTH/2.

Il dispositivo è munito di un oscillatore interno (piedino 7 - OSC) utilizzato per campionare il segnale esterno: la rete R1/C1 determina la frequenza dell'oscillatore.



Codice Manchester

Che cos'è la codifica Manchester?

Marco Varalta - Modena

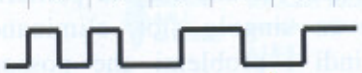
La codifica Manchester viene utilizzata per trasmettere dei bit (per esempio nelle reti Ethernet o negli hard disk). Il vantaggio di tale codifica è che trasporta sia i dati che il segnale di clock.

Infatti un messaggio quando viaggia su di un cavo, altro non è che una sequenza di bit, che possono assumere valore 0 o 1. In alcuni casi particolari, si può verificare il caso in cui si debbano trasmettere lunghe sequenze dello stesso simbolo, o tutti 0 o tutti 1. Se si utilizza una trasmissione sincrona (che non prevede bit di start e stop) in ricezione sarebbe difficile recuperare il sincronismo.

È stata quindi introdotta la codifica Manchester che prevede che a ogni simbolo corrisponda una transizione. Nel dettaglio: uno 0 logico viene rappresentato con un passaggio da stato basso a alto; viceversa un 1 logico è rappresentato con una transizione da alto a basso. In questo modo per ogni bit trasmesso si garantisce un cambiamento di stato logico, e quindi il recupero del sincronismo in ricezione risulta più semplice. L'unico svantaggio di tale codifica è che viene richiesta una frequenza di simbolo doppia rispetto al necessario; anche la banda occupata dalla trasmissione risulterà pertanto doppia.

Esempio di codifica manchester

1	1	0	1	0



Servizio consulenza tecnica

Per ulteriori informazioni sui progetti pubblicati e per qualsiasi problema tecnico relativo agli stessi è disponibile il nostro servizio di consulenza tecnica che risponde allo 0331-577982. Il servizio è attivo esclusivamente il lunedì e il mercoledì dalle 14.30 alle 17.30.

Specifiche Serial ATA

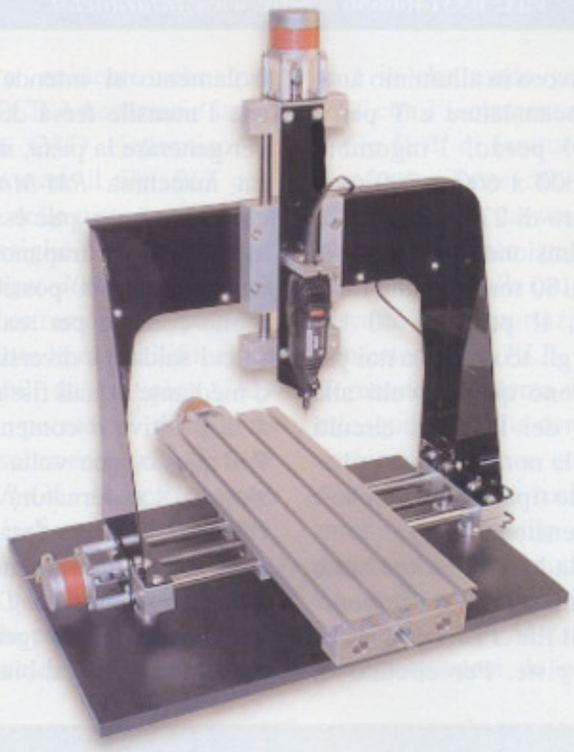
Nello scorso numero della rivista, nella rubrica dedicata al web, avete pubblicato il Sito del Serial Ata Working Group, l'ente che ha l'obiettivo di definire le specifiche del nuovo standard SATA. Ho visitato il sito e l'ho trovato molto interessante, ma non ho ben compreso che cosa comporteranno di innovativo le nuove specifiche. Potete darmi maggiori indicazioni?

Mattia Acampora - Salerno

Le nuove specifiche Serial ATA sostituiranno, a breve, le "vecchie" interfacce Parallel ATA che realizzano la connessione tra hard disk e schede madri all'interno dei PC. Questo nuovo tipo di interfaccia offre, rispetto al vecchio standard, alcuni vantaggi: innanzitutto i cavi di collegamento sono meno ingombranti e possono essere più lunghi delle classiche "piattine" utilizzate nell'interfaccia ATA parallela, essendo formati da 7 fili invece dei 40 necessari per il collegamento parallelo. Ciò consente di agevolare il flusso d'aria all'interno dei case dei Pc, migliorando il raffreddamento dei componenti interni (diventato critico negli ultimi tempi con l'introduzione di nuovi processori sempre più veloci, ma che richiedono anche un maggior smaltimento termico) e facilitando anche l'installazione di nuovi dischi. Un altro importante vantaggio di Serial ATA è che risulta completamente compatibile a livello software con lo standard parallelo; i nuovi dischi possono quindi essere tranquillamente utilizzati con gli attuali sistemi operativi e applicazioni, per esempio software di backup o di partizionamento. Inoltre, lo standard SATA prevede la connessione di una singola periferica ad un singolo slot, eliminando quindi i problemi che possono insorgere utilizzando la configurazione master e slave tipica del collegamento Parallel ATA. Un'altra funzionalità da non sottovalutare è la possibilità di collegare e scollegare "a caldo" le unità (*hot swap*). L'interfaccia Serial ATA supporta una velocità di trasmissione massima superiore a quella consentita dall'attuale standard parallelo. Al momento, infatti, utilizzando le nuove specifiche, si possono già raggiungere transfer rate di 150 MByte/s, contro i 100 dello standard ATA/100 e i 133 di quello ATA/133. Si è comunque ancora in una prima fase; nei prossimi anni il nuovo standard verrà sicuramente migliorato permettendo quindi di raggiungere maggiori velocità di cifra (300 e 600 Mbyte/sec). L'importanza del nuovo standard diventa immediatamente chiara se si considera che, nei moderni PC, uno dei maggiori colli di bottiglia che ne limitano le prestazioni è proprio la velocità di trasmissione tra memorie di massa (hard disk) e CPU. A questo punto facciamo una precisazione: quando abbiamo detto che la nuova interfaccia consente transfer rate maggiori rispetto alla vecchia dovrebbe esservi sorto un dubbio. Infatti, a livello teorico, ci hanno sempre insegnato che, a pari frequenza di clock, la trasmissione parallela è più veloce di quella seriale; il primo tipo trasmette infatti *N* bit informativi per ciclo di clock, rispetto al singolo trasmesso dal tipo seriale. Il punto è che, utilizzando i collegamenti seriali, si riesce ad aumentare in maniera considerevole la frequenza del clock trasmissivo. Il motivo è essenzialmente dovuto a problemi di irradiazioni elettromagnetiche. Infatti, ogni filo che trasporta informazioni, si comporta come una piccola "antenna" che irradia del "disturbo" nei fili vicini. Nello standard parallelo sono presenti molti più cavi per ogni collegamento; viene quindi generato più rumore nel singolo filo, limitando di fatto la frequenza del clock. Nello standard seriale, invece, sono presenti solo due fili che realizzano il collegamento; il disturbo generato risulta quindi minore e sono permesse più alte frequenze di clock. Inoltre, all'interno dello standard Serial ATA, è stato utilizzato il seguente "trucco": ogni collegamento è costituito da 4 fili (2 per verso) che trasportano l'informazione. In ogni filo che realizza la coppia, sono trasportati gli stessi dati, ma invertiti (sfasati di 180°) uno rispetto all'altro. In questo modo, i due fili genereranno disturbi uno opposto all'altro (se infatti il primo filo genererà il campo *+noise*, il secondo genererà *-noise*). I due termini sommandosi, a causa dei segni opposti, si annullano a vicenda. In realtà i due contributi non si elidono completamente (infatti, nello spazio, i due cavi non sono centrati perfettamente, ma risultano distanti di meno di 1 mm); comunque si attenuano, e di molto. Riassumendo, sfruttando questo meccanismo, si riesce a ridurre al minimo il rumore prodotto da ogni verso di trasmissione; di conseguenza è possibile aumentare la frequenza di clock e quindi anche il transfer rate della comunicazione, riuscendo di fatto a superare quello permesso dal Parallel ATA. In più, riducendo le emissioni elettromagnetiche, è anche possibile diminuire i disturbi verso gli altri dispositivi che costituiscono internamente il computer.

Costruzione rapida di PCB tramite Fresatura

di Ronchini Massimo



Macchina utensile a controllo numerico ideale per la realizzazione delle basette che compongono i circuiti elettronici. Mediante fresatura permette la creazione delle piste in rame; per la foratura delle basette può invece lavorare come trapano. Compatibile con i file in formato: HPGL, Excellon, Syeb&Meier e ISO G-code 2D/3D.

Tutti gli appassionati di elettronica sanno che una delle operazioni più importanti e difficili per realizzare e mettere in pratica le proprie idee e i propri progetti è quella di costruire le basette che compongono i circuiti stampati. Le basette devono "trasferire" nella realtà i concetti teorici che sono stati inseriti all'interno dello schema elettrico; le stesse devono contenere sia le piste in rame che realizzano i collegamenti tra i diversi componenti elettronici sia i fori previsti per le relative operazioni di saldatura o fissaggio del circuito a dispositivi o contenitori esterni. Oggigiorno per la realizzazione delle basette esistono diverse tecniche; generalmente tutte quante partono da un supporto plastico di spessore variabile a seconda dell'applicazione, di cui un singolo lato (nel caso di circuiti singola faccia) o entrambi i lati (nel caso di circuiti doppia faccia) sono completamente rivestiti da un sottile spessore di rame. Successivamente il rame in "eccesso" viene eliminato mediante diversi procedimenti in modo da disegnare sul circuito le piste ramate. A questo scopo, una tecnica che sempre più spesso viene utilizzata è quella di realizzare un'asportazione diretta del rame, mediante >

*I prezzi si intendono IVA compresa

Per il MATERIALE

La RM-MiniMill è disponibile in due diverse configurazioni: con avanzamenti trapeziodali (cod. RM-MiniMill/T DOS; 4.560,00 Euro) oppure con avanzamenti a ricircolo di sfere (cod. RM-MiniMill/R DOS; 5.310,00 Euro). Entrambe le configurazioni comprendono la struttura meccanica auto-portante; l'elettronica di controllo; il software 3D per ambiente DOS (PCNC 3.1 in lingua italiana); i due motori passo passo da 2A; il supporto standard per mini trapano. Separatamente è disponibile il Mini Trapano Fresa Multivelocità al prezzo di 114,00 Euro. Come optional è disponibile il software di controllo per Windows in lingua italiana e una serie di accessori: per l'elenco completo consultare il sito internet: www.ronchinimassimo.com.

Il software di progettazione elettronica TARGET 3001! è disponibile in 3 diverse versioni che si differenziano solamente per il numero massimo di pin e di strati che sono in grado di gestire: Versione 10 LIGHT 400pins/2 strati (cod. T3L; 49,00 Euro); Versione 10 ECONOMY 1000 pins/2 strati (cod. T3L; 549,00 Euro); Versione 10 PROFESSIONAL pins illimitati/100 strati (cod. T3P; 1.599,00 Euro); Versione 10 SCHOOL PROFESSIONAL pin illimitati 100 strati Riservata ad Istituti Tecnici (cod. T3S; 249,00 Euro).

Tutti i prezzi indicati si intendono IVA compresa.
Il materiale va richiesto a: Ronchini Massimo s.a.s., Via Capitini 2, Sala Baganza (Parma)
Tel: 0521835690 ~ Fax: 0521835690 ~ Sito: *www.ronchinimassimo.com*
Il software TARGET 3001! è invece disponibile presso: Futura Elettronica, V.le Kennedy 96, 20027 Rescaldina (MI). Tel: 0331576139 ~ Fax: 0331466686 ~ Sito: *www.futuranet.it*

fresatura. Per realizzare questa operazione si utilizzano macchine a controllo numerico (CNC); in commercio se ne trovano di vari modelli caratterizzati da diverse funzioni, complessità e prezzo.

In queste pagine vi presentiamo la *RM-MiniMill*, una macchina che appartiene a questa categoria di impianti CNC. La *RM-MiniMill* non ha nessuna pretesa di competere con i più complessi e costosi sistemi che si possono trovare normalmente in commercio; intende invece soddisfare le esigenze di tutti coloro che sono alla ricerca di un sistema economico ed essenziale, ma estremamente efficiente.

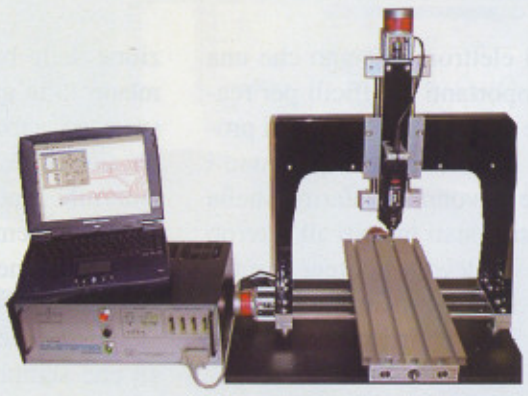
La *RM-MiniMill* realizza un sistema meccanico tridimensionale completo; può infatti essere utilizzata per diverse operazioni tipiche delle macchine utensili, per esempio per fresare, forare, incidere, dosare, tagliare, marcare a micropercussione, ecc.

Tra le caratteristiche principali indichiamo: la struttura molto robusta e completamente auto-portante; gli alberi in acciaio cromato rettificato, con scorrimento su cuscinetti autolubrificanti; la disponibilità di un piano di lavoro in alluminio anodizzato con scanalature a T per il fissaggio del pezzo; l'ingombro massimo di 600 x 600 x 700 mm; l'area di lavoro di 270 x 220 x 120 mm; la dimensione del piano di lavoro 490 x 180 mm con una Luce di 145 mm; il peso di 40 kg. Chiaramente gli scopi che a noi più interessano sono quelli rivolti alla realizzazione dei PCB dei circuiti stampati. Il file normalmente utilizzato per questo tipo di lavorazione è l'*HP-GL* (estensione *.PLT*).

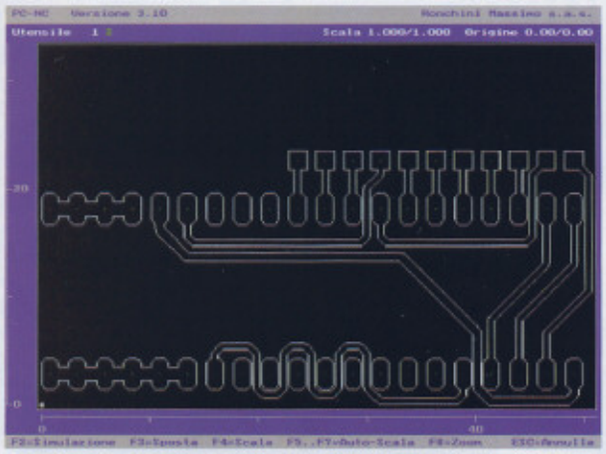
Per lavorare la basetta è necessario ottenere dal file "Top" o "Bottom" del file PCB il file *.PLT* con l'isolamento delle piste. Per circuito di isolamento si intende il percorso che l'utensile fresa deve compiere per generare la pista, il pad, ecc.

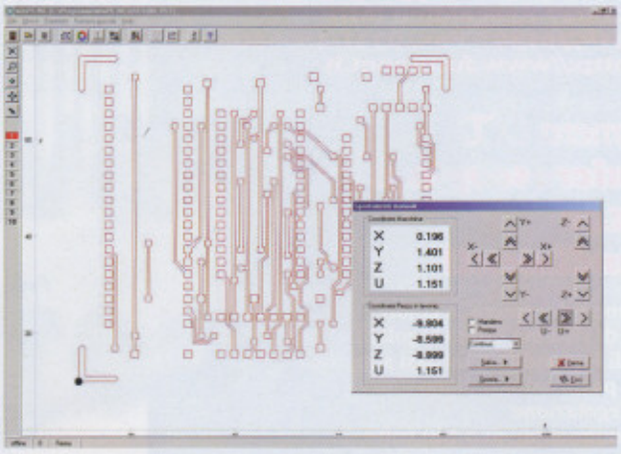
La macchina *RM-MiniMill*, oltre che come fresa, può essere utilizzata anche come trapano; per i nostri interessi questa possibilità risulta molto comoda per realizzare i fori in cui saldare i diversi componenti o mediante i quali fissare il circuito a dispositivi o contenitori esterni. Per questo, una volta che l'operazione di scontornatura è ultimata, il PCB potrà essere forato automaticamente semplicemente caricando il corrispondente file *Drill*.

Dalla descrizione generale della macchina che abbiamo appena >



*Il sistema completo risulta composto dalla macchina **RM-MiniMill** e dal sistema di controllo **2A Stepper Motor Controller**, in grado di gestire da 1 a 4 motori passo passo direttamente tramite porta parallela di un PC.*



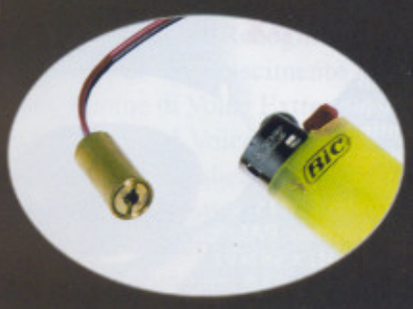


*Affinché la macchina "sappia" come muovere l'utensile lungo le tre dimensioni X, Y, e Z è necessario un software 3D che ne comandi i tre motori passo passo lungo i tre assi. Sono disponibili due versioni: il primo (**PCNC 3.1**) funzionante in ambiente DOS e il secondo (**Win-PCNC**) compatibile con qualsiasi sistema operativo della famiglia Windows. Entrambi sono in grado di importare files nel formato: **HP-GL, DIN/ISO 66025 (G-CODE 2D/3D)** e i due differenti formati di foratura **Syeb&Mayer** ed **Excellon**.*

visto risulta evidente che per costruire il PCB è necessario lavorare con un CAD elettronico in grado di esportare il file *.PLT* con l'isolamento delle piste.
Il nostro consiglio è di utilizzare il software *Target 3001!*; si tratta di un software estremamente potente, che oltre a tutte le normali funzioni rese disponibili da un tipico CAD elettronico, permette anche di esportare l'isolamento delle piste in formato *HP-GL* semplicemente impostando il diametro dell'utensile fresa che si andrà ad utilizzare. La parte che normalmente crea confusione in chi per la prima volta si avvicina a questo tipo di lavorazioni è in che modo generare il file *PLT* con l'isolamento delle piste e il file *Drill* di foratura; utilizzando il software *Target 3001!* questa operazione risulta davvero molto semplice in quanto potrete passare direttamente dalla progettazione del circuito all'esportazione del file con estensione *.PLT*.
Tornando al sistema da noi presentato, questo risulta composto da due sezioni: la prima costituita dalla *RM-MiniMill* che realizza il movimento dell'utensile lungo le tre dimensioni; la seconda composta da un sistema di controllo denominato *2A Stepper Motor Control* che, interfacciandosi da una parte alla *RM-MiniMill* e dall'altra al software CNC di comando macchina, è in grado di gestire il movimento nello spazio dell'utensile secondo le indicazioni provenienti dal computer.
In conclusione di articolo vi facciamo presente che la macchina viene venduta parzialmente montata in kit, con gli assi del movimento già assemblati.
Prima della messa in funzione della stessa è necessario prevedere tutte le opportune protezioni e sicurezze per renderla conforme in base alla direttiva macchine; la realizzazione e la messa a norma della macchina deve essere realizzata esclusivamente da personale competente.

puntatori laser

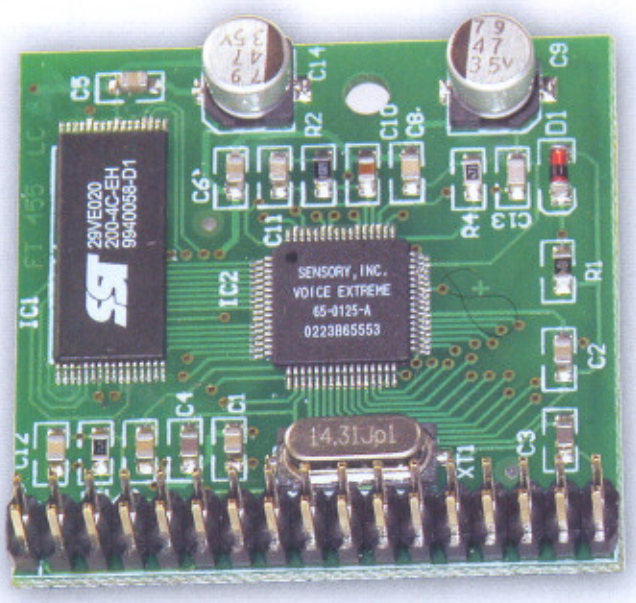
FUTURA ELETTRONICA
V.le Kennedi 96
Rescaldina (MI)



Moduli laser allo stato solido comprendenti un diodo a semiconduttore con emissione a 635 o 670 nanometri. I puntatori comprende un diodo laser, un collimatore con lenti in plastica e un alimentatore a corrente costante realizzato in SMD. Il diametro del modulo è di appena 10,5 millimetri con una lunghezza di 19,8 mm.
FR30 (670 nm) 28,00 euro
FR53 (635 nm) 65,00 euro
Schede tecniche dettagliate su: www.futuranet.it

Modulo Avanzato Voice Extreme

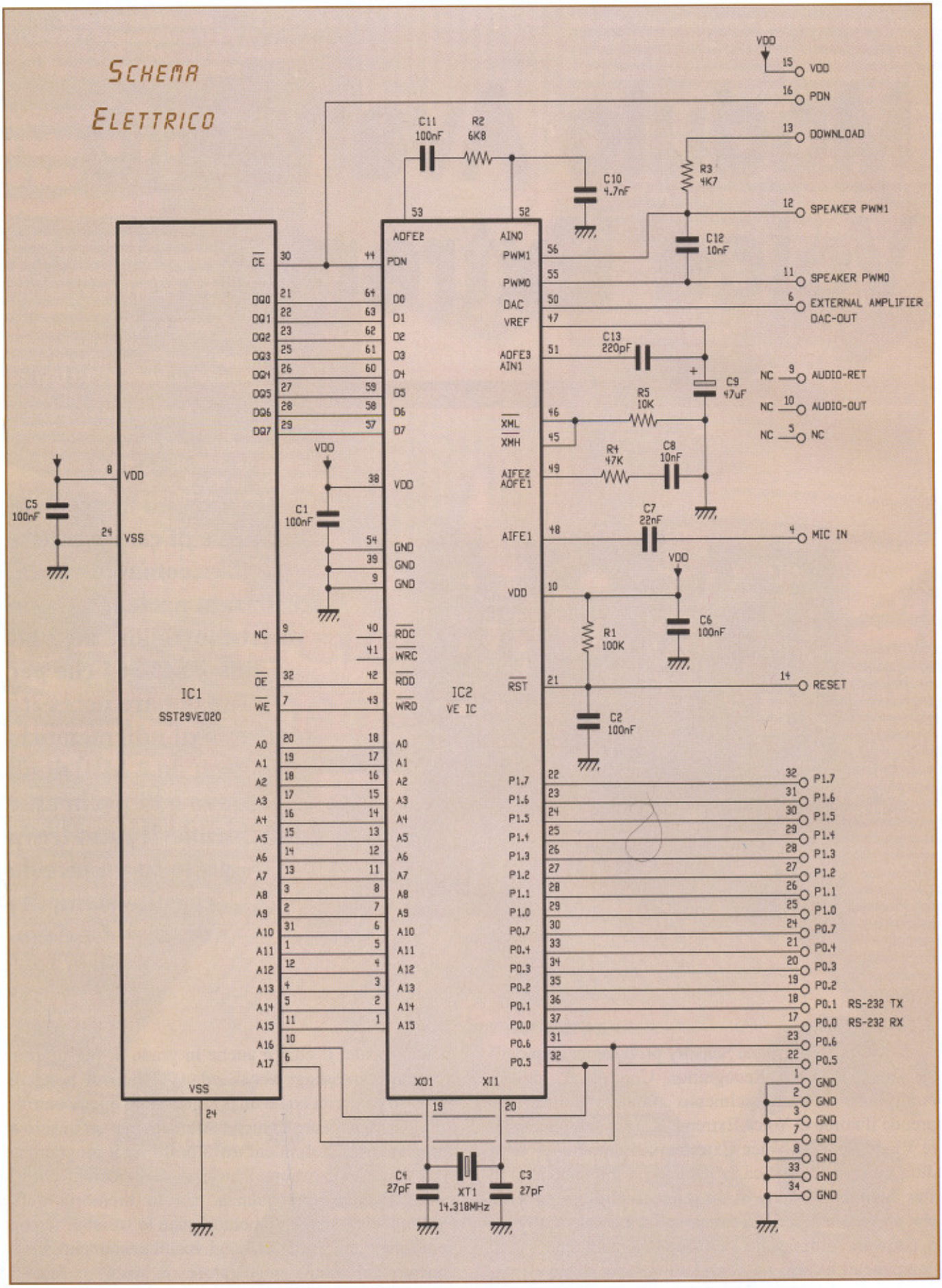
di *Carlo Vignati*



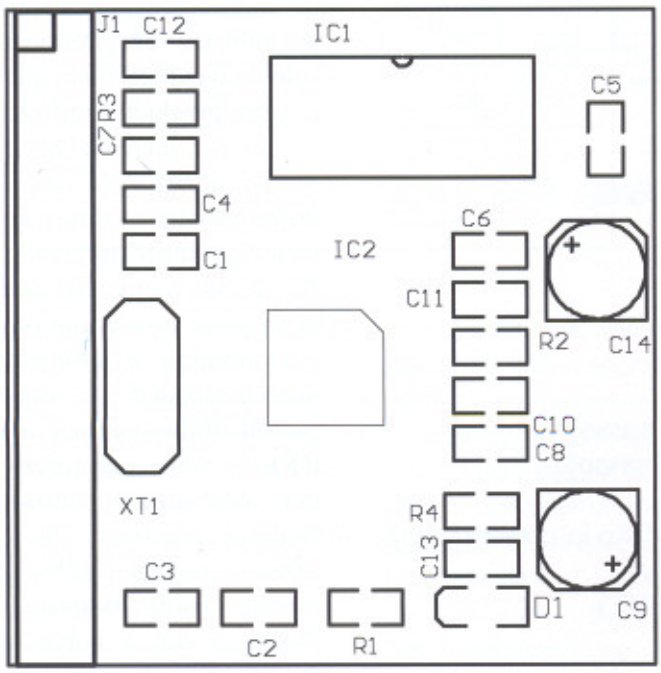
Un circuito in grado di parlare e di comprendere comandi vocali. Realizzato partendo da un microcontrollore ad 8 bit tipo RSC-364 che per funzionare necessita solamente di una memoria Flash da 2 MB, di un quarzo e di pochi altri componenti. Il tutto trova posto su un piccolo circuito stampato di circa 4 x 4 cm.

La ditta californiana Sensory produce un chip definito "Speech Recognition Controller", ovvero controllore di riconoscimento vocale. Questo chip prende il nome di Voice Extreme IC (più semplicemente VE-IC) poiché Voice Extreme è il nome con cui la ditta costruttrice indica la tecnologia implementata al suo interno. Si tratta di un microcontrollore ad 8 bit che prevede una serie di funzionalità a livello hardware e software ottimizzate per implementare con pochi componenti esterni un completo sistema di riconoscimento vocale. Il chip è anche in grado di registrare e riprodurre messaggi vocali ed è dotato di porte di ingresso ed uscita come un normalissimo microcontrollore. Per funzionare richiede esternamente un quarzo e una memoria Flash in cui trova posto sia il programma vero e proprio, ovvero l'insieme di istruzioni che il micro esegue in sequenza, sia la memoria dati. Nell'area dati vengono memorizzate le variabili (come un normale micro), i messaggi vocali campionati direttamente dal micro (registratore di suoni) o inseriti >

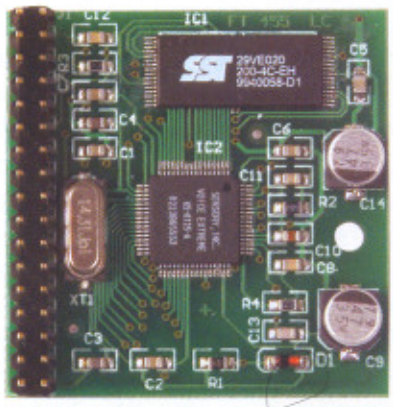
SCHEMA ELETTRICO



DISPOSIZIONE COMPONENTI



In alto, la serigrafia del modulo in scala doppia.



COMPONENTI:

U1: SST29VE020
2Mbit EEPROM
U2: VE-IC
SPEECH
RECOGNITION
CONTROLLER
D1: 1N4148

R1: 100 KOhm
R2: 6,8 KOhm
R3: 4,7 KOhm
R4: 47 KOhm
C1: 100 nF
C2: 100 nF
C3: 27 pF
C4: 27 pF
C5: 100 nF
C6: 100 nF
C7: 22 nF
C8: 10 nF
C9: 47 µF
C10: 4,7 nF
C11: 100 nF
C12: 10 nF
C13: 220 pF
C14: 47 µF
XTAL1: Quarzo 14.318MHz
J1: CONNETTORE STRIP
17+17 POLI
PASSO 2,54mm

Tutti i componenti sono in contenitore a montaggio superficiale.

CIR-810
Registratore
Digitale
531/988 minuti
Porta USB



CAPTURE VIEW
Binocolo 8x con macchina fotografica digitale
Porta USB



140-1196
Registratore Digitale 22/90 minuti
Microfono direzionale
Porta USB



CD-2
All bands
Cellular Killer
medio raggio
180mW

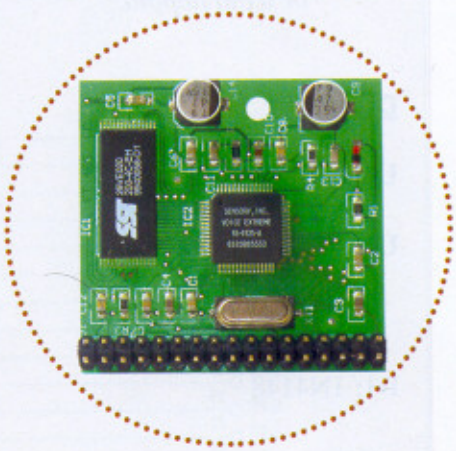


BIAS S.a.
www.bias.sm
Strada del Lavoro,33 47892 Gualdicciolo
REPUBBLICA DI S. MARINO
Tel. 0549.999408. Fax 0549.999431

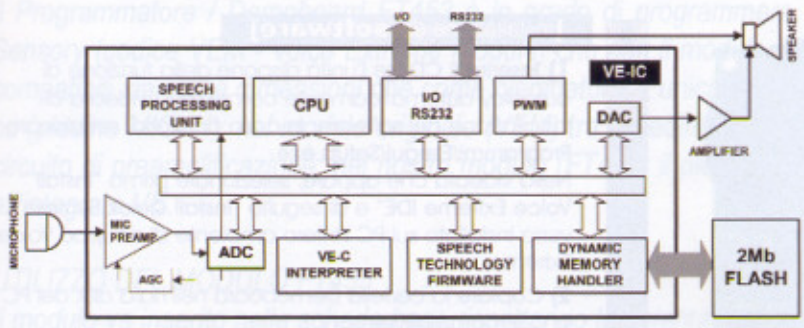
DESCRIZIONE DEL CONNETTORE J1

PIN	SIGLA	DESCRIZIONE
1	GND	GROUND
2	GND	GROUND
3	GND	GROUND
4	MIC-IN	MICROPHONE IN
5	NC	NOT CONNECTED
6	DAC-OUT	DAC OUTPUT
7	GND	GROUND
8	GND	GROUND
9	AUDIO-RET	- SPEAKER
10	AUDIO-OUT	+ SPEAKER
11	PWM0	PWM OUT 0
12	PWM1	PWM OUT 1
13	DOWNLOAD	DOWNLOAD (attivo basso)
14	RESET	HWD RESET (attivo basso)
15	VDD	VDD (2.85V ÷ 3.3V)
16	PDN	Power Down Output (alto in power down)
17	P0-0	RS232 DATA IN (RCV)
18	P0-1	RS232 DATA OUT (XMT)
19	P0-2	I/O
20	P0-3	I/O
21	P0-4	I/O
22	P0-5	USE ONLY FOR KEYPAD
23	P0-6	USE ONLY FOR KEYPAD
24	P0-7	I/O
25	P1-0	I/O
26	P1-1	I/O
27	P1-2	I/O
28	P1-3	I/O
29	P1-4	I/O
30	P1-5	I/O
31	P1-6	I/O
32	P1-7	Serial port enable output (0=off, 1=on)
33	GND	GROUND
34	GND	GROUND

durante la programmazione (frasi che il micro sarà in grado di riprodurre), ed i segnali vocali necessari per compiere il riconoscimento del parlato.
All'interno di questo minuscolo integrato troviamo anche un interprete C, denominato Voice Extreme C Interpreter, in grado di comprendere sia i normali comandi C che una serie di potenti istruzioni proprietarie della tecnologia VE e studiate per il riconoscimento vocale.
I programmi scritti per questo chip devono essere realizzati in linguaggio C; una volta scritte le varie istruzioni, e caricate nella memoria Flash, queste sono tradotte dall'interprete in linguaggio macchina ed eseguite dalla CPU. Per scrivere un programma in C adatto al VE-IC e per trasferirlo nella Flash la Casa madre mette a disposizione un software per PC IBM compatibili denominato Voice Extreme IDE.
L'integrato VE-IC è disponibile in contenitore TQFP a 64 pin, è quindi molto piccolo e può essere assemblato sul circuito stampato solo da una macchina: questa particolarità preclude l'utilizzo del VE-IC ad un mercato hobbystico o semi-professionale che desidera utilizzarlo senza dover necessariamente produrre migliaia di pezzi.
Per questo motivo abbiamo deciso di proporre un circuito (disponibile già montato e collaudato) che implementi su di una sola basetta di piccole dimensioni (circa 4 x 4 cm) il VE-IC nella configurazione minima necessaria al funzionamento.
Occorre precisare che la stessa Sensory produce una basetta, denominata Voice Extreme Module (VEM), con a bordo il VE-IC. Questo modulo implementa il VE-IC in contenitore DIE senza interprete interno, una memoria ROM con il VE-C Interpreter, una memoria Flash e un circuito di preamplificazione microfonica. La scelta di prevedere una memoria esterna per l'interprete rende il modulo VEM piuttosto complesso; la presenza del circuito di preamplificazione microfonica prevede di dover agire su alcuni componenti del modulo VEM qualora si desideri variare il guadagno del circuito. L'insieme di questi motivi: semplificare il circui- >

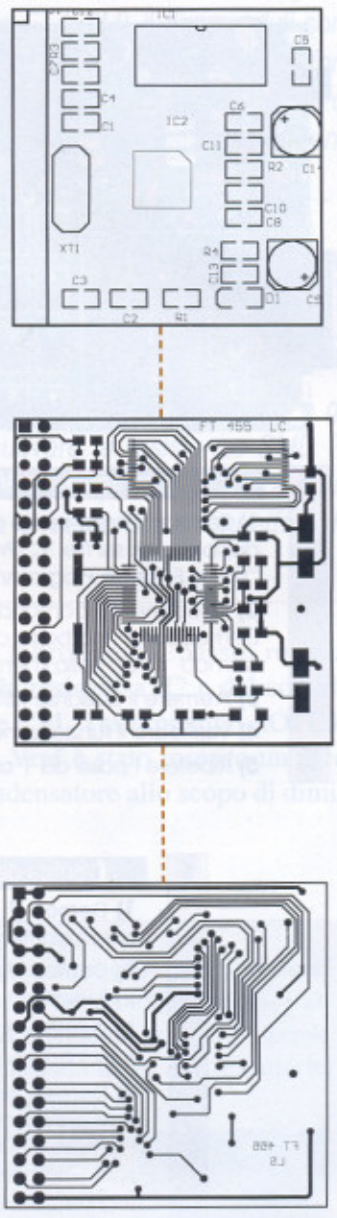


SCHEMA A BLOCCHI DEL VE-IC



Sopra, lo schema a blocchi dell'integrato Voice Extreme (più semplicemente detto VE-IC) prodotto da Sensory Inc. Si tratta in pratica di un microcontrollore a 8 bit che integra tutta una serie di funzionalità a livello hardware e software ottimizzate per implementare, con pochi componenti esterni, un completo sistema di riconoscimento vocale. Come si può notare si tratta di un microcontrollore molto particolare; è possibile collegare direttamente un microfono al chip, in quanto è dotato di una rete analogica che effettua la preamplificazione necessaria, utilizzando anche un AGC, ovvero un controllo automatico del guadagno per limitare correttamente la dinamica del segnale. Vi è poi il blocco di conversione analogico/digitale (ADC) per poter campionare il segnale analogico del microfono e renderlo disponibile per l'elaborazione. In uscita, il micro mette a disposizione un segnale PWM, in cui cioè viene modulato il duty cycle (ovvero il rapporto fra livello logico alto e basso) di un'onda quadra, oppure un'uscita derivante da un convertitore digitale/analogico (DAC), che in pratica restituisce un segnale analogico pronto per essere amplificato e filtrato per poter pilotare un altoparlante. Esternamente è necessario collegare una memoria FLASH (ovvero cancellabile e riscrivibile elettricamente); questa memoria serve per contenere sia il programma che dovrà eseguire il micro sia i dati relativi ai segnali vocali campionati e deve avere una capacità di 2 Mbit. Proseguendo nell'analisi dei blocchi funzionali, troviamo la CPU vera e propria, in pratica il cuore digitale del chip, l'interfaccia per le linee di I/O e per la gestione della comunicazione via RS232 ed un blocco di elaborazione del segnale vocale ("Speech Prosessing Unit"). Troviamo poi il VE-C Interpreter che è l'interprete per il linguaggio C. Questo significa che i programmi scritti per questo chip devono essere realizzati in linguaggio C. Una volta scritte le varie istruzioni, e caricate nella memoria flash, queste sono tradotte dall'interprete nelle istruzioni in linguaggio macchina che la CPU eseguirà. Troviamo, infine, lo "Speech Technology Firmware", ovvero quella parte di software che permette di realizzare tutte le funzioni strettamente legate alla tecnologia vocale, come ad esempio il riconoscimento vocale.

LA BASETTA



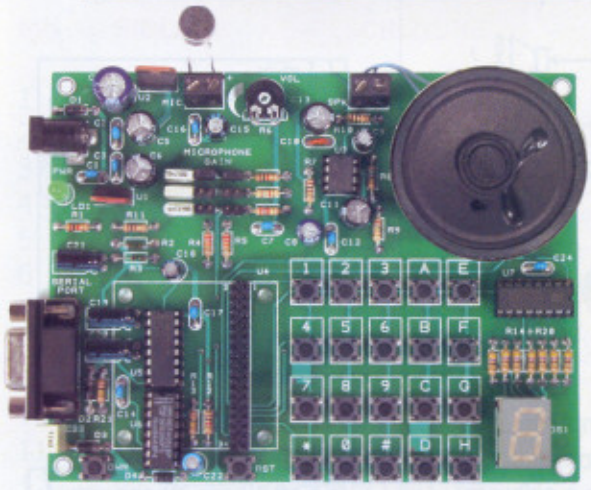
Dall'alto verso il basso, serigrafia, lato componenti, lato saldature della basetta in dimensioni reali.

to, ridurre il costo, estrapolare il circuito di preamplificazione oltre a quelli esposti in precedenza ci hanno spinto a realizzare un nostro modulo caratterizzato comunque da uguali dimensioni e da un uguale connettore di interfaccia.

Schema elettrico

Il circuito prevede due soli integrati: il VE-IC in contenitore TQFP a 64 pin e la memoria Flash da 2 Mbit in contenitore TSOP a 32 pin. Il micro si interfaccia con la memoria tramite un bus dati a 8 bit (piedini da D0 a D7) e un bus indirizzi a 16 bit (piedini da A0 a A15). L'uscita di Power Down (PDN) del micro controlla l'abilitazione della memoria (Chip Enable) ed è disponibile sul pin 16 del connettore di >

PER PROGRAMMARE L'INTEGRATO VOICE EXTREME



[installazione del software]

1) Inserire il CD, se l'unità dispone della funzione di autoplay automaticamente compare la videata di installazione del software; in caso contrario, selezionare **Programmi/Esegui/Setup.exe**.
Nella videata che appare, selezionare prima **"Install Voice Extreme IDE"** e di seguito **"Install Quick Syntesis"**: verrà installato sul PC l'intero ambiente di sviluppo Voice Extreme.

2) Copiare la cartella Demoboard nell'hard disk del PC in **C:\Programmi\Sensory**.

3) Collegare la seriale del PC alla linea seriale della Demoboard e **alimentare la scheda** con una tensione continua e stabilizzata di **12 Volt**.

[programmazione del VE-IC]

1) Selezionare ed **aprire** il primo programma dimostrativo:
doppio click sul file **"C:\Programmi\Sensory\Demoboard\Demob_1\Demob_1.veb"**; appare la videata del software **"Voice Extreme Programmer"**.

2) Selezionare il comando **"Start Download"** e **premere**, entro 5 secondi, il **pulsante DWN** disponibile sulla Demoboard: una barra di scorrimento indica il progredire della programmazione: viene trasferito il file binario Demob_1.veb nella memoria Flash disponibile sulla Demoboard.

3) Premere il pulsante **RST** della Demoboard.

4) Verificare il funzionamento del programma dimostrativo **Demob_1.veb**.

5) Ripetere i passi da 1 a 4 per programmare e **verificare** il funzionamento degli **altri listati** dimostrativi.

[verifica del funzionamento dei listati dimostrativi]

1) **Demob_1.veb** - Premere i pulsanti da 1 a 9 e verificare che il display a 7 segmenti visualizzi il numero del pulsante premuto.

2) **Demob_2.veb** - Premere i pulsanti da 1 a 9, i pulsanti *, #, A, B, C, D e verificare che venga generato il relativo tono DTMF.

3) **Demob_3.veb** - Premere i pulsanti disponibili sulla Demoboard e verificare che la demoboard pronunci il numero o la lettera del pulsante premuto; agire sul trimmer R6 per regolare il volume.

4) **Demob_4.veb** - Premere il pulsante 1, attenersi alle frasi pronunciate dalla Demoboard per inserire due comandi vocali in tecnologia Speaker Dependent.
Premere il pulsante 2, pronunciare il primo comando vocale inserito: il display 7 segmenti deve visualizzare il numero 1. Premere il pulsante 2, pronunciare il secondo comando vocale inserito: il display 7 segmenti deve visualizzare il numero 2.

5) **Demob_5.veb** -Il programma consente di registrare e riprodurre dei messaggi vocali. Allo scopo vengono utilizzati i seguenti pulsanti: 0 = Avvia la registrazione di un messaggio; 1 = Avvia la riproduzione in sequenza dei messaggi preregistrati; 2 = Cancella tutti i messaggi disponibili in memoria; 3 = Ferma la registrazione o la riproduzione dei messaggi.

6) **Demob_6.veb** - Il programma riproduce 3 brani musicali. Allo scopo vengono utilizzati i seguenti pulsanti: 1 = Avvia riproduzione brano 1; 2 = Avvia riproduzione brano 2; 3 = Avvia riproduzione brano 3; 0 = Ferma riproduzione brano.

7) **Demob_7.veb** - Premere il pulsante 1 per avviare la procedura di riconoscimento Speaker Independent; pronunciare in inglese i numeri da 0 a 9; verificare che il display a 7 segmenti visualizzi la cifra del numero pronunciato.

8) **Demob_8.veb** - Funziona come il dimostrativo numero 7, ma il numero riconosciuto viene inviato in uscita sul canale seriale della Demoboard.

9) **Demob_9.veb** - Premere RST, premere il pulsante 1, rilasciare RST e mantenere premuto il pulsante 1 fino ad udire un beep. Attenersi alle frasi pronunciate dalla Demoboard per inserire due comandi vocali in tecnologia Continuous Listening. Pronunciare il primo comando e verificare che la Demoboard emetta un beep; pronunciare anche il secondo comando e verificare che la Demoboard emetta 3 beep.

MODIFICHE AL MODULO VEM

Il Programmatore / Demoboard FT453 è in grado di programmare e di lavorare sia con il modulo originale Sensory (codice VEM - Voice Extreme Module) che con il modulo FT455 (codice FT455M). I due moduli sono compatibili sia come dimensioni che come piedinatura. L'unica differenza è nella sezione di ingresso microfonico (piedino MIC-IN): nel modulo originale (VEM) tra il piedino MIC-IN del modulo e l'integrato VE è inserito un circuito di preamplificazione; nel nostro modulo (FT455) il piedino MIC-IN del modulo è collegato direttamente all'integrato VE.

UTILIZZO DEL MODULO FT455

Il modulo va inserito nella scheda base rispettando la polarità; nessuna modifica è necessaria.

UTILIZZO DEL MODULO VEM

E necessario escludere il circuito di amplificazione microfonica previsto sul modulo poiché tale circuito è già implementato sulla scheda base FT453. Per fare ciò occorre dissaldare e rimuovere la resistenza SMD R7 implementata sul modulo VEM. Effettuare la rimozione di R7 utilizzando un saldatore di piccola potenza e una pinzetta. La resistenza R7 è posizionata sul retro del modulo (quello in cui si trova la memoria e il connettore J1) e sul lato esterno in basso.

uscita J1. L'uscita Data Read Strobe (RDD) del micro controlla la linea di abilitazione delle uscite (OE) della memoria; infine l'uscita WRD (Data Write Strobe) pilota l'ingresso di Write Enable della memoria ed in pratica dice alla memoria se il processo in corso è di scrittura o di lettura. Tra i piedini AOFE2 e AIN0 del micro è previsto un filtro passa basso realizzato dalla resistenza R2 e dal condensatore C11. Tra il piedino AOFE3 e il pin Vref è stato inserito un piccolo condensatore allo scopo di diminui- >

Per il MATERIALE



Il circuito presentato in queste pagine è disponibile già montato e collaudato (cod. FT455M) al prezzo di 39,00 Euro. Il modulo implementa l'integrato Voice Extreme, la memoria Flash e tutti i componenti indicati nel piano di cablaggio; il connettore di uscita è di tipo pin-strip a passo 2,54 mm.

Per imparare le tecniche di programmazione e di utilizzo dell'integrato Voice Extreme è disponibile un Corso di Programmazione in lingua italiana (cod. CPR-VE; 15,00 Euro) e una Demoboard (cod. FT453K; 45,00 Euro).

La Demoboard/Programmatore FT453K è disponibile in scatola di montaggio e comprende tutti i componenti, la basetta forata e serigrafata, l'altoparlante, il microfono, un cavo seriale per il collegamento al PC, le minuterie, un CD con l'ambiente di sviluppo IDE del VE-IC attraverso il quale è possibile scrivere il programma in C e scaricarlo attraverso la seriale nella memoria flash, il programma Quick Synthesis, la documentazione originale in lingua inglese e i programmi dimostrativi in italiano a livello sorgente accuratamente commentati; il kit non comprende il modulo FT455M.



Tutti i prezzi indicati sono comprensivi di IVA.
Il materiale va richiesto a: Futura Elettronica, V.le Kennedy 96, 20027 Rescaldina (MI).
Tel: 0331/576139 ~ Fax: 0331/466686 ~ Sito: *www.futuranet.it*

re il più possibile il rumore di fondo del circuito di preamplificazione; l'ingresso di questo stadio risulta collegato, attraverso un condensatore di disaccoppiamento (C7), al piedino 4 (MIC-IN) di J1. Un circuito di reset composto da R1 e da C2 è stato inserito per permettere al micro di "avviarsi" correttamente quando riceve alimentazione; il pin RST del micro è inoltre collegato al pin 14 (Reset) di J1. Tutte le linee di ingresso uscita del micro sono state portate al connettore J1. A tale proposito, occorre precisare che le porte P0.6 e P0.5 vengono utilizzate come bus degli indirizzi verso la memoria e possono essere usate esternamente solo per il controllo di una tastiera a matrice tramite le relative funzioni disponibili nel VE-C. Le porte P0.0, P0.1 e P1.7 oltre ad essere delle normali linee di ingresso / uscita sono anche il canale seriale del micro: viene utilizzato ad esempio per comunicare con il PC in fase di programmazione della Flash. Infine, per funzionare il micro richiede un quarzo esterno da 14.32 MHz (XT1) e di due piccoli condensatori C3 e C4. Sono previste due distinte uscite del segnale audio: una uscita è realizzata in PWM ed è disponibile sui piedini PWM0 e PWM1, una seconda uscita analogica è disponibile sul piedino DAC del micro. Tutte e tre le linee vengono portate sul connettore J1. L'uscita PWM0 / PWM1 può pilotare direttamente un altoparlante con impedenza di 32 Ohm. Si noti che il pin PWM1 risulta anche connesso, tramite la resistenza R3, al piedino Download di J1. Questo perché, a seguito di un Reset, il micro trasforma per un breve istante l'uscita PWM1 in ingresso: un cambiamento di stato su questo pin significa che intendiamo attivare la procedura di programmazione della Flash (download dei dati nella Flash). Abbiamo detto che il modulo viene fornito già montato e collaudato, ciò significa che non è necessario spiegare la realizzazione pratica; in ogni caso, in queste pagine trovate l'elenco componenti, il piano di cablaggio e le tracce rame in scala reale del modulo. Per poter realizzare un programma adatto al nostro modulo e più in generale al VE-IC è ora necessario installare su di un PC IBM compatibile il software VE-IDE e imparare le funzioni che l'interprete VE-C mette a disposizione.

Possiamo affermare che scrivere un programma in C per il VE-IC è più semplice del previsto. Per mettere in grado chiunque di lavorare con questo innovativo integrato abbiamo realizzato anche un Corso di programmazione in lingua italiana e una Demoboard / Programmatore presentata nel fascicolo numero 77 di Elettronica In.

Buona programmazione...

ORDINARE A: *IDEA ELETTRONICA*
Via XXV Aprile n°76 – 21044 Cavaria con Premezzo - VA-Tel./Fax 0331215081 – Email: idele@tiscalinet.it
WWW.IDEAELETTRONICA.IT



FARFALLE e LIBELLULE
Cinetiche Formate da SMAs

Muove le ali come una Farfalla / Libellula! Animata da un piccolo filo di Flexinol Muscle Wire, la Farfalla è disponibile in tre modelli (Blu Morpho del centro America, Monarch del nord America e Old World dell' America e Eurasia), mentre la Libellula è disponibile in due modelli (Blue Dragonlet o Halloween Skimmer) .

FARFALLA O LIBELLULA
(complete di alimentatore da rete) € 36,20 cad. iva comp.

SHAPE MEMORY ALLOYs (leghe metalliche con memoria di forma)

Queste particolari leghe metalliche quando vengono attraversate da corrente o semplicemente riscaldate, subiscono cambiamenti di forma e durezza. Tra i vari tipi di SMAs, abbiamo scelto quella sotto forma di Filo detto Flexinol Muscle Wire: composto da Nickel e Titanio riduce la sua lunghezza quando viene riscaldato o attraversato da corrente, ed è in grado di sollevare un corpo pesante migliaia di volte rispetto al suo peso, in modo completamente silenzioso.



Flexinol 037µm € 18,10 – Flexinol 050µm € 18,10 – Flexinol 100µm € 18,91
Flexinol 150µm € 20,00 – Flexinol 250µm € 21,00 – Flexinol 300µm € 23,04
Flexinol 375µm € 24,20
PER MAGGIORI INFORMAZIONI VISITA IL NOSTRO SITO.

REGISTRATORE DIGITALE DA 6-12-24 ORE

Sistema di registrazione: flash memory - Registra un totale di 396 messaggi in 4 files separati (99 essaggi in ogni file)- Tasti: Play, Rec, Stop, Mode, Cancella - Funzione di ricerca rapida - Regolazione Volume - Funzione di blocco: impedisce gli azionamenti accidentali quando non è in uso - VOR (sistema di registrazione automatica): impedisce inutili registrazioni quando non c'è suono o voce - altoparlante 23mm di diametro - Back-up di messaggi al PC: immagazzina i messaggi registrati e la posta a voce - indicatore di batteria scarica - Dimensioni 104 x 26,8 x 17 mm - collegamento al telefono: registra conversazioni telefoniche.



La confezione contiene i seguenti accessori:
Cavo in/out, adattatore telefonico, cuffia, microfono esterno, batterie, software.

RD6 (6 ORE) € 196,26 - RD12 (12 ORE) € 216,92 - RD24 (24 ORE) € 237,60

DIRIGIBILE RADIOCOMANDATO GRANDE



Ruota di 360 Gradi e Vola a 15 metri L'altezza.
Il pallone è in mylar e misura 133 x 94 cm sigonfia con Elio, 3 microjet per la propulsione (il terzo microjet serve per farlo salire o scendere), il radiocomando è alimentato da una Batteria da 9 Volt (non inclusa) mentre il ricevitore da una pila da 3 Volt.
La confezione comprende: Il pallone, la Navetta con i tre micromotori, il Ricevitore e il Radiocomando(Portata: circa 400m).

Codice PIM33 € 144,61 iva compresa

CYCLONE ROBOT KIT

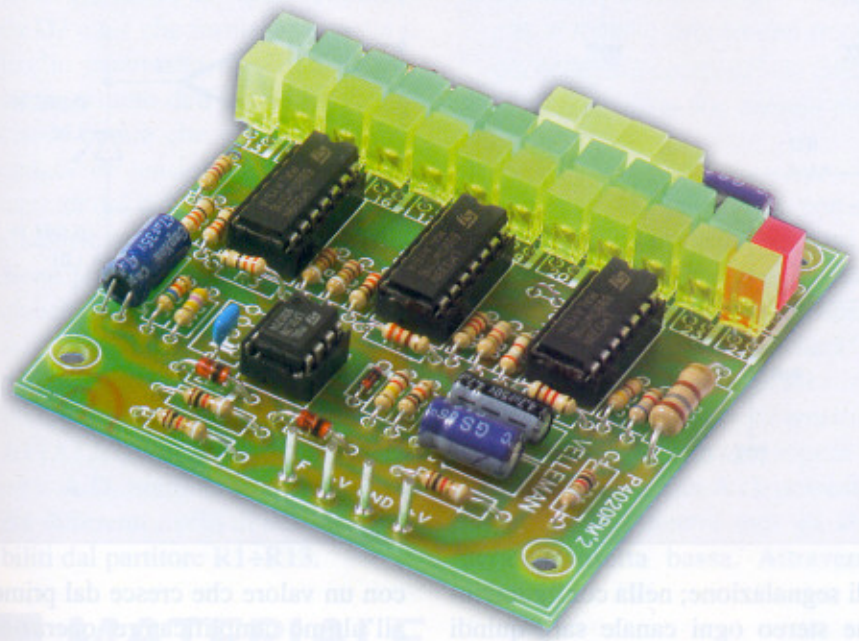
Cyclone robot kit, è una palla radiocomandata, risponde immediatamente al comando di avanti-indietro-destra-sinistra-accelera-decelera.
Caratteristiche: 2 ruote pilotate da 2 Motori DC
Controllo: Radiocomando
Alimentazione: n° 2 Pile "AA" da 1,5V per la Palla , n°2 Pile da 1,5V "AA" per il Radiocomando (non comprese) - Dimensione Palla: 14 cm circa.



Codice PIM38 € 93,00 iva compresa

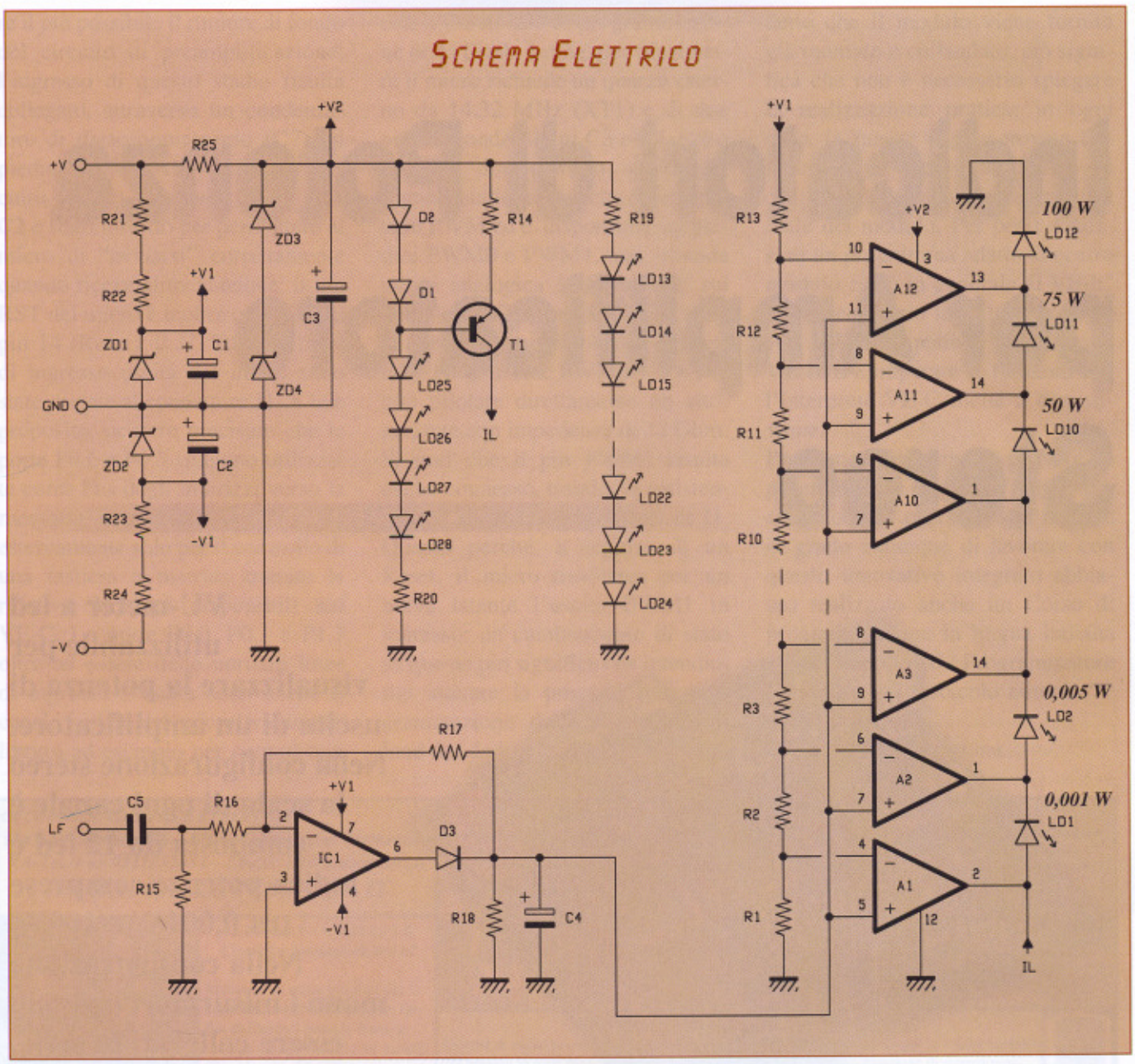
Indicatori di Potenza per Amplificatore Stereo

di Davide Ferrario



VU-meter a led utilizzabile per visualizzare la potenza di uscita di un amplificatore. Nella configurazione stereo la scala di ogni canale è composta da 12 led e indica potenze comprese tra 0,001W e 100W. Nella configurazione mono i misuratori possono essere collegati in serie, utilizzando 24 led con una scala compresa tra 0,001W e 400W.

Dopo avervi presentato nel fascicolo di giugno l'amplificatore Mosfet mono o stereo caratterizzato da una potenza di 2x300W, in questo articolo vi descriviamo un suo possibile e utile accessorio. Prenderemo infatti in considerazione un kit composto da due moduli Led Power Meter che, tramite una scala a led, consentono di visualizzare la potenza istantanea del segnale di uscita dell'amplificatore. Probabilmente sapete già quali sono le funzionalità di un circuito del genere: si tratta infatti di un dispositivo che realizza i VU-METER che equipaggiano gli stadi finali dei moderni hi-fi. L'utilità di queste scale, oltre ad una pura funzione estetica, è quella di permettere di riconoscere a vista se la potenza richiesta all'amplificatore si avvicina al limite massimo; in questo modo è possibile regolare il volume del preamplificatore prima di raggiungere tale soglia riuscendo quindi ad evitare la relativa distorsione del segnale audio. All'interno del kit sono presenti due circuiti; infatti nella configurazione stereo ogni modulo visualizza la potenza di un singolo canale (destro e sinistro). Invece nel caso di configurazione mono i due circuiti devono essere commutati in >



serie in modo da realizzare un "doppio indicatore". Per quanto riguarda la scala delle potenze visualizzate, nella configurazione stereo ogni misuratore è in grado di indicare se la potenza risulta compresa nel range 0,001W÷100W nel caso di impedenza degli altoparlanti pari a 8 ohm; se invece l'impedenza è pari a 4 ohm il range visualizzato è di 0,002W÷200W. Infine, nel caso di configurazione "a ponte" e con impedenza di 8 ohm, i misuratori forniscono un'indicazione da 0,001W a 400W.
Su ogni scheda sono presenti 12 led di segnalazione; nella configurazione stereo ogni canale sarà quindi caratterizzato da 12 livelli. Nella configurazione a ponte, essendo i due circuiti collegati in serie, la scala risultante sarà invece composta da 24 livelli.

Schema elettrico

Passiamo ora ad analizzare lo schema elettrico del circuito. In sostanza questo risulta centrato intorno ai 12 comparatori di tensione A1÷A12 (chip IC1÷IC3) che confrontano il livello della tensione di ingresso con un valore che cresce dal primo all'ultimo amplificatore operazionale. A seconda di quanti comparatori presentano uscita positiva, verrà acceso un numero differente di led (contrassegnati con LD1÷LD12).
Il circuito è composto da una prima sezione di alimentazione che preleva le tensioni +/- 40V e massa direttamente dall'amplificatore. Tramite i due zener ZD1 e ZD2 vengono generati i livelli +/- 16V (indicati con +/- V1) utilizzati per alimentare l'integrato IC1. Partendo invece da +40V e tramite ZD3 e ZD4 vengo- >

no generati i +32V (+V2) che alimentano gli integrati che realizzano i 12 amplificatori operazionali.
Il transistor T1 è utilizzato come generatore di corrente; infatti grazie a D1, D2 e la caduta di tensione di circa 0,7V presente tra la base e l'emettitore dello stesso BJT, ai capi della resistenza R14 è localizzato un voltaggio di circa 0,7V; per questo attraverso la stessa scorre una corrente IL di circa 12,5 mA che, come vedremo più dettagliatamente in seguito, verrà utilizzata per accendere la serie di led LD1÷LD12.
Il segnale audio di ingresso è fornito al morsetto LF; attraverso C5 e R15 viene realizzato un primo filtraggio. Successivamente è presente un rilevatore di picco realizzato da D3 e C4 che fornisce in uscita il livello istantaneo massimo di tensione assunto dall'ingresso. Vi facciamo notare che, per eliminare la caduta di tensione di circa 0,7V causata dal diodo e tipica di questo stadio, è stata introdotta la retroazione R17 e l'amplificatore operazionale IC1.
A questo punto il segnale viene portato in ingresso al morsetto non invertente dei comparatori A1÷A12; gli ingressi invertenti di ogni A.O. sono invece collegati a dei differenti livelli di tensione, stabiliti dal partitore R1÷R13.

APPLICAZIONE STEREO o MONO

Gli indicatori di potenza a LED sono stati sviluppati per visualizzare la potenza di uscita dell'amplificatore Mosfet K4020 presentato nel fascicolo scorso (che, vi ricordiamo, può assumere due configurazioni: mono o stereo).
Per ognuna delle due configurazioni dell'amplificatore sono necessari due indicatori (realizzati da due schede); nel caso di amplificatore stereo le due basette (canale destro e sinistro) sono identiche.
Invece nel caso di amplificatore mono i misuratori (che devono essere commutati in serie) e le due basette differiscono per il valore assunto da alcuni componenti.
In particolare, come è possibile notare nel piano di montaggio, sono presenti alcuni componenti comuni a entrambe le applicazioni; altri elementi dipendono invece dal tipo di amplificatore utilizzato (stereo o mono); nel caso mono esistono anche differenze tra la prima e la seconda scheda.
Per quanto riguarda invece il collegamento verso il finale K4020, in entrambe le applicazioni i misuratori di destra e di sinistra sono collegati al relativo circuito dell'amplificatore. Utilizzando un filo di cablaggio collegare i punti LF, -V, GND e +V di ogni indicatore ai corrispondenti punti sulle due schede dell'amplificatore.

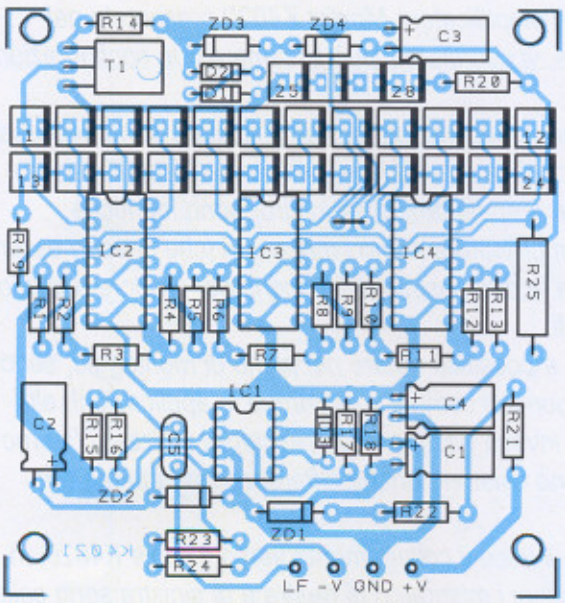
Supponiamo che in ingresso non sia applicato nessun segnale; tutti i comparatori presenteranno uscita bassa, tutti i led saranno quindi spenti e la corrente IL si "scaricherà" a massa attraverso l'uscita di A1.
Quando invece il picco di tensione del segnale di ingresso supera il livello VR1, il primo A.O. presenterà uscita alta, mentre tutti gli altri avranno uscita bassa. Attraverso LD1 può passare la corrente IL (il led sarà quindi acceso); LD2 sarà invece spento e IL si scaricherà attraverso l'uscita di A2.
Utilizzando la stessa logica è facile comprendere che quando il segnale di ingresso supera il livello VR1+VR2, i due led LD1 e LD2 saranno accesi e la corrente IL si scaricherà attraverso A3 e così via per tutti gli altri valori di tensione, >

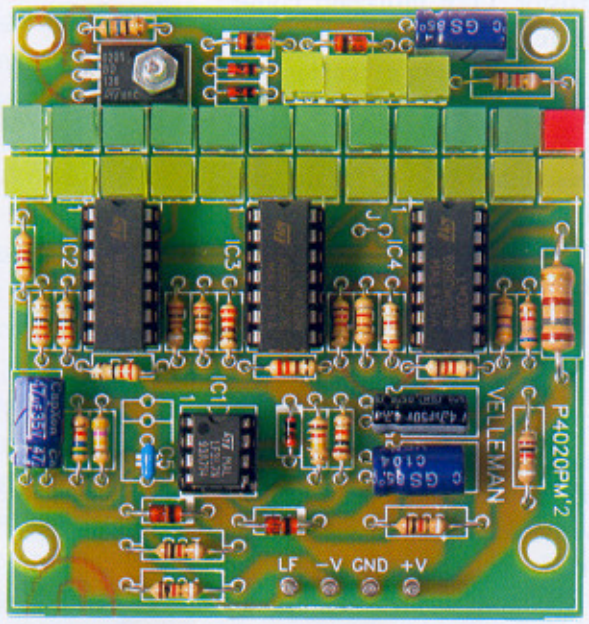
Per il MATERIALE

Il progetto descritto in queste pagine è un prodotto Velleman distribuito in scatola di montaggio in Italia dall'azienda Futura Elettronica. Il kit (cod. K4021, Euro 48,50) comprende il materiale per realizzare due circuiti VU-Meter: tutti i componenti, le due basette forate e serigrafate, tutti gli integrati, le minuterie e le istruzioni per il montaggio, disponibili anche in lingua italiana. Il kit è stato pensato come accessorio per la scatola di montaggio dell'Amplificatore Mosfet mono o stereo da 600W (cod. K4020, Euro 510,00) presentato nello scorso fascicolo. Tutti i prezzi indicati sono da intendersi IVA inclusa.

Il materiale va richiesto a: Futura Elettronica, V.le Kennedy 96, 20027 Rescaldina (MI).
Tel: 0331576139 ~ Fax: 0331466686 ~ Sito: *www.futuranet.it*

PIANO DI MONTAGGIO





ELEMENTI COMUNI:

R14: 56 Ohm
R15: 56 Ohm
R16: 470 KOhm
R17: 120 KOhm
R18: 10 KOhm
R19: 270 Ohm
R20: 1,5 Kohm
R21÷R24: 1 KOhm
R25: 180 Ohm 1W

C1: 47 µF 100 VL elettr.
C2: 47 µF 100 VL elettr.
C3: 47 µF 100 VL elettr.
C4: 4,7 µF 100 VL elettr.
C5: 470 nF multistrato

D1, D2, D3: 1N4148

DZ1: Zener 16 V 1,3W
DZ2: Zener 16 V 1,3W
DZ3: Zener 16 V 1,3W
DZ4: Zener 16 V 1,3W

T1: BD136

IC1: LF357
IC2: LM339
IC3: LM339
IC4: LM339

Varie:
- zoccolo 7+7 pin (3 pz.);
- zoccolo 4+4 pin;
- vite 3MA 8 mm;
- dado 3MA;
- circuito stampato K4021.

CONFIG. STEREO:

R1÷R3: 33 Ohm
R4, R5: 100 Ohm
R6, R7: 330 Ohm
R8: 1,2 KOhm
R9: 820 Ohm
R10: 3,9 KOhm
R11: 1,2 KOhm
R12: 470 Ohm
R13: 6,8 KOhm

LD1÷LD11: led quad. verde
LD12: led quad. rosso
LD13÷LD28: led quad. giallo

CONFIG. MONO:

Scheda 1

R1: 15 Ohm
R2: 6,8 Ohm
R3: 12 Ohm
R4: 12 Ohm
R5: 22 Ohm
R6, R7: 39 Ohm
R8: 68 Ohm
R9: 120 Ohm
R10: 120 Ohm
R11: 220 Ohm
R12: 390 Ohm
R13: 15 KOhm

LD1÷LD12: led quad. verde
LD13÷LD28: led quad. giallo

Scheda 2

R1: 1,5 KOhm
R2: 330 Ohm
R3: 270 Ohm
R4: 820 Ohm
R5: 680 Ohm
R6: 560 Ohm
R7: 470 Ohm
R8: 1 KOhm
R9: 1 KOhm
R10: 820 Ohm
R11: 680 Ohm
R12: 1,2 KOhm
R13: 5,6 KOhm

LD1÷LD10: led quad. verde
LD11, LD12: led quad. rosso
LD13÷LD28: led quad. giallo

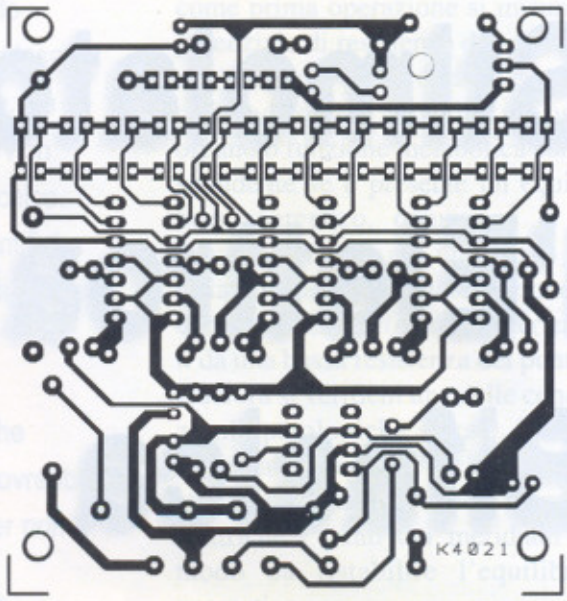
fino a quando il livello del morsetto LF non risulta massimo, condizione in cui tutti i led saranno accesi e la IL si scaricherà a massa. Prima di concludere l'analisi dello schema facciamo due precisazioni: la prima riguarda la corrente IL. Dallo schema elettrico si vede infatti che, nel caso che il segnale in ingresso sia massimo, questa passa attraverso tutti i 12 led; in questo modo gli stessi presenteranno uguale lumino- >

sità, indipendentemente dall'ordine con cui "vengono accesi".
La seconda precisazione riguarda la presenza di LD13÷LD28: dalla configurazione mostrata nello schema elettrico si può notare che questi led risultano sempre accesi. La loro utilità è puramente estetica; sono infatti utilizzati esclusivamente per illuminare le scale e le scritte presenti nella serigrafia del pannello anteriore dell'amplificatore.

Realizzazione pratica

Terminiamo l'articolo con la descrizione delle operazioni da compiere per realizzare e mettere in funzione il circuito.
Come sempre la prima cosa da fare è costruirsi le basette (vi ricordiamo che sono necessari due stampati): partite da una fotocopia delle tracce rame pubblicata e realizzatele mediante fotoincisione o PnP.
A questo punto è possibile iniziare a saldare i componenti in comune (indicati nel piano di montaggio). Successivamente le operazioni si differenziano a seconda che, per l'amplificatore, si sia scelta una configurazione stereo o mono.
Nel primo caso le due basette montano gli stessi componenti e andranno completate con le resistenze R1÷R13 e i led LD1÷LD28; vi facciamo però notare che questi ultimi andranno posizionati sul lato saldatura per il canale destro e sul lato componenti per il canale sinistro.
Nel secondo caso tra le due basette variano invece i valori assunti da alcuni elementi elettronici; infatti la prima basetta (scheda 1) dovrà essere montata sul lato sinistro, mentre la seconda (scheda 2, riconoscibile per la presenza del led rosso) andrà montata sul lato destro.
Durante le operazioni di saldatura, vi ricordiamo di prestare attenzione agli elementi che presentano polarità; per ogni dubbio fate sempre riferimento alle immagini pubblicate nel piano di montaggio.
Terminata la costruzione fissate i due misuratori al pannello anteriore dell'amplificatore, utilizzando le viti già presenti sullo stesso.
L'ultima operazione che rimane da compiere è il collegamento tra gli indicatori e i due circuiti che compongono l'amplificatore; in tutte le applicazioni i due misuratori (di destra e di sinistra) andranno collegati al relativo canale dell'amplificatore.
Utilizzando un filo di cablaggio sottile (0,5 mm) realizzate la connessione tra i punti LF, -V, GND e +V di ogni misuratore ai corrispondenti punti delle schede dell'amplificatore.

TRACCE LATO **RAME**



Tracce lato rame in scala 1:1 del circuito. Realizzatene una buona fotocopia e utilizzatela come base per costruire il circuito stampato mediante la tecnica della fotoincisione o il nuovo sistema Press 'n' Peel.

RM ELETTRONICA SNC
vendita componenti elettronici
rivenditore autorizzato:
FUTURA ELETTRONICA
NUOVA ELETTRONICA
G.P.E.
ELSE Kit
Via Val Sillaro, 38 - 00141 ROMA - tel. 06/8104753

Stimolatore per l'Elettroagopuntura Bifasica

di *Andrea Silvello*



Le prime notizie sull'agopuntura possono essere fatte risalire addirittura al XVI secolo avanti Cristo. La teoria dell'agopuntura si basa sull'assunto che nel corpo umano l'energia fluisca dagli organi interni alle estremità grazie ai cosiddetti "Meridiani di Energia", che sono la sua via di scorrimento "preferenziale". Fino a poco tempo fa l'esistenza dei meridiani non era completamente accettata dalla medicina moderna, ma da qualche anno diversi ricercatori ne hanno dimostrato l'esistenza tramite alcuni particolari esperimenti che hanno permesso di individuarne il percorso. L'energia, nel suo cammino, può incontrare degli "ostacoli": blocchi del flusso, carenze o eccessi dello stesso, percorsi in contro corrente, ecc. Tali impedimenti possono determinare delle anomalie del movimento energetico che, se non vengono trattate, finiscono per causare un danno biologico nei tessuti o negli organi interessati, dando origine a vari disturbi o alle malattie vere e proprie. Utilizzando metodologie proprie, la figura dell'agopuntore riesce a comprendere gli >

Il mondo dell'elettroagopuntura e, più in generale, quello dell'agopuntura è davvero vasto; questo articolo non ha la pretesa di presentarvi tali tecniche nel dettaglio o in modo esauriente. Se l'argomento vi interessa, il nostro suggerimento è di recarvi in libreria e acquistare uno tra i tanti libri dedicati a questa tecnica. Potrete trovare testi che analizzano la nascita, le prove scientifiche e i campi di applicazione dell'elettroagopuntura; altre pubblicazioni si occupano invece dei Meridiani di Energia, riportando tutti i 1170 punti di applicazione presenti nel corpo umano.
Sempre a proposito di questo argomento, ricordiamo che l'elettroagopuntura è una terapia medica e, come tale, non dovrebbe mai essere applicata senza cognizione di causa e senza aver prima sentito il parere di un medico.

Strumento progettato appositamente per l'applicazione della terapia medica dell'elettroagopuntura.
Prevede la regolazione dell'intensità della corrente utilizzata per il trattamento e la possibilità di applicare, mediante un opportuno manipolo, l'elettroagopuntura alla propria persona. Inoltre combinando il manipolo ed un elettrodo, il trattamento può essere applicato anche a terzi.
Completo di batterie ricaricabili che ne permettono l'utilizzo in qualsiasi luogo e di un circuito che provvede alla ricarica delle stesse tramite un alimentatore esterno a 12 Volt.

squilibri energetici che hanno portato ai disturbi del paziente; inoltre, stimolando in modo appropriato i punti sui meridiani, il terapeuta cerca di riportare lo stato energetico alla normalità.
Questo obiettivo può essere raggiunto in diversi modi: tramite l'infissione di aghi, il riscaldamento dei punti (moxibustione), il massaggio (shatzu) o mediante la stimolazione con blande cariche elettriche (elettroagopuntura).
In questo articolo prenderemo in considerazione proprio quest'ultima tecnica; senza entrare troppo nel dettaglio della pratica (a questo scopo, in libreria, sono presenti molti volumi dedicati all'argomento), l'elettroagopuntura prevede che come prima operazione si misuri il potenziale di resistenza dei punti in cui "passano" i meridiani di energia riuscendo quindi a stabilire per ogni organo o funzione metabolica corrispondente se è presente un equilibrio energetico, oppure se ci si trova di fronte ad uno stato di infiammazione, di intossicazione o di degenerazione (stati caratterizzati da una bassa resistenza dei punti). Qualora si verifichi una delle condizioni patologiche citate, come seconda operazione è necessario stimolare mediante deboli correnti elettriche i punti dei meridiani in modo da ristabilire l'equilibrio energetico.
Per una corretta stimolazione, le correnti utilizzate devono avere caratteristiche ben determinate: innanzitutto devono possedere delle corrette intensità, il cui amperaggio deve essere individuato dal curatore a seconda del caso che dovrà trattare. Inoltre, le correnti non devono essere di tipo continuo, ma devono essere composte da impulsi rettangolari applicati a coppie (uno positivo e uno negativo) caratterizzati dalla medesima ampiezza e aventi durata di circa 150 µsec e frequenza di circa 10Hz.
Negli ultimi anni l'elettroagopuntura è entrata a far parte delle medicine alternative ed è stata accettata anche dalla medicina ufficiale (soprattutto come sostegno alla terapia del dolore e delle malattie psicosomatiche); la scienza moderna spiega le funzioni dell'agopuntura secondo i tre seguenti assunti:
-agendo su alcuni punti del corpo umano, l'elettroagopuntura stimola il sistema nervoso a rilasciare sostanze chimiche (le endorfine) nei muscoli, nel midollo spinale e nel cervello. Questi elementi inizialmente hanno la funzione di cambiare la percezione del dolore; >



Il nostro dispositivo è composto da due basette, da 8 batterie ricaricabili da 1,2V e dal manipolo di applicazione del trattamento. La procedura di selezione del trattamento è completamente basata su un solo pulsante (TRIP).

successivamente determinano il rilascio di sostanze chimiche (i neuromediatori) che influenzano il sistema di regolazione interno dell'organismo;

-l'impulso elettrico dal punto di applicazione arriva fino al midollo spinale; qui riesce a "filtrare" gli stimoli provenienti dagli organi periferici, determinando quindi una riduzione della percezione del dolore;

-la scienza moderna sostiene che l'unità base del corpo sono le cellule e che queste seguono il movimento degli elettroni. Quando le cariche positive e negative sono squilibrate, la cellula non funziona correttamente, determinando quindi un disturbo; un'opportuna stimolazione elettrica può riportare l'equilibrio tra le cariche.

Vista una breve introduzione all'elettroagopuntura, passiamo ora a considerare il nostro dispositivo; questo è in grado di generare e fornire alle sue uscite le corrette forme d'onda di corrente; queste sono poi applicate al paziente mediante un manipolo e degli elettrodi. Su questo punto vi facciamo notare un particolare: la nostra apparecchiatura può essere utilizzata sia per un utilizzo personale sia per applicare l'elettroagopuntura a terze persone. Sul pannello anteriore sono infatti disponibili due uscite (OUT e AUX): la prima deve essere utilizzata per collegare il manipolo nel caso di utilizzo personale. Il "corpo" dello stesso realizza infatti il polo negativo, mentre la punta costituisce il polo positivo. In questo modo, afferrando il manipolo con una mano, questa diventerà la prima estremità; il secondo polo sarà invece dato dal punto del proprio corpo umano in cui si appoggerà la punta dello strumento. In questo modo, tra i due punti, si verificherà un passaggio di corrente che realizzerà l'elettroagopuntura. Nel caso invece si desideri applicare il trattamento a una seconda persona, è necessario collegare il manipolo all'uscita OUT e un elettrodo (disponibile come optional) all'uscita AUX. In questo modo l'elettrodo (che andrà posizionato in un punto preciso del corpo del paziente) sarà il polo negativo, mentre la punta del manipolo rappresenterà l'estremità positiva.

Posizionando quest'ultima sul corpo del soggetto si verificherà il passaggio di corrente che permetterà di applicare l'elettroagopuntura a terze persone.

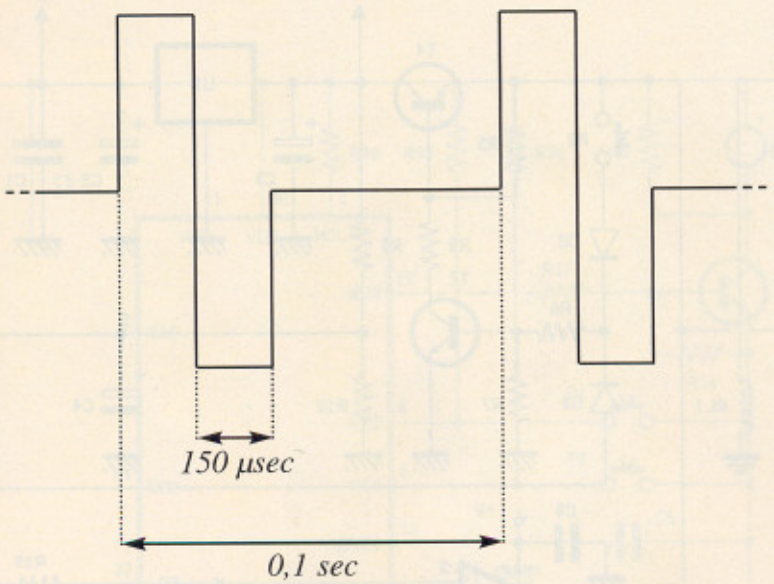
Il circuito richiede un'alimentazione di +12V; questa viene fornita da delle batterie ricaricabili presenti all'interno del dispositivo. Nel caso che le batterie inizino a scaricarsi, è presente un led che segnala la situazione all'utente; per caricarle è stata prevista la possibilità di collegare un alimentatore da +12V esterno. Vi facciamo notare che quando tale alimentatore viene collegato, la logica principale del circuito viene staccata; in questa situazione non è quindi possibile utilizzare lo stimolatore per applicare il trattamento.

L'interfaccia verso l'esterno del dispositivo è composta da un tasto di accensione (POWER ON), da un tasto utilizzato per selezionare le funzioni della terapia (TRIP) e da un tasto di STOP. Sono inoltre presenti tre led (SEARCH, ACTIVE e TIME) che indicano la funzione prescelta e due led che segnalano che le batterie sono scariche e che le stesse sono in carica. Infine è presente una serie di 10 led che realizza una scala graduata che, come vedremo, assume differenti significati a seconda di quale funzione del circuito risulta attiva.

Esempio di applicazione dell'elettroagopuntura

Prima di passare ad analizzare lo schema elettrico del circuito, vedia- >

LE FORME D'ONDA GENERATE



Affinché l'elettroagopuntura sia efficace è necessario che le forme d'onda generate siano composte da impulsi rettangolari applicati a coppie (uno positivo e uno negativo) caratterizzati dalla medesima ampiezza e aventi durata di 150 µsec e frequenza di 10Hz.

mo insieme un esempio pratico di applicazione del trattamento: come prima operazione è necessario collegare il manipolo all'uscita OUT del circuito. Successivamente premete il tasto POWER ON in modo da accendere il dispositivo; a questo punto i led che compongono la scala si accendono uno ad uno per evidenziare che il circuito è pronto all'utilizzo.

Premendo il tasto TRIP si attiva la funzione di ricerca del punto a minima resistenza del meridiano di energia (si accende il led SEARCH); muovete quindi la punta del manipolo sulla superficie del corpo umano fino a quando non trovate il punto corretto. La resistenza misurata viene mostrata attraverso la scala graduata (i led rossi, posizionati sul lato destro della stessa scala, indicano resistenze più basse). Quindi, quando trovate un punto in cui si accende l'ultimo led verso destra, significa che è stato trovato il corretto punto di applicazione.

A questo punto mantenete posizionato il manipolo e premete nuovamente il tasto TRIP per passare alla seconda funzione (segnalata dal led ACTIVE), ossia la selezione degli ampère della corrente di applicazione. Premete una o più volte il tasto TRIP per aumentare l'amperaggio (che viene indicato attraverso la scala graduata); ci sono 10 secondi di tempo a disposizione per modificare la corrente. Trascorso questo intervallo il circuito passa automaticamente nella terza e ultima fase, quella di visualizzazione dello scorrere del tempo (TIME) in cui viene applicato il trattamento. Questa fase dura circa 60 secondi (la scala graduata indica lo scorrere del tempo) e in qualsiasi istante può essere bloccata premendo il tasto STOP.

Un'ultima caratteristica che vi facciamo notare è che il circuito dispone di un meccanismo di autospegni- >

CELLULAR KILLER CPM-043
2003 EDITION
7 W e.r.p.
Rinnovato
Ancora più efficace
Rispondente alle severe norme di sicurezza ANSI-IRPA-NATO





CELLULAR KILLER "small" CPM-056
200 mW
raggio di azione 2-10 mt.

DETECTOR PER CELLULARI DMC
Ideale per monitorare la sicurezza in uffici e mezzi mobili
Range 4-25 mt.



RICEVITORE DI ANALISI E SORVEGLIANZA MRA
Rileva istantaneamente qualsiasi nuova trasmissione radio indesiderata.
Range 43-2700 MHz.
100 volte più veloce di qualsiasi scanner
MRA



AMPLIFICATORE BOOSTER
1,5 W da 5 mW di ingresso.
Range 0,2-2 GHz
Ideale per amplificare tx video, audio, per strumentazione, antenne
CPM-055



SISTEMA DI SORVEGLIANZA PERIMETRALE WIRELESS SOLARBEAM
Per grandi spazi, N°1 in U.S.A. !

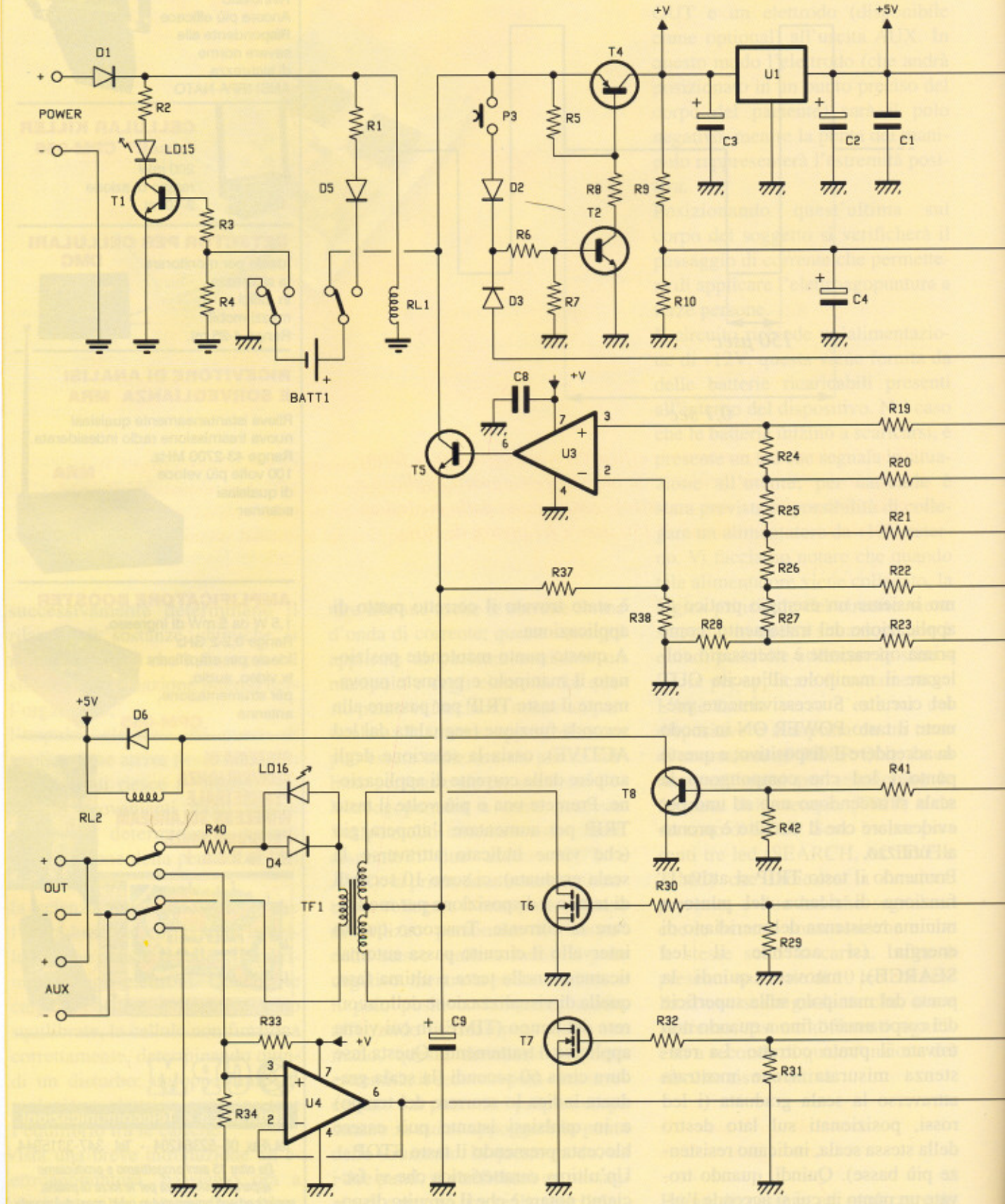


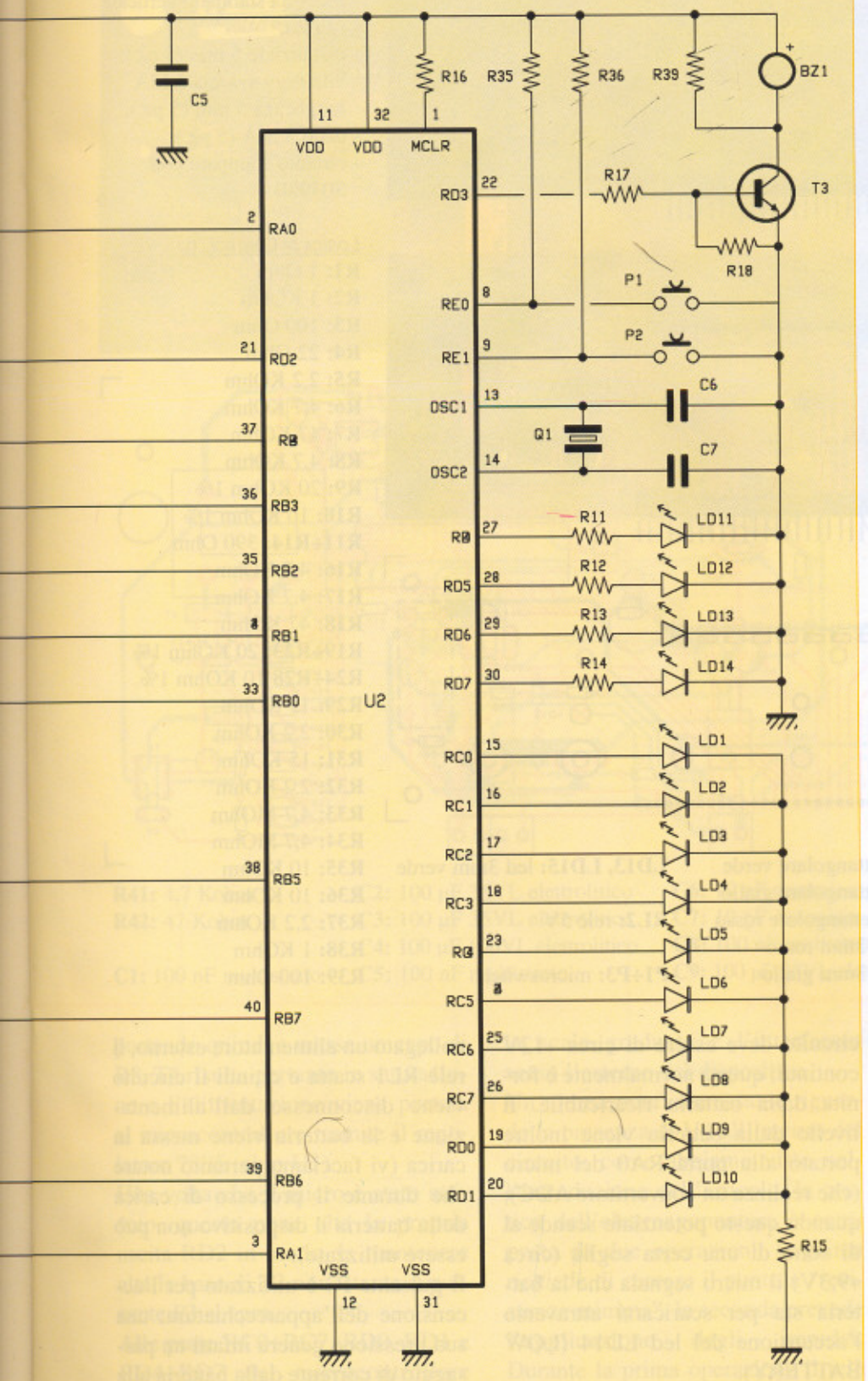




CPM
Elettronica e Telecomunicazioni
Tel./Fax 06-52364204 - Tel. 347-3315944
Da oltre 15 anni progettiamo e produciamo apparati di sicurezza per le forze di polizia, società ed enti governativi in molti paesi del mondo
WWW.CPMELETTRONICA.COM

Schema Elettrico





mento; infatti passati circa 30 secondi senza che non venga eseguita nessuna azione, il circuito si autodisattiva.

L'esempio appena visto riguarda il caso di impiego personale; la procedura da seguire nel caso di applicazione a terzi individui rimane essenzialmente la stessa; l'unica differenza è che in questo caso bisogna collegare anche l'elettrodo all'uscita AUX.

Schema elettrico

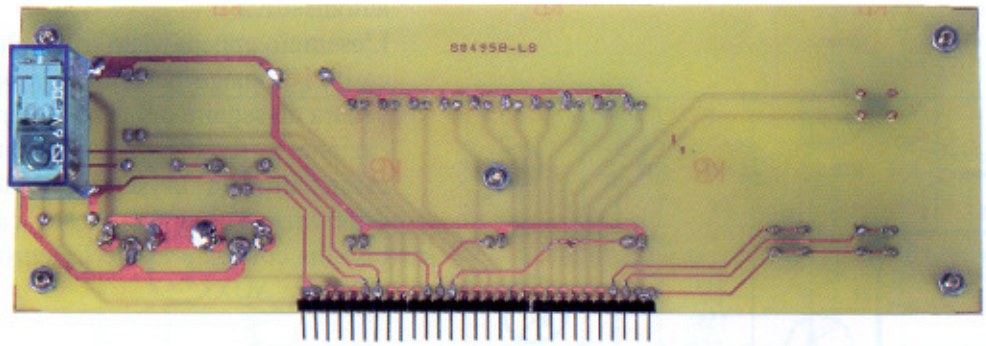
Essenzialmente la funzione principale del circuito è generare due onde quadre che, come abbiamo già visto, devono essere una positiva e una negativa e avere stessa ampiezza; inoltre abbiamo già sottolineato che nel circuito sono presenti due uscite (OUT e AUX) collegate in parallelo.

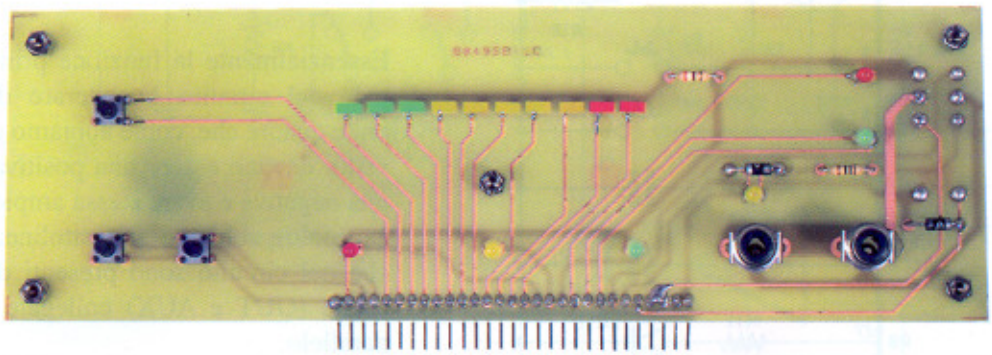
Il cuore del circuito è il microcontrollore PIC16F877 (chip U2) che genera gli impulsi tramite i pin RB6 e RB7; questi impulsi comandano lo stadio push pull (transistor T6 e T7) che erogano una corrente dipendente dagli stati delle uscite RB0÷RB4 del micro.

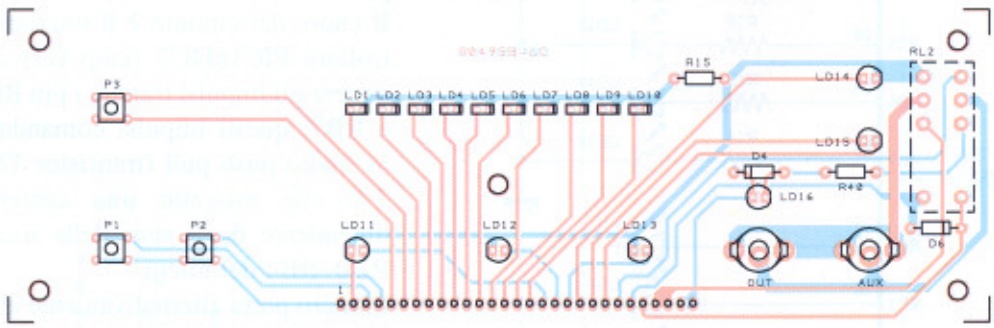
Il micro porta alternativamente alti i pin RB6 e RB7, in modo da far saturare (e quindi cortocircuitare) un singolo transistor (T6 o T7) alla volta. Quando RB7 è alto, T6 è un cortocircuito e quindi sulle uscite viene applicato un'onda quadra positiva; quando invece è RB6 a essere alto, T7 è un corto e quindi sull'uscita viene generata un'onda quadra negativa.

La corrente generata da T6 e T7 dipende dal valore logico assunto dalle porte RB0÷RB4 secondo il seguente meccanismo: in base a quale tra queste 5 porte risultano alte, cambia il potenziale applicato all'ingresso non invertente dell'operazionale U3. L'uscita di quest'ultimo comanda la base di T5 che quindi fornirà al proprio emet- >

PIANO DI MONTAGGIO







Varie:
- strip 29 poli maschio 90°;
- RCA da stampato verticale (2 pz.);
- distanziale 5 mm (5 pz.);
- vite testa svasata 3 MA lunghezza 5 mm (5 pz.);
- dado 3 MA (5 pz.);
- circuito stampato cod. S0495B.

LOGICA DI CONTROLLO:
R1: 1 Ohm
R2: 1 KOhm
R3: 100 Ohm
R4: 22 Ohm
R5: 2,2 KOhm
R6: 4,7 KOhm
R7: 47 KOhm
R8: 4,7 KOhm
R9: 20 KOhm 1%
R10: 10 KOhm 1%
R11÷R14: 390 Ohm
R16: 4,7 KOhm
R17: 4,7 KOhm
R18: 47 KOhm
R19÷R23: 20 KOhm 1%
R24÷R28 10 KOhm 1%
R29: 15 KOhm
R30: 2,2 KOhm
R31: 15 KOhm
R32: 2,2 KOhm
R33: 4,7 MOhm
R34: 4,7 MOhm
R35: 10 KOhm
R36: 10 KOhm
R37: 2,2 KOhm
R38: 1 KOhm
R39: 100 Ohm

INTERFACCIA:
R15: 390 Ohm
R40: 10 Ohm

D4,D6: 1N4007

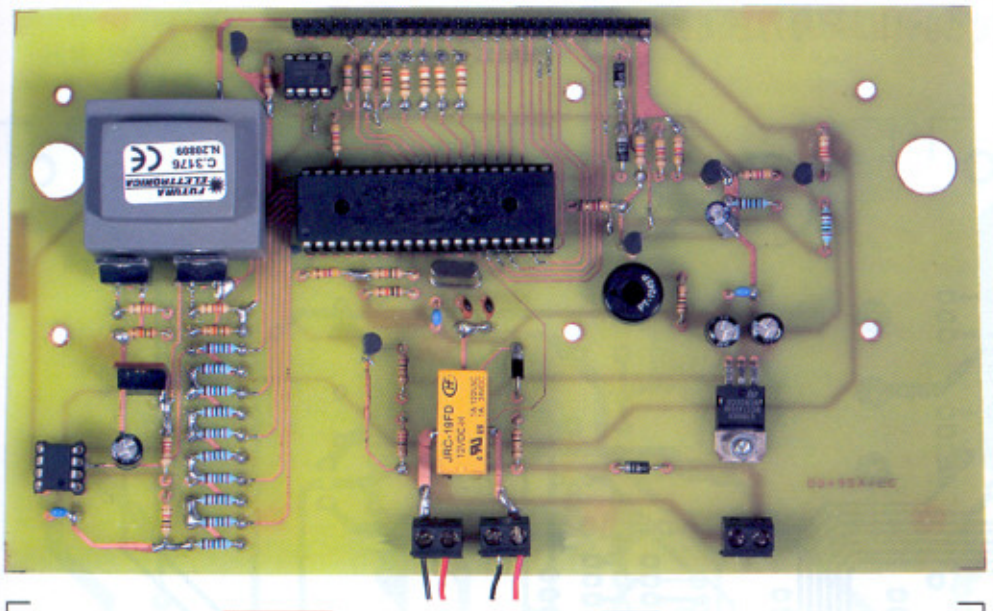
LD1÷LD3: led rettangolare verde
LD4÷LD8: led rettangolare giallo
LD9, LD10: led rettangolare rosso
LD11, LD14: led 3mm rosso
LD12, LD16: led 3mm giallo

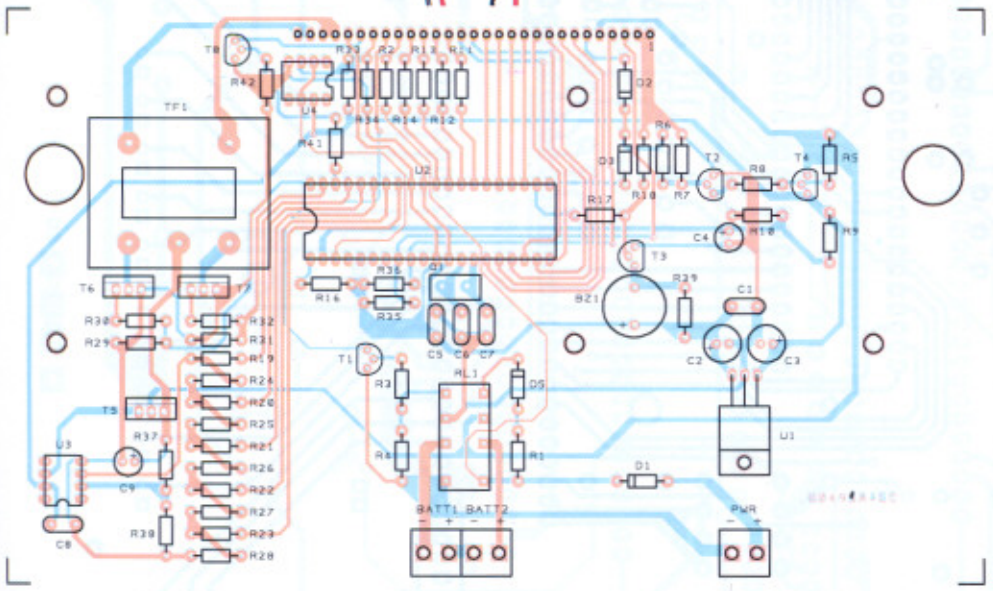
LD13, LD15: led 3mm verde

RL2: relè 5V

P1÷P3: microswitch

titore una corrente dipendente dalla tensione applicata alla propria base. La corrente erogata dall'emettitore di T5 entrerà nella presa centrale del primario di TF1 e si dirigerà in una delle due sezioni del primario a seconda di quale transistor tra T6 e T7 sarà in saturazione.
In questo modo, a seconda della direzione assunta dalla corrente, sul secondario di TF1 verranno generate delle onde quadre di segno positivo o negativo. L'alimentazione del circuito deve essere di circa +12V continui; questa normalmente è fornita dalla batteria ricaricabile. Il livello della batteria viene inoltre portato alla porta RA0 del micro (che realizza un convertitore ADC); quando questo potenziale scende al di sotto di una certa soglia (circa +9,3V) il micro segnala che la batteria sta per scaricarsi attraverso l'accensione del led LD14 (LOW BATTERY).
Quando ai morsetti POWER viene collegato un alimentatore esterno, il relè RL1 scatta e quindi il circuito viene disconnesso dall'alimentazione e la batteria viene messa in carica (vi facciamo pertanto notare che durante il processo di carica della batteria il dispositivo non può essere utilizzato).
Il pulsante P3 è utilizzato per l'accensione dell'apparecchiatura; una sua pressione genera infatti un passaggio di corrente dalla batteria alla base di T2; questo va in saturazione >





D1, D2: 1N4007
D3, D5: 1N4007

U1: 7805
U2: PIC16F877 (cod. MF495)
U3, U4: CA3140

T1÷T4: BC547
T5: TIP122
T6, T7: IRFZ44N
T8: BC547

RL1: relè 12V 2 sc.

Q1: quarzo 20 MHz

BZ1: buzzer senza elettronica

TF1: trasformatore elevatore in ferrite 3176

Varie:
- morsetto 2 poli (3 pz.);
- zoccolo 20 + 20 pin;
- zoccolo 4 + 4 pin (2 pz.);
- strip 29 poli femmina;
- clip per batteria 9V;
- portabatterie 4 AA (2 pz.);
- vite testa svasata 3 MA lunghezza 5 mm (1 pz.);
- vite autofilettante 3 MA lunghezza 5 mm (6 pz.);
- dado 3 MA (1 pz.);
- circuito stampato cod. S0495A.

Il circuito è composto da due basette: la prima realizza l'interfaccia del dispositivo mentre la seconda la logica di controllo.

R41: 4,7 Kohm
R42: 47 Kohm

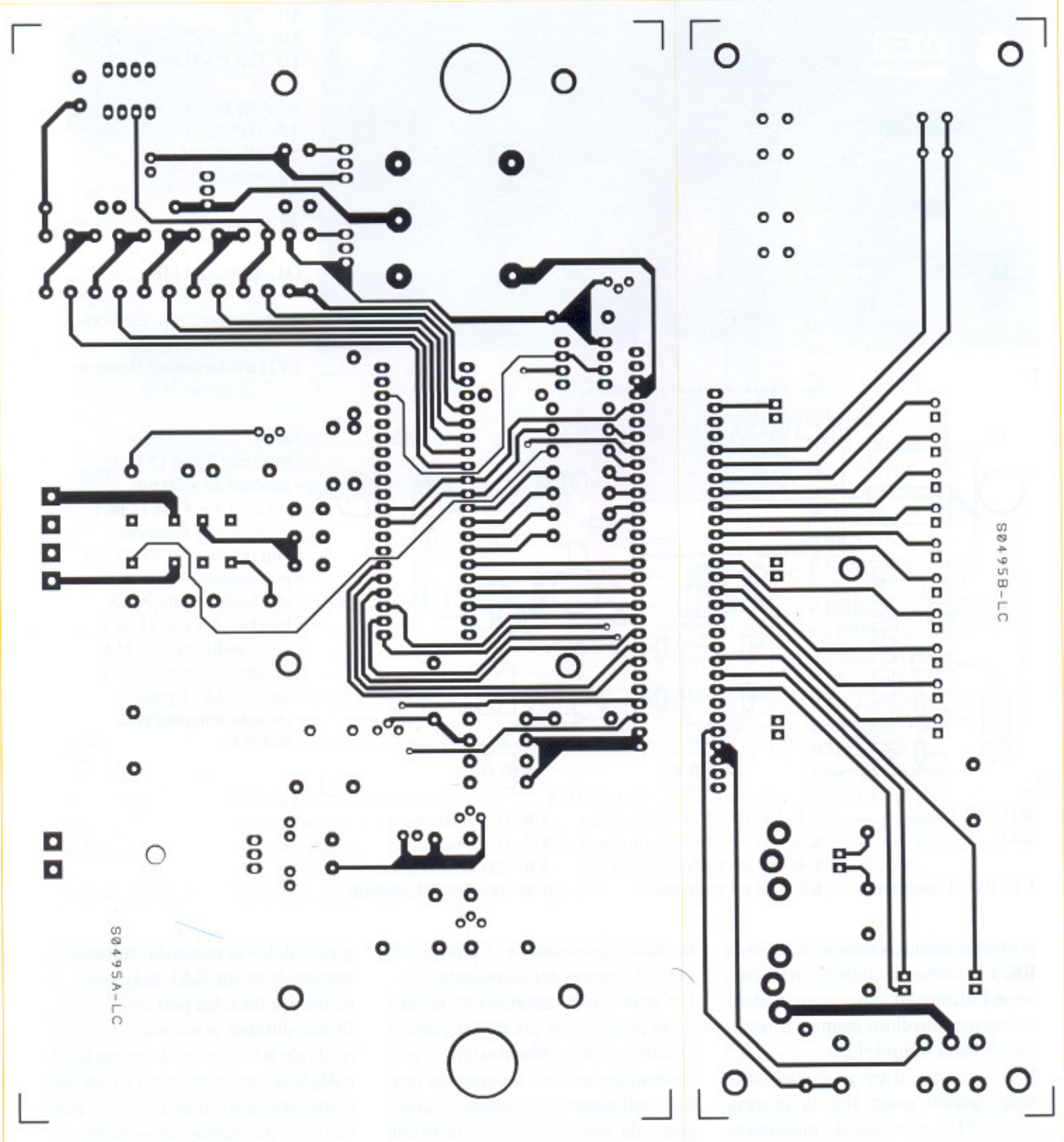
C1: 100 nF multistrato
C2: 100 µF 35VL elettrolitico
C3: 100 µF 35VL elettrolitico
C4: 100 µF 100VL elettrolitico
C5: 100 nF multistrato
C6: 10 pF ceramico
C7: 10 pF ceramico
C8: 100 nF multistrato
C9: 100 µF 35VL elettrolitico

portando quindi a massa un capo di R8. T4 risulta pertanto in saturazione e l'alimentazione viene portata al microcontrollore tramite il regolatore 7805 (chip U1).
Una volta che il micro è stato attivato, questo pone alta la propria uscita RD2 in modo da mantenere alta la base di T2 nonostante il pulsante P3 si apra.
Alle porte RC0÷RC7, RD0, RD1 e RD4÷RD7 del microcontrollore sono collegati i led che realizzano la scala graduata e l'interfaccia verso l'esterno del dispositivo.
Un'ultima nota interessante riguarda la presenza di U4 all'interno del circuito; come abbiamo visto nell'introduzione dell'articolo, la pratica dell'elettroagopuntura è composta da due operazioni: la prima richiede di cercare il punto a "resistenza minima", la seconda prevede l'applicazione degli impulsi. Durante la prima operazione il relè RL2 collega l'uscita OUT all'ingresso di U4 in modo da "misurare", tramite la porta RA1 del micro, la resistenza tra i due poli dell'uscita. Invece durante la seconda operazione il relè RL2 collega le uscite OUT e AUX al trasformatore TF1 da cui, come abbiamo appena visto, provengono gli impulsi di corrente.

Realizzazione pratica

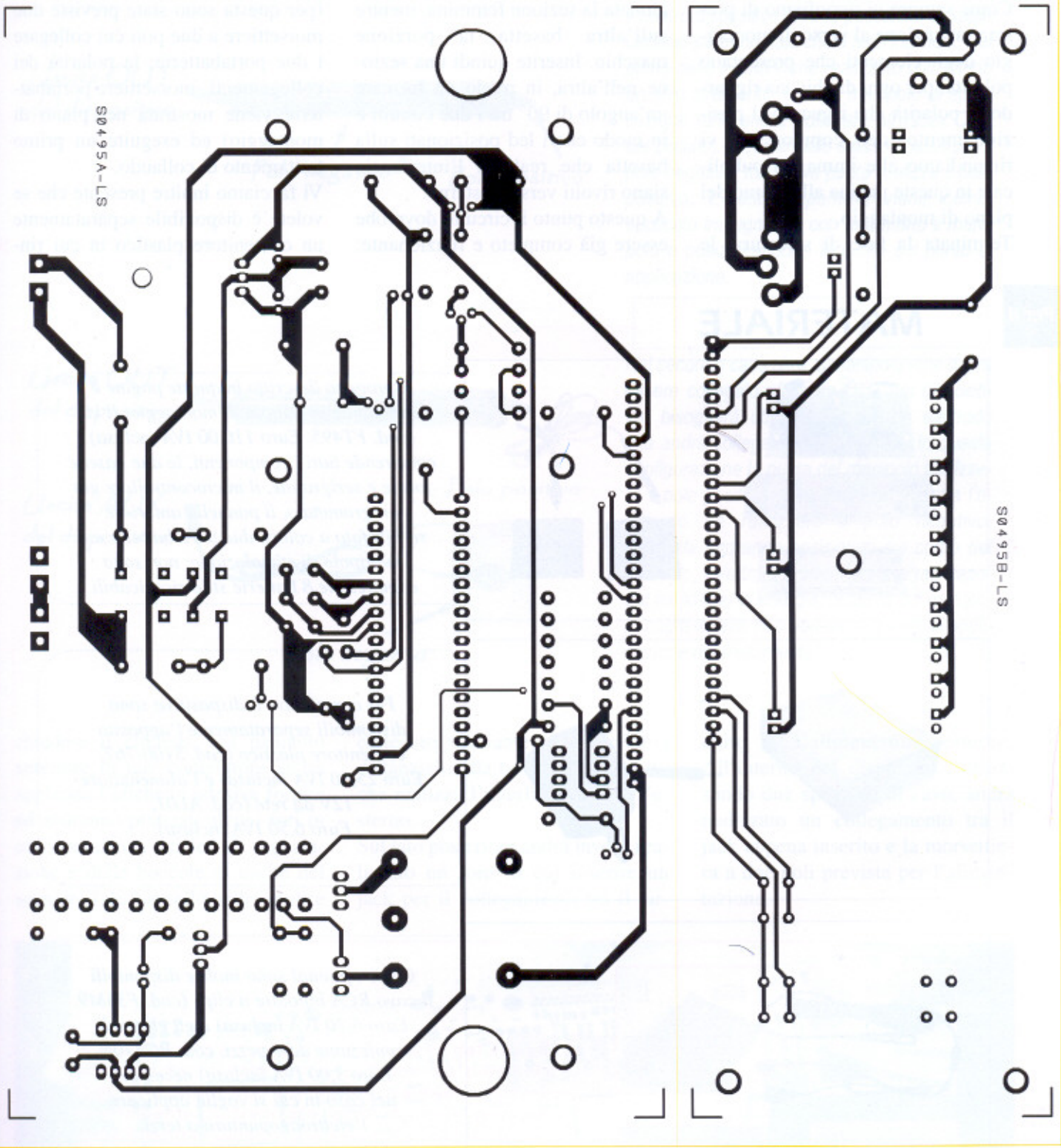
L'ultima sezione dell'articolo riguarda come costruire e mettere >

TRACCE IN RAME LATO COMPONENTI



in funzione il dispositivo. La prima operazione è quella di costruirsi le due basette che realizzano il circuito: la prima di dimensioni maggiori contiene la logica del dispositivo; la seconda di dimensioni più piccole realizza invece l'interfaccia verso l'esterno. Partite quindi dalle tracce rame pubblicate in queste pagine e realizzatele mediante fotoincisione o Press 'n' Peel. Entrambe le basette sono doppia faccia; in queste >

TRACCE IN RAME LATO SALDATURE

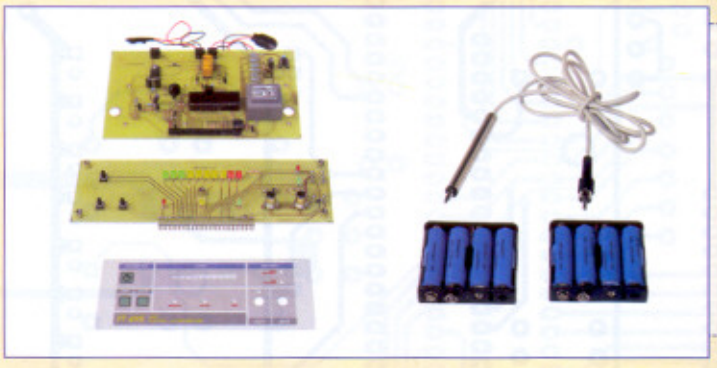


pagine sono infatti pubblicate quattro immagini (due per il lato componenti e due per il lato rame).
Una volta pronte le basette è il momento di iniziare l'operazione di saldatura degli elementi. Per la basetta caratterizzata dalle dimensioni più grandi tutti gli elementi andranno posizionati solo sul lato componenti, mentre per l'altra basetta alcuni elementi (il relè e i 29 connettori) andranno posizionati >

sul lato saldature. Vi facciamo notare che, essendo le basette doppia faccia, alcune saldature andranno realizzate anche sul lato componenti.
Come sempre vi ricordiamo di prestare attenzione al verso di montaggio degli elementi che presentano polarità; per ogni dubbio sia riguardo la polarità che riguardo il posizionamento dei componenti vi rimandiamo alle immagini pubblicate in queste pagine all'interno del piano di montaggio.
Terminata la fase di saldatura, le due basette andranno collegate insieme. Per l'operazione è stata prevista un'apposita strip a 29 poli; sulla basetta più grande è stata posizionata la sezione femmina, mentre sull'altra basetta la porzione maschio. Inserite quindi una sezione nell'altra, in modo da formare un'angolo di 90° tra i due circuiti e in modo che i led posizionati sulla basetta che realizza l'interfaccia siano rivolti verso l'esterno.
A questo punto il circuito dovrebbe essere già completo e funzionante; collegate quindi l'uscita OUT (che, vi facciamo notare, è quella posizionata più a sinistra) al manipolo e fornite l'alimentazione al circuito (per questa sono state previste due morsettiere a due poli cui collegare i due portabatterie; la polarità dei collegamenti morsettiere-portabatterie viene mostrata nel piano di montaggio) ed eseguite un primo trattamento di collaudo.
Vi facciamo inoltre presente che se volete è disponibile separatamente un contenitore plastico in cui rin- >

Per il

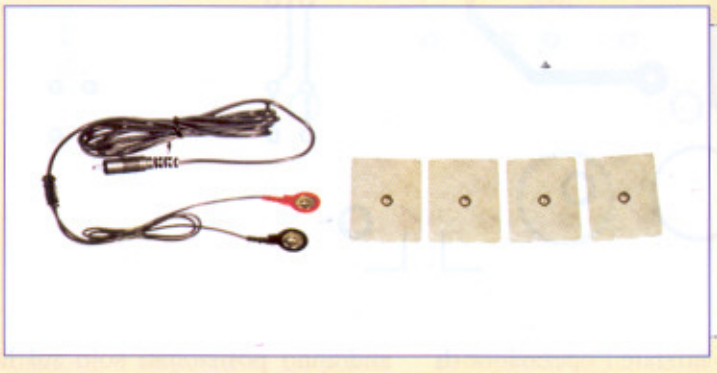
MATERIALE



Il progetto descritto in queste pagine è disponibile in scatola di montaggio. Il kit (cod. FT495, Euro 110,00 IVA inclusa) comprende tutti i componenti, le due basette forate e serigrafate, il microcontrollore già programmato, il pannello anteriore serigrafato a colori, due portabatterie e il manipolo di stimolazione; non sono comprese le 8 batterie stilo ricaricabili.



Per completare il dispositivo sono disponibili separatamente l'apposito contenitore plastico (cod. 5100-767, Euro 25,80 IVA inclusa) e l'alimentatore 12V da rete (cod. AL01, Euro 6,50 IVA inclusa).



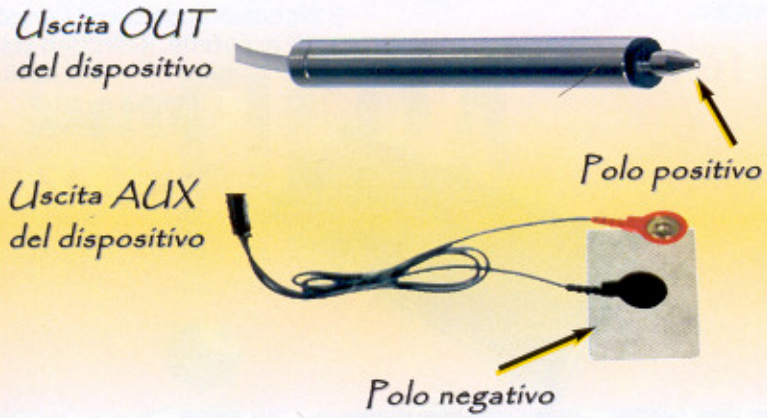
Come optional sono inoltre disponibili il cavo RCA bipolare a clips (cod. F35M9, Euro 6,20 IVA inclusa) e gli elettrodi (confezione da 4 pezzi, cod. PG470N, Euro 3,00 IVA inclusa) necessari nel caso in cui si voglia applicare l'elettroagopuntura a terzi.

Il materiale va richiesto a: Futura Elettronica, V.le Kennedy 96, 20027 Rescaldina (MI).
Tel: 0331576139 ~ Fax: 0331466686 ~ Sito: *www.futuranet.it*

APPLICAZIONE PERSONALE O A TERZI



Il nostro dispositivo offre la possibilità di applicazione sia personale che a terzi individui. Nel primo caso il tutto viene realizzato utilizzando un solo strumento, ossia il manipolo di applicazione (che andrà collegato all'uscita OUT del circuito). Il corpo dello stesso costituisce il polo negativo, mentre la punta realizza il polo positivo. L'elettroagopuntura viene quindi applicata impugnando con una mano il manipolo e posizionandone la punta sul punto di applicazione.



Nel secondo caso oltre al manipolo (che dovrà essere collegato all'uscita OUT del dispositivo) bisognerà utilizzare anche un elettrodo che andrà collegato all'uscita AUX. In questa configurazione la punta del manipolo realizzerà il polo positivo di applicazione, mentre l'elettrodo ne realizzerà il polo negativo. Entrambi andranno posizionati sul corpo del paziente; l'elettrodo andrà fissato a un "capo" del meridiano di energia; la punta del manipolo andrà invece posizionata sul punto di applicazione del trattamento.

chiudere il circuito. Sul pannello anteriore di tale contenitore andrà applicata l'etichetta adesiva fornita ed andranno praticati alcuni fori in corrispondenza dei led di segnalazione e delle boccole di uscita del segnale. Tale pannello andrà inoltre collegato, mediante 4 viti e relativi distanziali, alla basetta del circuito che realizza l'interfaccia verso l'esterno.

Sul lato posteriore andrà invece realizzato un foro in cui inserire un jack per il collegamento tra il circuito e l'alimentatore esterno. All'interno del dispositivo, utilizzando due spezzoni di cavo, andrà realizzato un collegamento tra il jack appena inserito e la morsettiera a due poli prevista per l'alimentazione.



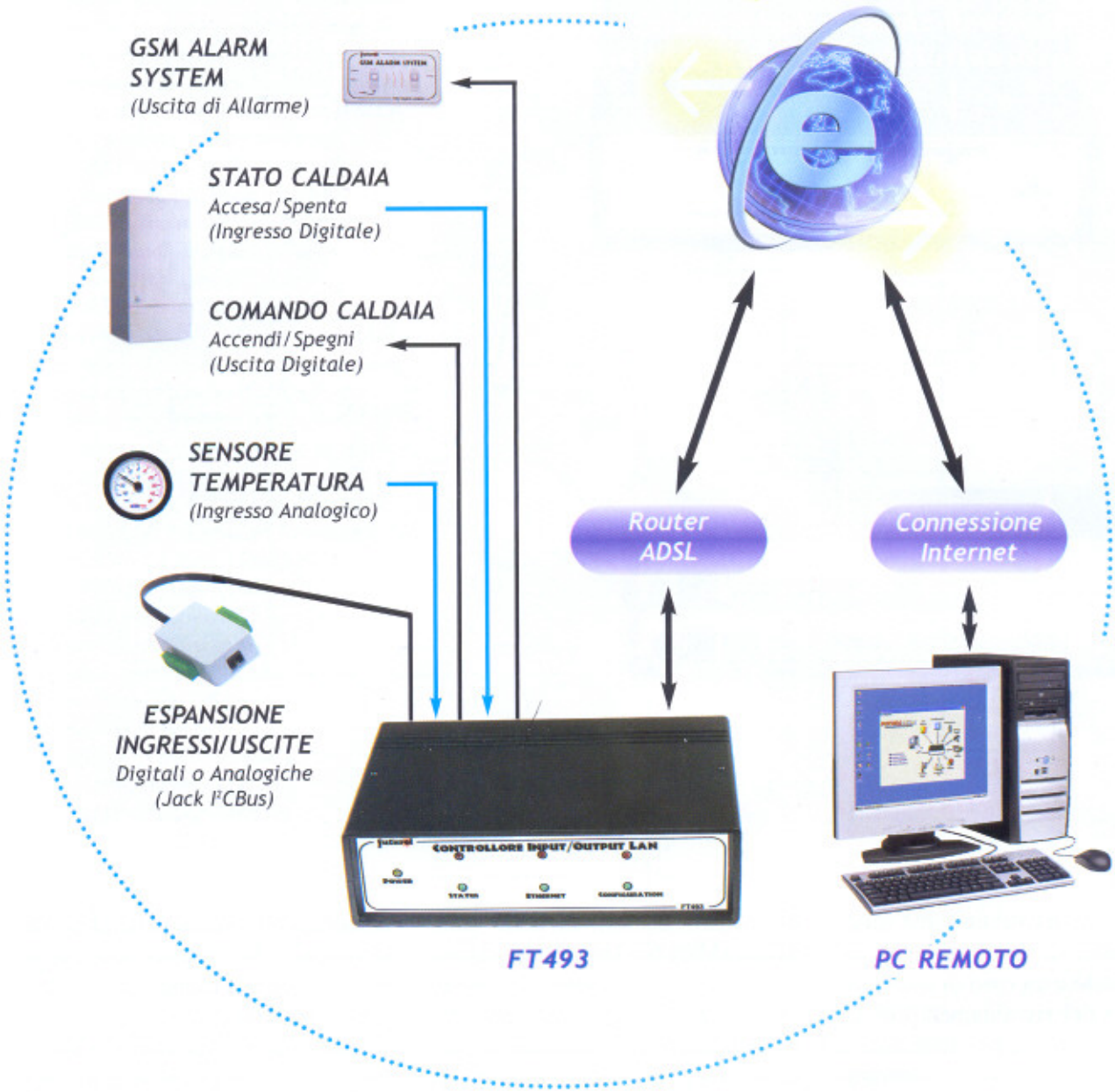
Controller Domotico Gestibile via Internet

di Boris Landoni

Dispositivo di gestione di ingressi e uscite digitali o analogiche che può essere controllato tramite Internet. Ideale per applicazioni di controllo remoto di un'abitazione. Munito di un'uscita di allarme e di una porta I²CBus utilizzabile per espandere il sistema con nuovi ingressi o uscite, sia digitali che analogiche.



Sono trascorsi alcuni mesi da quando sulla rivista n. 77 abbiamo presentato l'innovativo modulo ethernet Tibbo EM100; la caratteristica peculiare ed interessante di questo componente è che risulta munito di due porte (una seriale e una ethernet) ed è in grado di fornire e convertire il formato dei dati in arrivo su una porta all'altra e viceversa. Utilizzando questo modulo risulta quindi possibile collegare ogni periferica equipaggiata di porta seriale RS232 ad una rete LAN e raggiungerla tramite connessioni basate sul protocollo TCP/IP. La periferica sarà quindi accessibile da ogni postazione connessa alla rete locale; addirittura se la LAN ha un accesso verso l'esterno sarà possibile accedervi da qualunque computer connesso ad internet. Nello scorso numero della rivista abbiamo presentato un primo progetto che si basava sull'utilizzo del modulo ethernet Tibbo EM100; brevemente vi ricordiamo che era stato realizzato un controllo remoto di due relè che, nel caso specifico, erano stati utilizzati per accendere, spegnere o resettare un PC. In questo articolo vi >

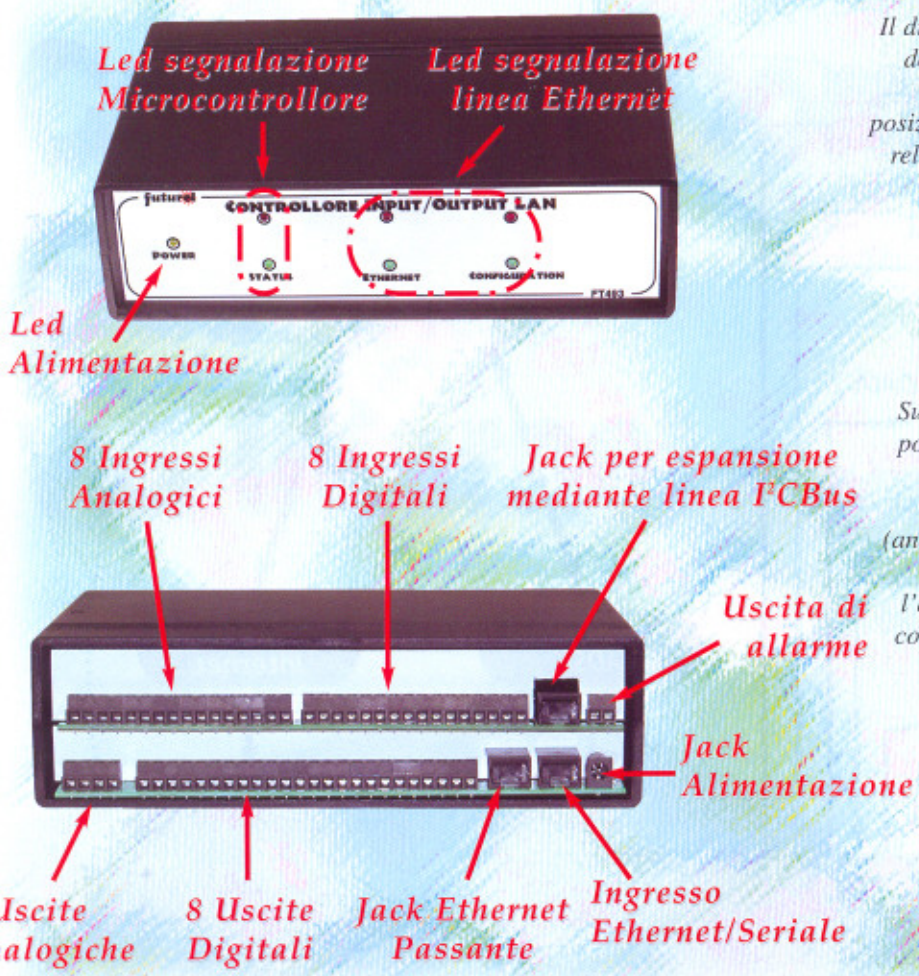


presentiamo invece un nuovo progetto più complesso e caratterizzato da un utilizzo finale più generale. Scopo del nuovo sistema è quello di rendere disponibili una serie di ingressi e uscite sia digitali che analogiche che, utilizzando il modulo ethernet EM100, saranno raggiungibili da qualsiasi PC connesso a internet. In particolare nel circuito sono presenti 8 ingressi digitali, 8 uscite digitali, 8 ingressi analogici, 2 uscite analogiche più un'uscita di allarme; inoltre è disponibile un jack che, come vedremo in seguito, consentirà di aggiungere al dispositivo nuovi ingressi o uscite, sia digitali che analogiche.

Chiaramente, per poter essere raggiungibile da internet, è necessario che il dispositivo sia collegato a quest'ultima, sia direttamente (per esempio tramite un router ADSL), sia attraverso una LAN munita di connessione internet.

Gli utilizzi finali del sistema sono vari e applicabili a tutte quelle situazioni in cui risulta necessario controllare alcuni ingressi o uscite; nell'immagine presente in apertura >

INTERFACCE DEL DISPOSITIVO



Il dispositivo è caratterizzato da una doppia interfaccia verso l'esterno. Sul pannello anteriore sono posizionati alcuni led di segnalazione relativi all'alimentazione, allo stato del microcontrollore presente nel circuito e allo stato della linea ethernet (questi ultimi, come è possibile notare osservando lo schema elettrico, sono gestiti direttamente dal modulo ethernet Tibbo EM100). Sul pannello posteriore sono invece posizionate le morsettiere a 2 o a 3 poli che realizzano gli ingressi e le uscite (analogici o digitali) del dispositivo. Sono inoltre presenti il jack per l'alimentazione (+12V continui), il connettore di espansione I²CBus e i due jack (connessi tra loro in parallelo) utilizzabili per il collegamento del dispositivo alla rete locale (ethernet) o ad un PC direttamente via porta seriale (la selezione tra rete LAN o collegamento seriale avviene mediante DS1).

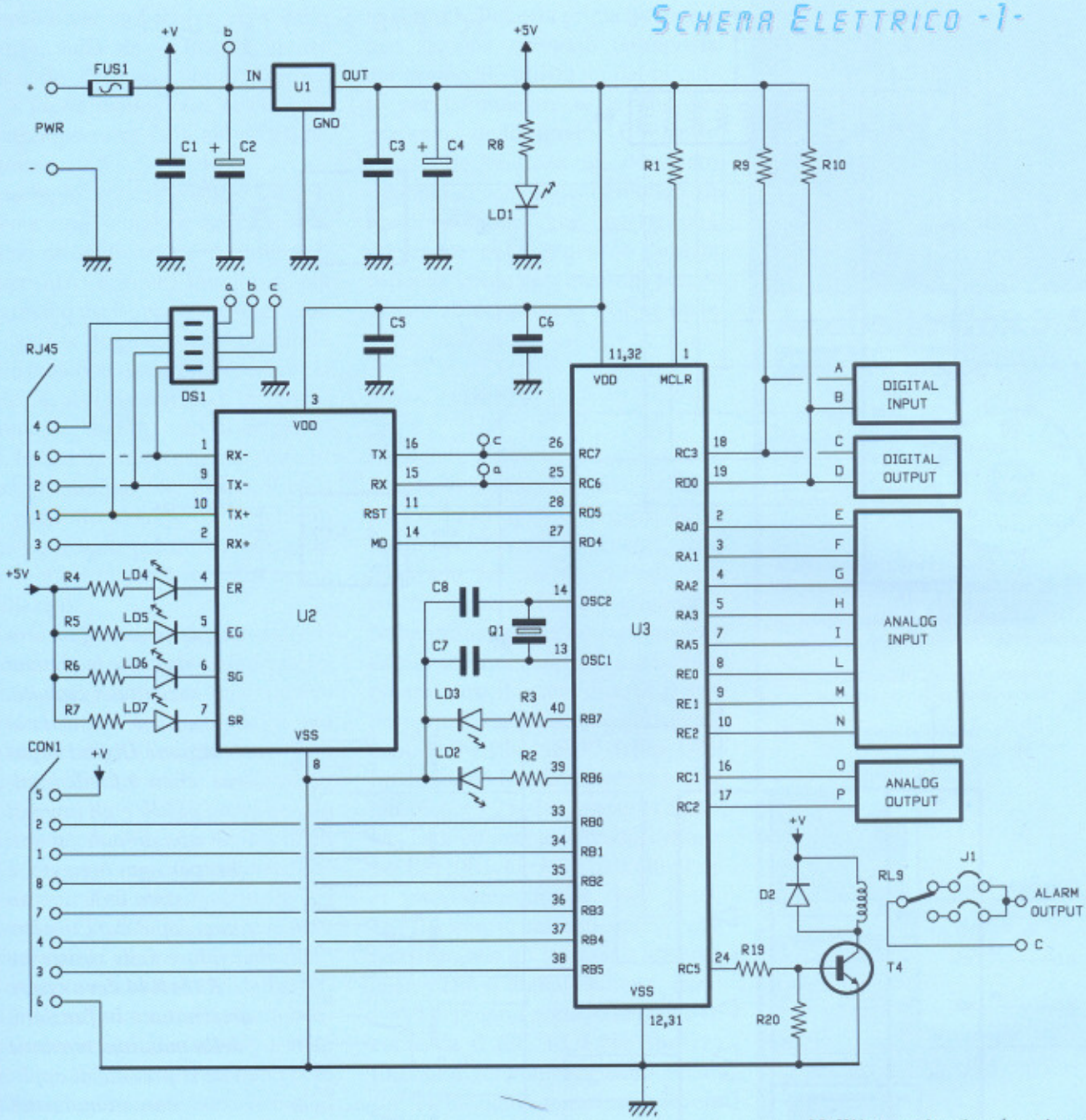
di articolo vi mostriamo una schematizzazione di un possibile esempio applicato a un caso di domotica (controllo del riscaldamento di un ambiente). Come è possibile notare è presente un sensore di temperatura collegato a un ingresso analogico del circuito; un'uscita digitale risulta invece collegata a un sistema che consente di accendere/spegnere una caldaia. In questo modo l'utente, collegandosi via software da un PC remoto tramite internet, sarà in grado di andare a leggere la temperatura misurata dal sensore ed eventualmente potrà comandare l'accensione o lo spegnimento della caldaia.

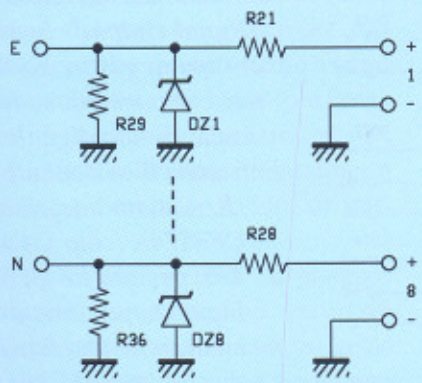
Nello schema è inoltre presente un dispositivo di verifica dello stato della caldaia (collegato a un ingresso digitale del circuito); in questo modo l'utente, dopo aver comandato di accendere o spegnere la caldaia, potrà controllare (sempre da remoto) che il comando sia stato effettivamente eseguito.

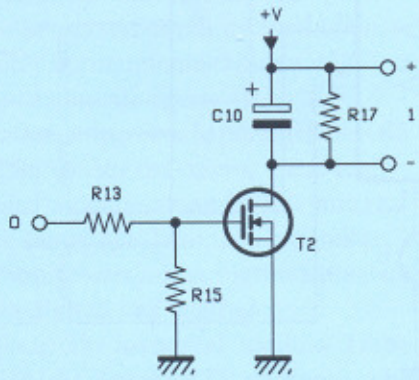
Nell'esempio e precedentemente è stato fatto riferimento anche ad un'uscita di allarme; il circuito dispone infatti di una particolare funzionalità che permette di attivare quest'uscita nel caso in cui risultino verificate certe condizioni (specificabili via software) su alcuni ingressi (per esempio se un ingresso digitale diventa alto o basso o se un ingresso analogico non risulta compreso all'interno di una finestra di valori). Nel nostro schema di esempio, questa uscita è stata collegata a un *GSM Alarm System* (presentato nella rivista numero 66); in questo modo se la temperatura rilevata dal sensore sale al di sopra di una certa soglia, l'uscita di allarme verrà attivata e quindi l'utente verrà avvisato tramite l'invio di un SMS.

Nello schema è inoltre visibile un'espansione degli ingressi o delle uscite; nel caso infatti che quelli disponibili direttamente sul circuito non siano sufficienti per le vostre applicazioni, il dispositivo è munito di una porta I²CBus cui possono essere collegate un numero massimo di 8 *espansioni a 8 uscite o 8* >

SCHEMA ELETTRICO -1-

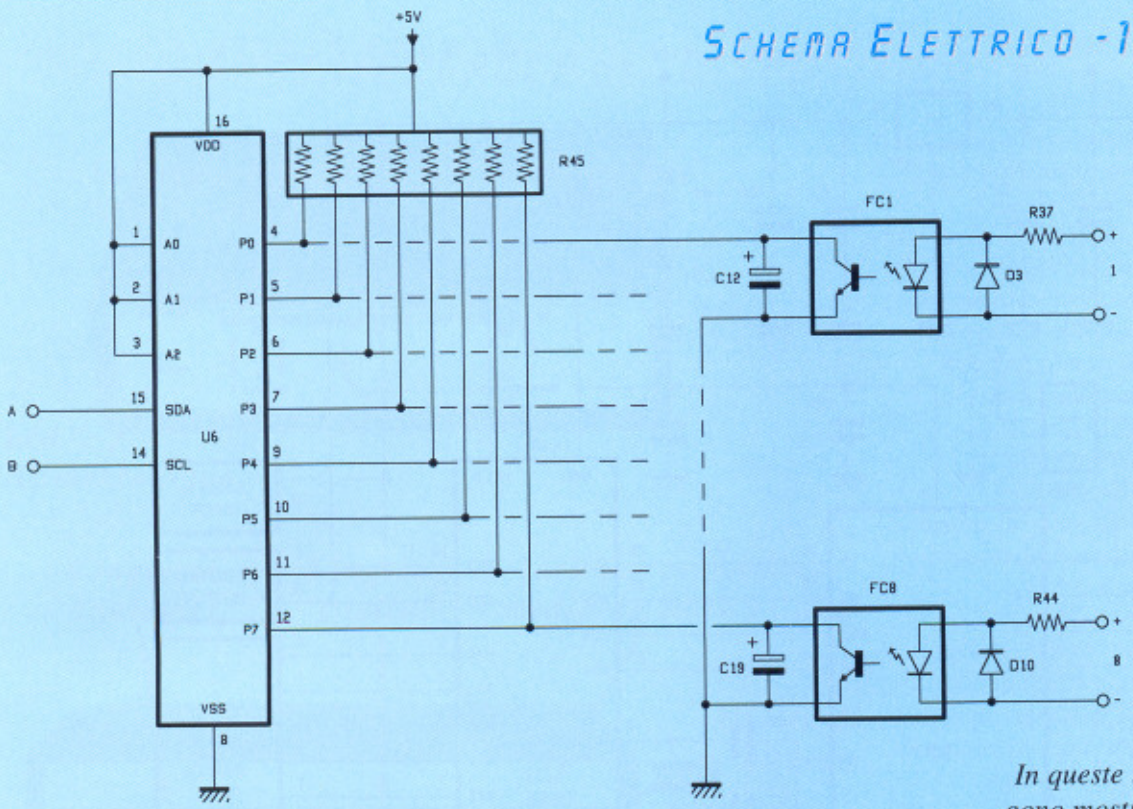


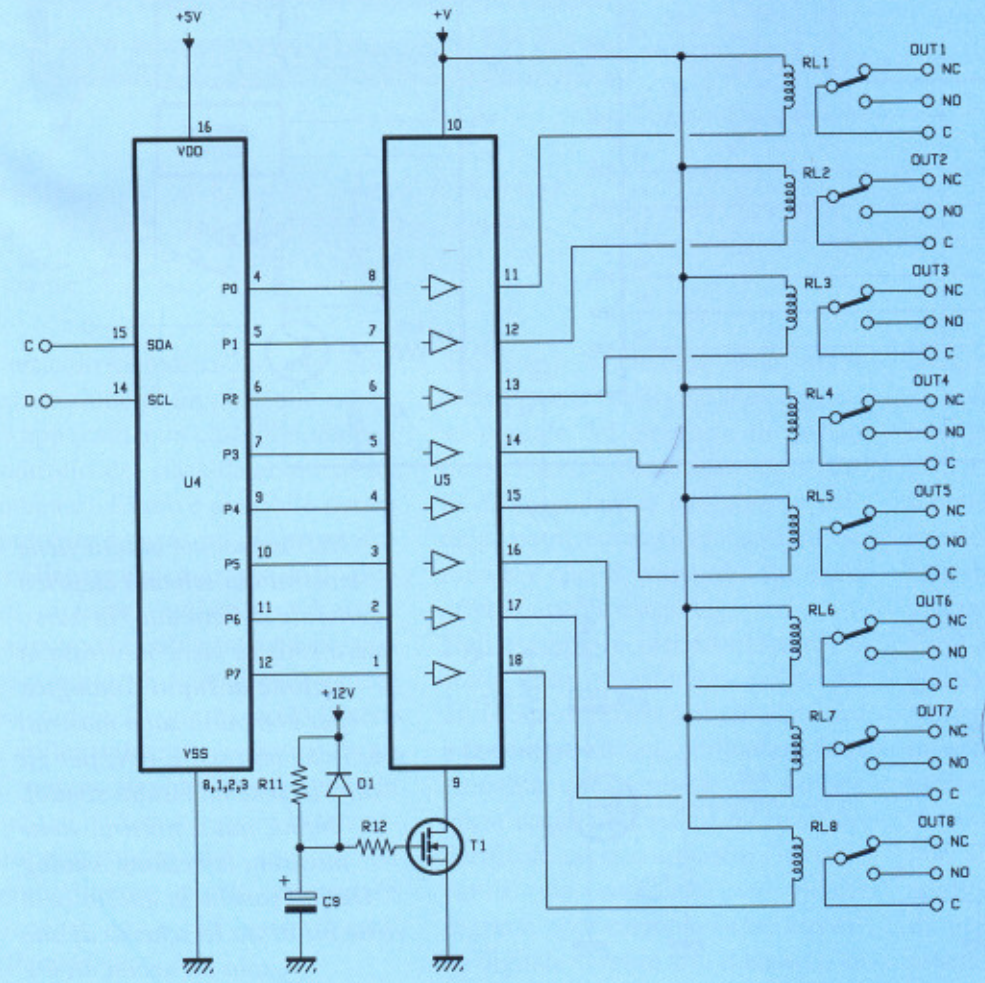




Nell'immagine in alto viene mostrato lo ***schema elettrico generale*** *del circuito; in basso più a sinistra viene mostrata la* ***sezione di Input Analogica*** *(per comodità sono mostrati solo i due ingressi E e N; per gli altri lo schema risulta simile). Infine, qui a sinistra, viene mostrata la* ***sezione Analog Output*** *(anche in questo caso viene mostrata lo schema di una sola un'uscita, la O).*

Schema Elettrico -1-





In queste immagini vengono mostrate le restanti sezioni del circuito. In alto viene mostrata la ***sezione Digital Input*** *che, come è facile intuire, gestisce gli ingressi di tipo digitale (si noti che, per ogni ingresso, è stato utilizzato un fotoaccoppiatore; inoltre il* ***valore delle resistenze R37÷R44 deve essere determinato in funzione della massima tensione*** *che si prevede di applicare ai morsetti). A sinistra è invece mostrata la* ***sezione Digital Output****, che si occupa di gestire gli 8 relè che comandano le uscite di tipo digitale. Entrambe le sezioni utilizzano un integrato PCF8574A (chip U4 e U6) per gestire gli 8 Input/Output direttamente tramite una linea del tipo FCBus.*

ingressi digitali, presentate rispettivamente nel numero 76 e 77 della rivista; sulla stessa linea è inoltre possibile collegare un'*espansione di 8 uscite analogiche*, la cui presentazione verrà fatta nel prossimo numero.

L'esempio appena visto è chiaramente una semplice schematizzazione di applicazione domotica; il dispositivo è infatti munito di più ingressi o uscite per cui, utilizzando la stessa logica, sarà possibile comandare e controllare, oltre la caldaia, altri sistemi quali un condizionatore, un impianto di irrigazione, le luci interne o esterne, dei sensori di umidità, ecc. Infine, il caso di controllo domestico è soltanto una tra le tante possibili applicazioni in cui il dispositivo potrà essere utilizzato.

Finora abbiamo sempre fatto riferimento al caso in cui per la trasmissione venga utilizzata una connessione internet; teniamo però a sottolineare che il dispositivo può essere utilizzato anche mediante un collegamento RS232 realizzato direttamente alla porta seriale di un PC (in questo caso è però necessario munirsi di un convertitore dei formati TTL/RS232; per esempio l'*interfaccia seriale* che abbiamo presentato sul numero 75 della rivista). La selezione tra le due modalità avviene agendo su 4 dip-switch presenti nel circuito; facciamo però notare che se si seleziona la modalità **Seriale, è assolutamente importante che il modulo EM100 venga fisicamente rimosso dal circuito e che quest'ultimo venga staccato dalla rete LAN** (per maggiori dettagli vi rimandiamo al box "Selezione collegamento ethernet o seriale" presente in fondo all'articolo).

Dalla descrizione del dispositivo vista finora dovrebbe risultare abbastanza chiaro che il progetto risulta composto da due parti: una sezione elettronica il cui schema e la cui realizzazione pratica saranno analizzati in questa prima puntata e una sezione software (composta dal protocollo utilizzato dal circuito per comunicare e da un programma denominato *Controllo Casa* che realizza un completo esempio di gestione domotico di una abitazione) che invece sarà presentata nella prossima puntata insieme, come accennato in precedenza, all'espansione I²CBus delle uscite analogiche.

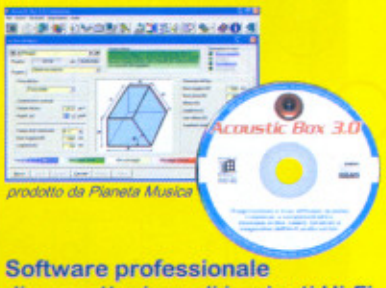
Schema elettrico

Passiamo, a questo punto, ad analizzare lo schema elettrico del dispositivo. In queste pagine sono mostrate diverse immagini, che mostrano le cinque sezioni che costituiscono il circuito completo.

Nella prima immagine è rappresentato lo schema generale: è possibile notare come il centro di tutto il dispositivo sia rappresentato dal microcontrollore PIC16F877 (chip U3). Infatti, da un lato (attraverso i propri pin numero 25÷28) questo si interfaccia direttamente al modulo ethernet EM100 (chip U2); dall'altro lato si interfaccia invece (pin 2÷10; 16÷19) con le restanti sezioni che realizzeranno gli ingressi e le uscite analogiche o digitali. Inoltre, attraverso la propria porta RC5 (pin 24), comanda il relè RL9 che realizza l'uscita di allarme. Riguardo a questa, vi facciamo notare un piccolo particolare: attraverso il jumper J1 è possibile selezionare se portare sulla morsettiera di uscita il connettore normalmente chiuso (NC) o normalmente aperto (NA) di RL9.

Infine, attraverso le porte RB0÷RB5 (pin 33÷38) del micro, viene realizzata la connessione I²CBus utilizzata, come già detto, per espandere il circuito con nuovi ingressi o uscite sia digitali che analogiche.

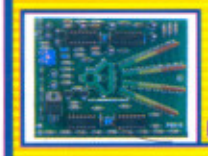
Tornando invece al modulo Tibbo EM100 (chip U2) vediamo che questo da un lato si interfaccia >

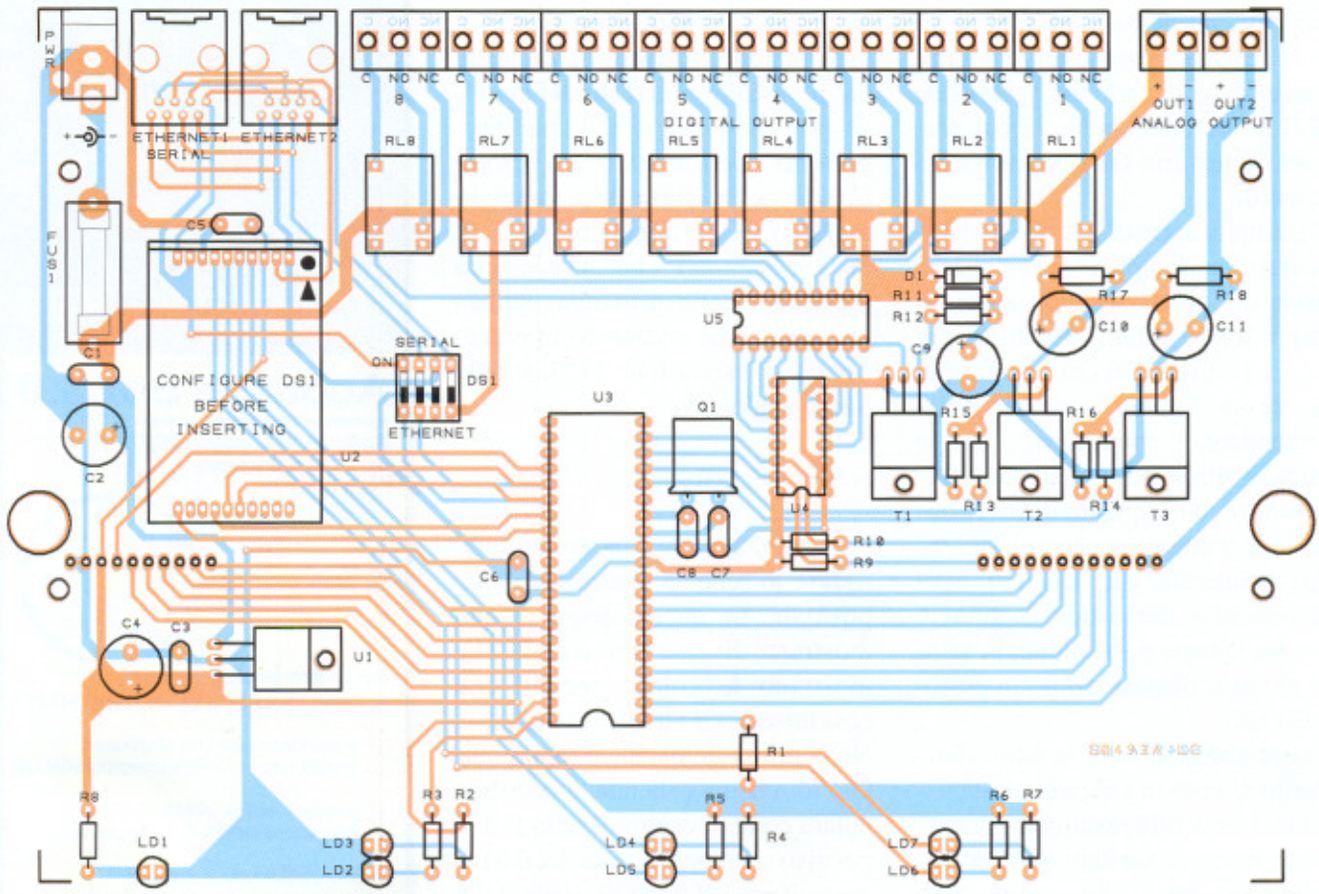

Acoustic Box 3.0


prodotto da Pianeta Musica
Software professionale
di progettazione di impianti Hi-Fi
progettazione e test di diffusori acustici, crossover e componentistica;
gestione ordini, clienti, fornitori e magazzino dell'Hi-Fi audio center;
completamente in italiano!

Circuiti stampati


Strumenti


Libri e CD-ROM

Kit di montaggio
www.pianetaelettronica.it



ELENCO COMPONENTI SCHEDA A:

R1: 4,7 KOhm
R2: 470 Ohm
R3: 470 Ohm
R4: 470 Ohm
R5: 470 Ohm
R6: 470 Ohm
R7: 470 Ohm
R8: 470 Ohm
R9: 4,7 KOhm
R10: 4,7 KOhm
R11: 47 KOhm
R12: 10 KOhm
R13: 5,6 KOhm
R14: 5,6 KOhm
R15: 100 KOhm
R16: 100 KOhm
R17: 470 Ohm 1W
R18: 470 Ohm 1W

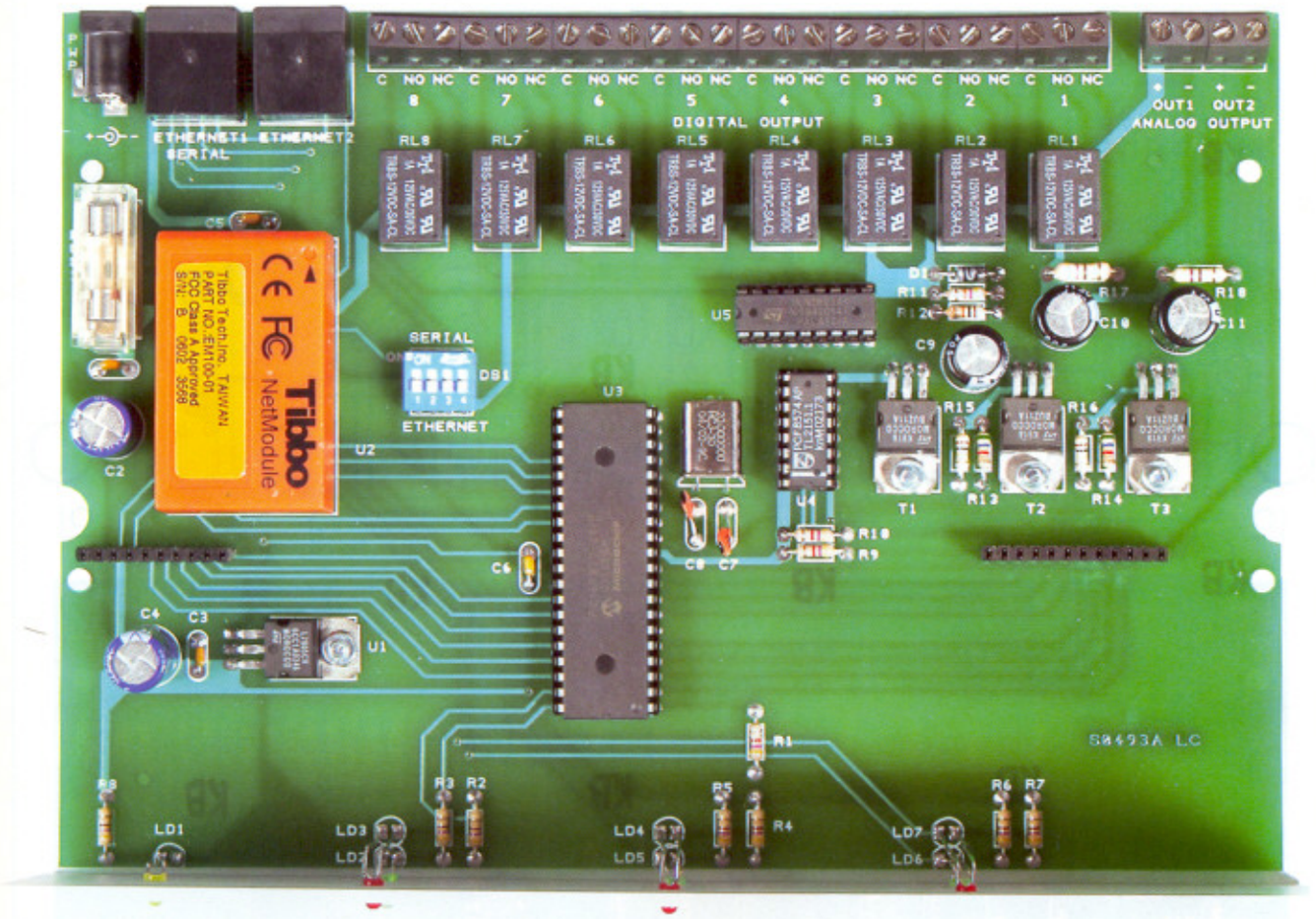
C1: 100 nF multistrato
C2: 470 µF 25VL elettrolitico
C3: 100 nF multistrato
C4: 470 µF 25VL elettrolitico
C5: 100 nF multistrato
C6: 100 nF multistrato
C7: 10 pF ceramico
C8: 10 pF ceramico
C9: 220 µF 35VL elettrolitico
C10: 220 µF 35VL elettrolitico
C11: 220 µF 35VL elettrolitico

D1: 1N4007

LD1: led 3mm giallo
LD2: led 3mm verde
LD3: led 3mm rosso
LD4: led 3mm rosso
LD5: led 3mm verde
LD6: led 3mm verde
LD7: led 3mm rosso

U1: 7805

direttamente alla porta ethernet RJ45, mentre dall'altro lato (come appena visto) al microcontrollore.
A questo punto è possibile comprendere come viene realizzata la selezione (realizzata tramite il dip-switch DS1) tra ingresso ethernet o seriale: infatti, osservando lo schema elettrico, si può notare che quando tutti i DIP sono a ON l'integrato U2, in un certo senso viene "bypassato" e i pin 2 e 4 della porta RJ45 (che trasportano i dati) vengono portati direttamente alle porte RC7 e RC6 del PIC. Dovrebbe anche risultare chiaro perché, nel caso si selezioni la **modalità seriale, è necessario togliere l'EM100 (chip U2) dal circuito e staccare quest'ultimo dalla LAN**. Infatti tramite DS1 vengono portati i +12V dell'alimentazione al pin 1 del connettore RJ45; questo **livello di tensione potrebbe danneggiare sia il modulo che le periferiche connesse alla rete locale**.
A questo punto passiamo ad analiz- >



U2: modulo Tibbo EM100
U3: PIC16F877 (MF493)
U4: PCF8574A
U5: ULN2803

T1÷T3: BUZ11

RL1: relè 12V
RL2: relè 12V
RL3: relè 12V
RL4: relè 12V
RL5: relè 12V
RL6: relè 12V
RL7: relè 12V
RL8: relè 12V

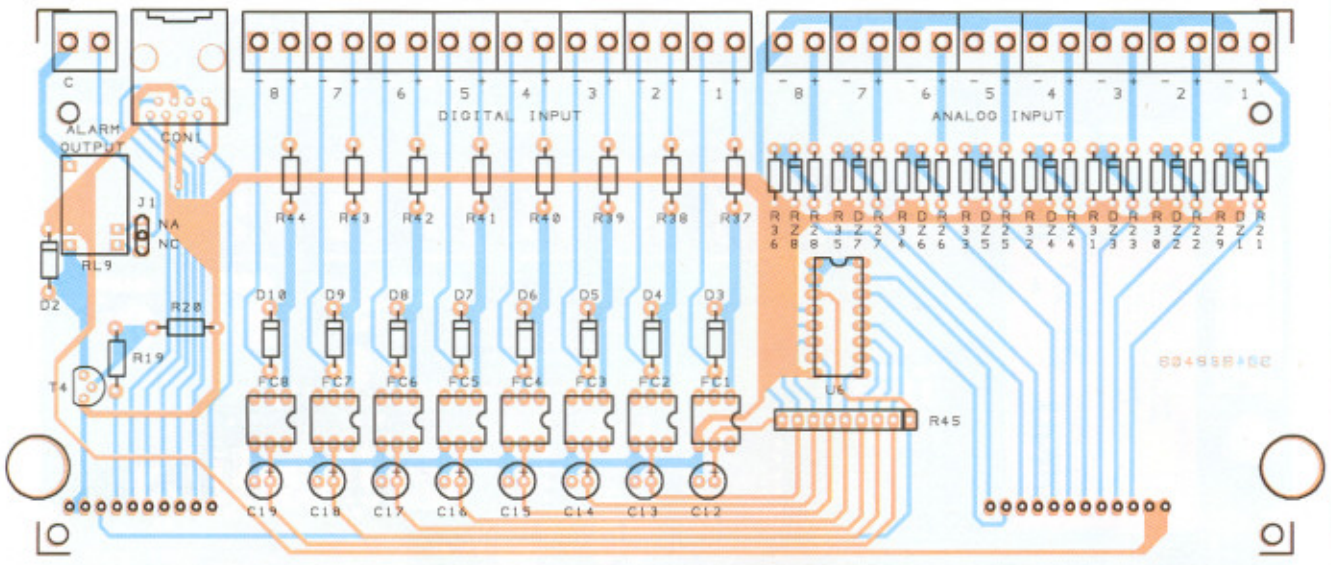
Q1: quarzo 20 MHz

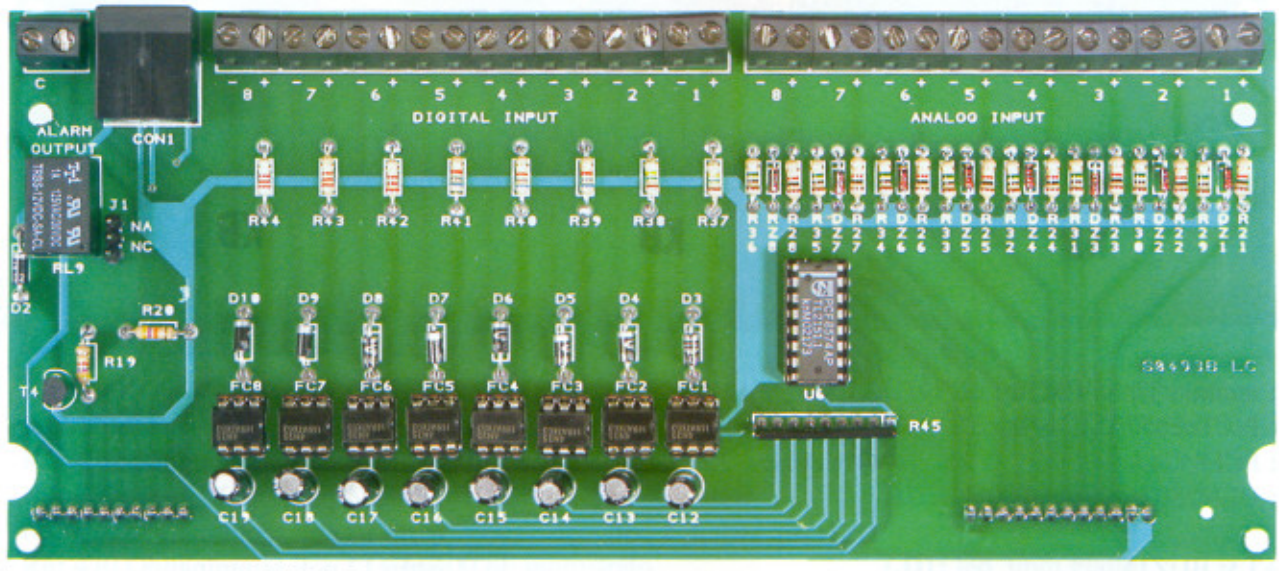
DS1: dip switch 4 poli

FUS1: fusibile 1A

Varie:
-morsetto 2 poli (2 pz.);
-morsetto 3 poli (8 pz.);
-connettore RJ45 (2 pz.);
-portafusibile da stampato;
-zoccolo 20+20 pin;
-zoccolo 9+9 pin;
-zoccolo 8+8 pin;
-strip 10 poli femmina passo 2 mm (2pz.);
-strip 10 poli femmina;
-strip 12 poli femmina;
-plug alimentazione;
-vite testa svasata 3 MA lunghezza 10 mm (4 pz.);
-dado 3 MA (4 pz.);
- circuito stampato cod. S0493A.

zare la sezione input/output analogica: la prima risulta composta da un semplice partitore (più uno zener di protezione) che preleva la tensione in ingresso (che, vi facciamo notare, deve essere compresa tra circa 0V e circa +5V) e la riporta, praticamente identica, ai pin 2÷10 del micro che, successivamente, tramite un ADC interno, la convertirà in digitale.
Le due uscite analogiche sono invece composte da un convertitore di onde PWM in livelli di tensione. Infatti il micro, via software, genererà ai propri pin 16 e 17 un'onda PWM di duty cycle proporzionale al livello desiderato di tensione in uscita; i Mosfet, i condensatori e le resistenze (T2, C10 e R17 per l'uscita O) convertiranno l'onda quadra in un livello di tensione (facciamo notare che il valore della tensione generata in uscita può essere compreso tra circa 0V e circa +12V).
Passiamo ora ad analizzare le sezio- >





ELENCO COMPONENTI SCHEDA B:

R19: 4,7 KOhm
R20: 47 KOhm
R21÷R28: 1 KOhm
R29÷R36: 1 MOhm
R37÷R44: 150 KOhm (220VAC)
6,8 KOhm (12VDC)
2,2 Kohm (5VDC)
R45: rete resistiva 100 KOhm

C12: 4,7 µF 100VL elettrolitico
C13: 4,7 µF 100VL elettrolitico
C14: 4,7 µF 100VL elettrolitico
C15: 4,7 µF 100VL elettrolitico
C16: 4,7 µF 100VL elettrolitico
C17: 4,7 µF 100VL elettrolitico
C18: 4,7 µF 100VL elettrolitico
C19: 4,7 µF 100VL elettrolitico

D2: 1N4007
D3: 1N4007
D4: 1N4007
D5: 1N4007
D6: 1N4007
D7: 1N4007
D8: 1N4007
D9: 1N4007
D10: 1N4007

ZD1÷ZD8: Zener 5,1V

U6: PCF8574A

FC1: 4N25
FC2: 4N25
FC3: 4N25
FC4: 4N25
FC5: 4N25
FC6: 4N25
FC7: 4N25
FC8: 4N25

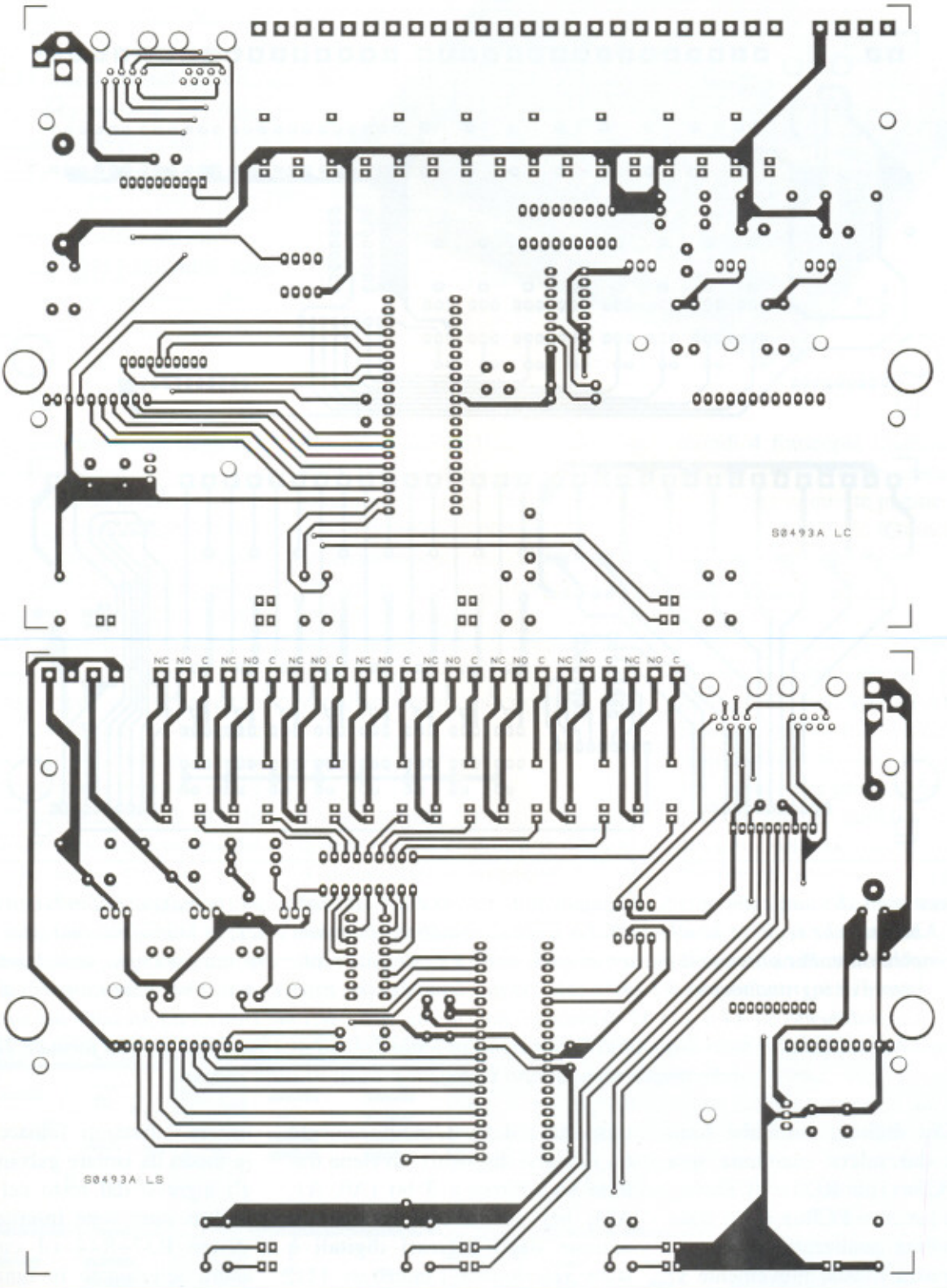
T4: BC547

RL9: relè 12V

Varie:
-morsetto 2 poli (17 pz.);
-connettore RJ45;
-zoccolo 8+8 pin;
-zoccolo 3+3 pin (8 pz.);
-strip 3 poli maschio;
-jumper;
-strip 10 poli maschio lungo;
-strip 12 poli maschio lungo;
-distanziale 20 mm (4 pz.);
-vite testa svasata 3 MA lunghezza 10 mm (4 pz.);
-dado 3 MA (4 pz.);
-circuito stampato cod. S0493B.

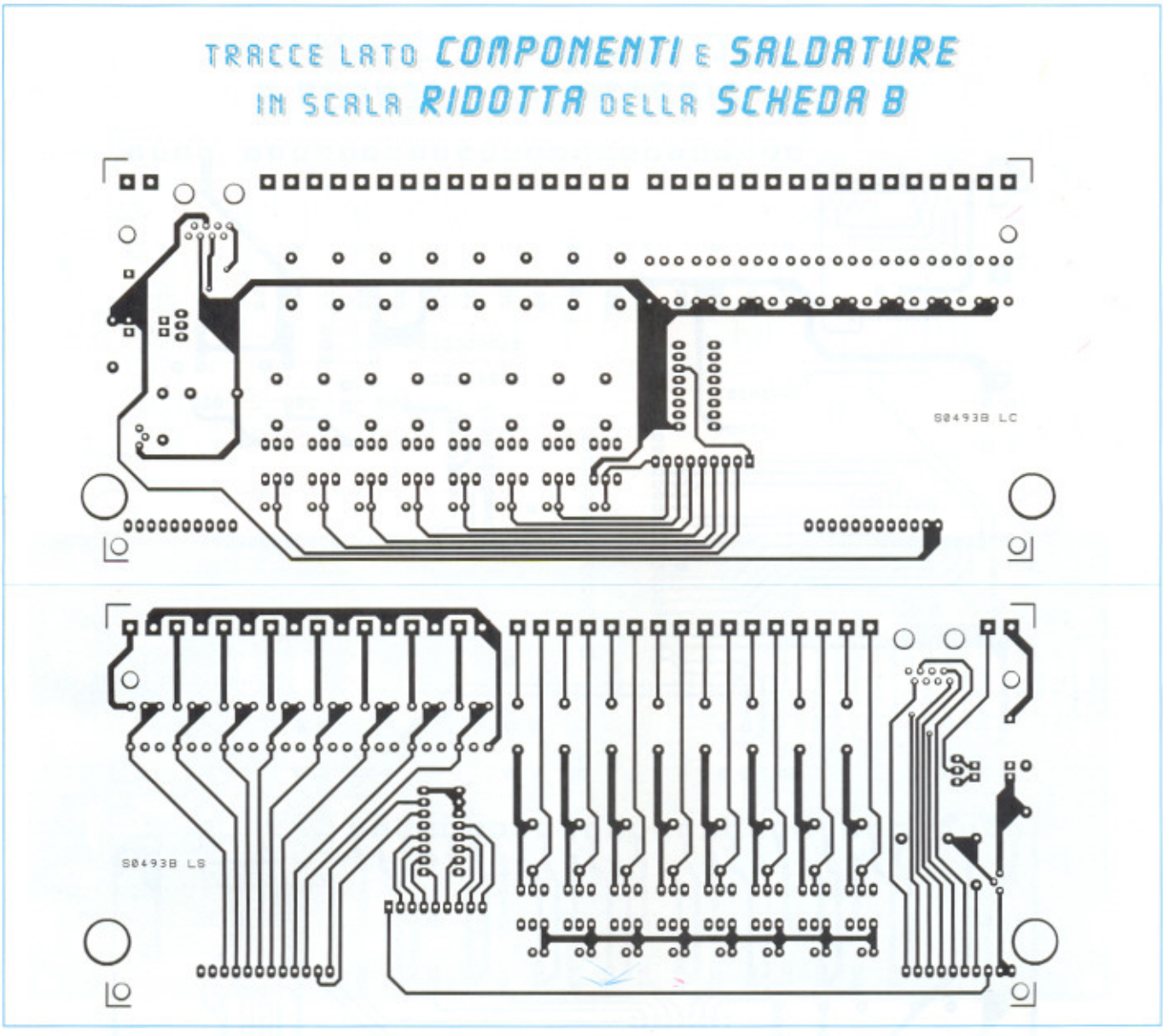
Vi ricordiamo che il valore assunto da R37÷R44 dipende dalla massima tensione che si vuole applicare ai singoli ingressi digitali.

TRACCE LATO **COMPONENTI** E **SALDATURE** IN SCALA **RIDOTTA** DELLA **SCHEDA A**



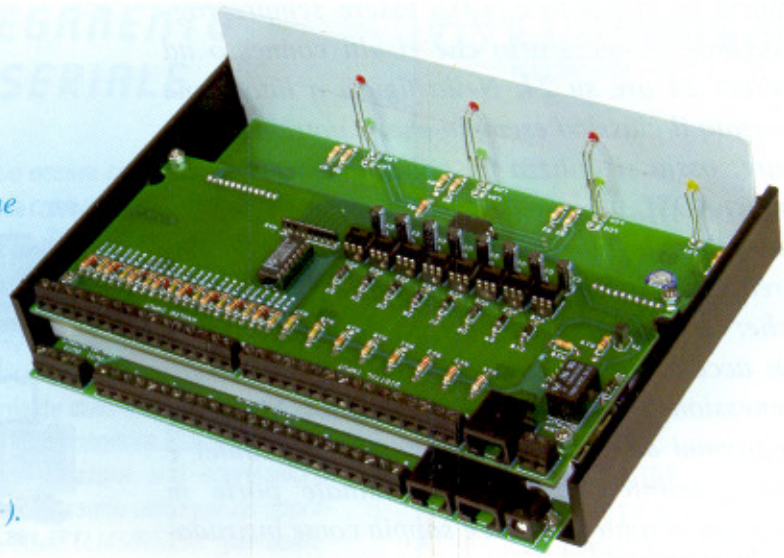
Tracce rame del lato componenti (immagine più in alto) e del lato saldature (immagine più in basso) della basetta che realizza la scheda A. Le immagini sono state rimpicciolite; per ottenere il master in scala reale è sufficiente realizzare una fotocopia utilizzando il fattore di ingrandimento del 141%.

TRACCE LATO COMPONENTI E SALDATURE IN SCALA RIDOTTA DELLA SCHEDA B



Tracce rame del lato componenti (immagine più in alto) e del lato saldature (immagine più in basso) della basetta che realizza la scheda B. La scala di visualizzazione non è 1:1; le immagini sono state rimpicciolite. Per ottenere il master in scala reale è sufficiente realizzare una fotocopia, utilizzando il fattore di ingrandimento del 141% (che, vi facciamo notare, rappresenta il normale rapporto di ingrandimento da A4 a A3). Vi ricordiamo inoltre che, nella sezione Download presente nel sito www.elettronicain.it, sono disponibili le immagini in scala 1:1, ad alta risoluzione ed in formato TIFF delle tracce rame di tutti i progetti presentati sulla rivista.

ni In/Out digitali; entrambe sono gestite dal micro mediante una linea I²CBus (pin RD0 e RC3).
Il collegamento I²CBus è già stato ampiamente analizzato all'interno della nostra rivista; brevemente vi ricordiamo che consente di gestire la comunicazione tra diverse periferiche utilizzando due soli pin (di cui SCL realizza il clock, mentre SDA trasporta i dati). L'indirizzamento tra i diversi dispositivi avviene tramite un address a 3 bit (A0, A1, A2); nel nostro circuito, per la sezione degli ingressi digitali è stato selezionato l'indirizzo 111, mentre per quella di uscita è stato scelto l'indirizzo 000.
Come si vede dallo schema, per gli 8 input digitali è stato scelto di utilizzare altrettanti fotoaccoppiatori in modo da isolare galvanicamente gli ingressi dal resto del circuito. Inoltre sono state inserite le resistenze R37÷R44 (il cui valore andrà selezionato in funzione del livello di tensione che si prevede di applicare in ingresso; riferitevi all'elenco presente nel piano di montaggio per i dettagli) ed i diodi >

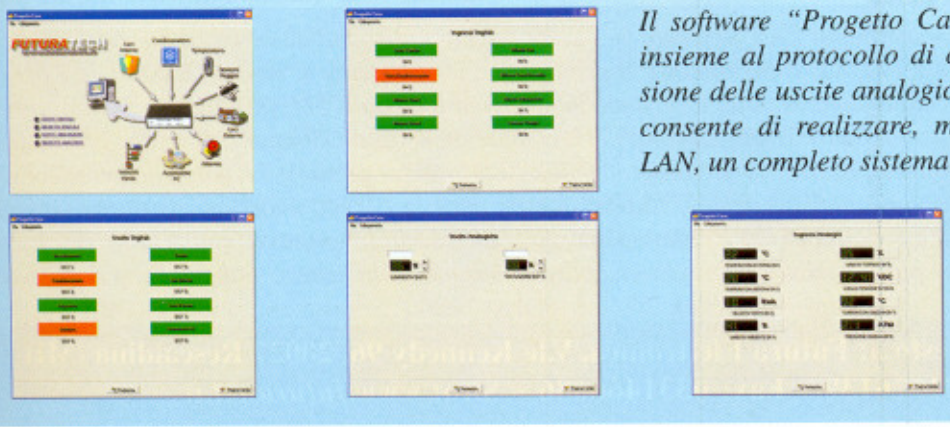


Un'immagine del dispositivo al termine del montaggio. La foto chiarisce come bisogna posizionare i 7 led presenti nel pannello anteriore del circuito e come vanno sovrapposte le due basette (il collegamento tra i due circuiti stampati è realizzato mediante due strip da 10 e 12 connettori metallici, le cui sezioni maschio e femmina sono rispettivamente montate sulla scheda superiore - B - e inferiore - A -).

D3÷D10 come protezione degli FC dalle tensioni inverse.
Per quanto riguarda invece le uscite digitali, queste comandano lo stato di 8 relè; è stato utilizzato un integrato ULN2803 (chip U5) in modo da convertire i livelli TTL forniti dal chip U4 ai livelli +12V necessari per poter comandare i relè.
Inoltre, per evitare che U5 si attivi prima di U4 (e più in generale prima del resto del circuito), è stata inserita la sezione composta da T1 e C9 sull'alimentazione di U5. In questo modo, dopo l'accensione del circuito, è necessario un certo tempo prima che C9 si carichi e che quindi l'integrato ULN2803 venga alimentato. Questo meccanismo di ritardo dell'attivazione è stato inserito in quanto, all'accensione, lo stato assunto dagli ingressi di U5 assumono valori "casuali" che quindi potrebbero portare ad attivare erroneamente dei relè. Invece, una volta che il transitorio di accensione del circuito è stato completato, gli ingressi di U5 vengono definiti dal microcontrollore e quindi verranno attivati solo i corretti relè.

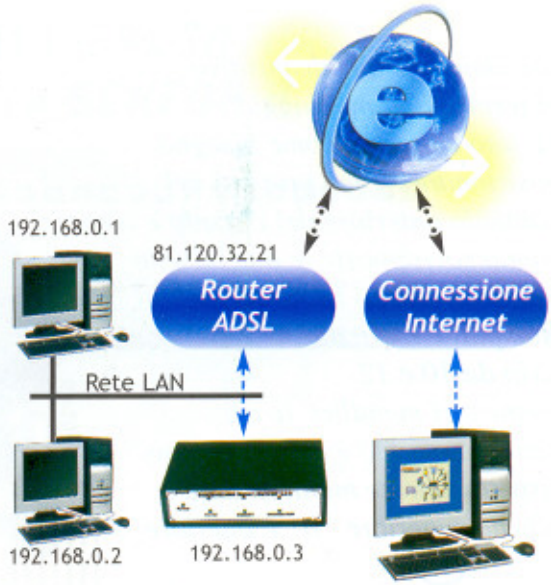
Realizzazione pratica

Come ultima analisi, vediamo insieme come costruire il circuito. Il dispositivo è costituito da due basette (scheda A e scheda B) entrambe doppia faccia che andranno collegate tra loro mediante 2 strip di 10 e 12 poli; realizzate quindi 4 fotocopie (ingrandite con fattore 141%) delle tracce rame pubblicate in queste pagine e utilizzate la tecnica della fotoincisione o il metodo PnP.
A questo punto iniziate l'operazione di saldatura dei diversi componenti, riferendovi sempre alle immagini pubblicate nel piano di montaggio per ogni dubbio che possa sorgervi relativamente alla realizzazione pratica. Per gli integrati sono stati previsti dei comodi zoccoli; iniziate quindi a saldare al circuito soltanto questi ultimi e inserite gli integrati solo a montaggio completato. Vi facciamo inoltre notare che, per la basetta A, tutti gli elementi andranno saldati sul lato componenti (per il modulo Tibbo >



Il software "Progetto Casa" (che verrà presentato, insieme al protocollo di comunicazione e all'espansione delle uscite analogiche, nella prossima puntata) consente di realizzare, mediante il Controllore I/O LAN, un completo sistema di gestione domotico di una abitazione. Il software è in grado di gestire collegamenti sia internet basati sul protocollo TCP/IP, sia seriali direttamente da un PC.

Affinché il dispositivo possa essere sempre raggiungibile, è necessario che risulti connesso ad internet 24 ore su 24. Nella figura a lato viene mostrato il classico esempio che si potrebbe verificare; ossia si utilizza un collegamento (tramite router) ADSL.



Generalmente, allo scopo di proteggere l'eventuale rete LAN, i router ADSL implementano due tecniche: la prima prevede che, di default, il router non accetti, su nessuna porta TCP, richieste di connessione provenienti dall'esterno. Mediante programmi di configurazione forniti coi router è però possibile "aprire" determinate porte in modo che il router accetti e sappia come instradare sulla LAN alcune connessioni esterne (per maggiori informazioni sull'operazione vi rimandiamo al manuale del vostro router ADSL).

La seconda tecnica prende invece il nome di NAT (Network Address Translation) e, senza entrare troppo nel dettaglio, consente di "nascondere" all'esterno la configurazione della rete locale. In pratica ciò significa che, per accedere da remoto ai dispositivi connessi alla LAN, non bisogna indirizzare l'IP locale (192.168.0.3 nell'esempio in figura) bensì l'IP esterno del router (81.120.32.21, assegnato dal provider ADSL); sarà poi il router che, in base alle proprie tabelle di instradamento, saprà come "raggiungere" il dispositivo.

EM100 sono state previste due strip femmina ciascuna da 10 poli); la basetta B prevede invece che le strip maschio a 10 e 12 poli vengano montate sul lato saldature.

Un'ultima nota riguarda il posizionamento dei 7 led: questi vanno posizionati tutti sulla scheda A; inoltre bisognerà lasciare i terminali metallici di opportune lunghezze (alcuni andranno addirittura allungati) in modo da posizionarli alla corretta altezza nel pannello anteriore dell'eventuale contenitore (osservate l'immagine presente in queste pagine che mostra nel dettaglio il posizionamento); nel pannello anteriore andranno anche praticati dei fori dal diametro di 3 mm in modo che i 7 led siano ben visibili all'esterno.

Terminate tutte le operazioni di saldatura, è possibile collegare le due basette: sovrapponete quindi la scheda B alla A e incastratele tra loro mediante le due strip a 10 e 12 poli.

A questo punto il montaggio risulta terminato; inserite le due basette nel contenitore esterno e fissatele allo stesso mediante alcune viti.

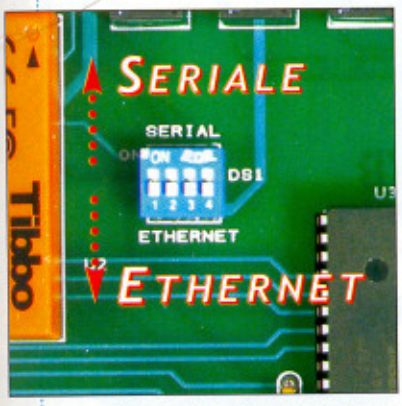
Nel caso che pensiate di utilizzare il dispositivo all'interno di una LAN, l'operazione successiva consiste nel programmare il modulo EM100 inserito nel circuito; colle- >

Per il MATERIALE

Il progetto del Controllore Input/Output LAN descritto in queste pagine è disponibile in kit in due versioni (cod. FT493KS, Euro 103,00 e cod. FT493KT, Euro 186,00): il primo viene fornito senza il modulo EM100 ed è pertanto utilizzabile solo mediante connessione seriale; il secondo implementa il modulo EM100 e può funzionare sia con connessione seriale che in rete ethernet TCP/IP. Entrambi i kit comprendono tutti i componenti, le due basette forate e serigrafate, il microcontrollore PIC16F877 già programmato, il software di gestione e il pannello anteriore serigrafato; il kit FT493KT comprende anche il modulo EM100. Il solo modulo EM100 è disponibile anche separatamente cod. EM100, Euro 95,00. Separatamente è inoltre disponibile il contenitore plastico (cod. 5100-NE229, Euro 9,00) di dimensioni opportune per il corretto inserimento delle due basette. Tutti i prezzi indicati sono da intendersi IVA inclusa.

Il materiale va richiesto a: Futura Elettronica, V.le Kennedy 96, 20027 Rescaldina (MI)
Tel: 0331576139 ~ Fax: 0331466686 ~ Sito: *www.futuranet.it*

SELEZIONE COLLEGAMENTO ETHERNET o SERIALE



Il circuito può essere collegato a un PC sia mediante una connessione Ethernet basata sul protocollo TCP/IP, sia mediante una connessione seriale RS232. In entrambi i casi viene utilizzato il jack ethernet/seriale in formato RJ45 presente nel circuito; quindi se si utilizza una connessione LAN il cavo proveniente dalla rete andrà collegato direttamente al connettore RJ45; se invece si sceglie di utilizzare una connessione seriale sarà necessario utilizzare un opportuno convertitore RJ45/RS232 (per esempio l'interfaccia seriale che è stata presentata, insieme al localizzatore GPS, nel numero 75 della rivista).

La selezione tra le due modalità avviene tramite i 4 dipswitch ***DS1*** *presenti nel circuito. Se questi sono impostati* ***TUTTI A OFF*** *il collegamento avviene mediante* ***RETE ETHERNET*** *(viene utilizzato il modulo Tibbo EM100); se invece i 4 dip sono impostati* ***TUTTI A ON*** *la connessione avviene mediante* ***LINEA SERIALE*** *(i dati in ingresso vengono "dirottati" direttamente al microcontrollore). Vi facciamo notare che* ***È ASSOLUTAMENTE IMPORTANTE CHE, NEL CASO IN CUI SI SELEZIONI LA MODALITÀ SERIALE, IL MODULO EM100 VENGA FISICAMENTE TOLTO DAL CIRCUITO E, PER NESSUN MOTIVO, IL DISPOSITIVO RISULTI COLLEGATO ALLA LAN,*** *altrimenti si rischia il danneggiamento sia dell'EM100 che di alcune periferiche collegate alla LAN. Pertanto vi consigliamo, durante il passaggio da un tipo di collegamento all'altro, di* ***SPEGNERE IL CIRCUITO, INSERIRE O TOGLIERE L'EM100, IMPOSTARE I 4 DIPSWITCH, COLLEGARE CORRETTAMENTE IL DISPOSITIVO ALLA LAN O ALLA PORTA SERIALE E RIACCENDERE IL DISPOSITIVO.***

gate quindi il circuito alla rete locale (**prima verificate di aver impostato TUTTI I 4 dipswitch DS1 a OFF**) e utilizzando il software *Connection Wizard* (scaricabile da intérnet dal sito *www.tibbo.com*) la cui presentazione è stata fatta sulla rivista numero 77, specificate l'indirizzo IP che il dispositivo dovrà assumere all'interno della LAN, il tipo di protocollo da utilizzare (TCP/IP), la porta TCP che vorrete utilizzare per la comunicazione, le impostazioni della porta seriale (baud rate 38.400 bps, nessuna parità, 8 bit di dati e nessun controllo di flusso) e la modalità di funzionamento Slave. A questo punto provate a collegarvi, utilizzando il software *Progetto Casa*, al circuito e verificate che lo stesso invii i dati letti o che modifichi le proprie uscite in base ai comandi inviati.

Se tutto funziona correttamente, potete collegare gli I/O del circuito ai sistemi esterni.

Se invece volete utilizzare il dispositivo direttamente via porta seriale, dovrete **rimuovere l'EM100 dalla scheda A, impostare a ON TUTTI i 4 DS1** e collegare il circuito alla porta seriale del PC mediante un convertitore TTL/RS232; il software *Progetto Casa* funziona anche con connessioni seriali RS232.

tommesani
dal 1980 ELETTRONICA, AUDIO & COMPUTER

Rivenditore autorizzato:

Futura Elettronica
Monacor Italia
NL Industries
GBC Store

RES
SIPE
CIARE
AUDAX
AUDIOTOP
DB Tecnologies
PHONOCAR
CORAL
PROEL
RCF

Importatore esclusivo
GAMMA-SCOUT®

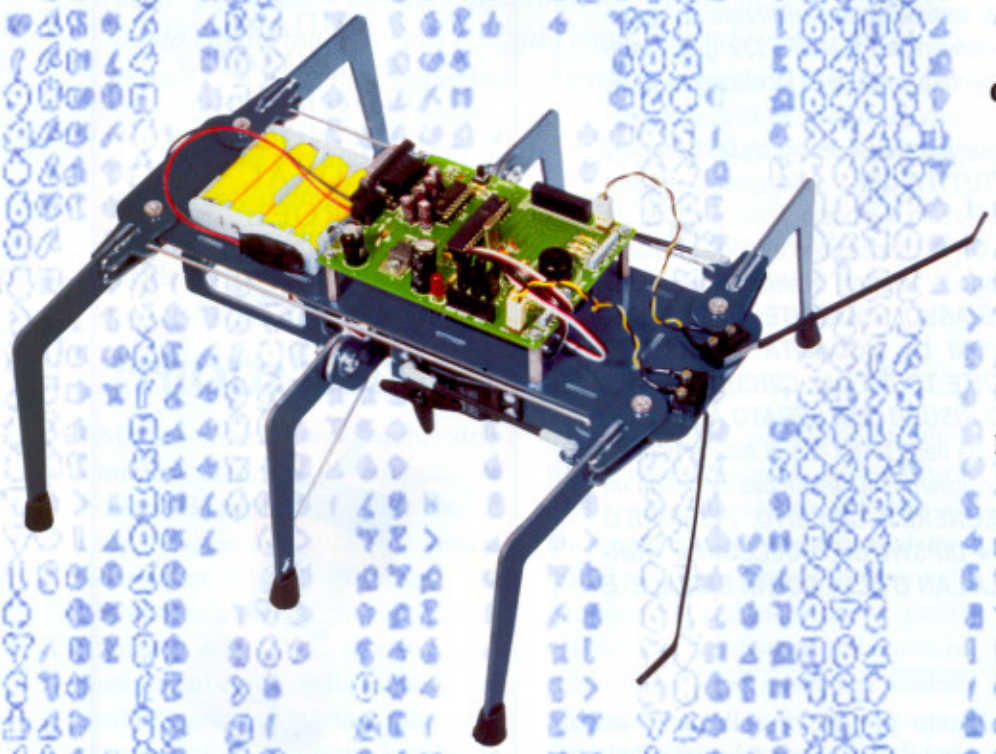


GEIGER COUNTER
Radiation detector
Made in Germany
Durata batterie 10 anni
Interfaccia PC

Trust
AUTHORISED TRUST POINT

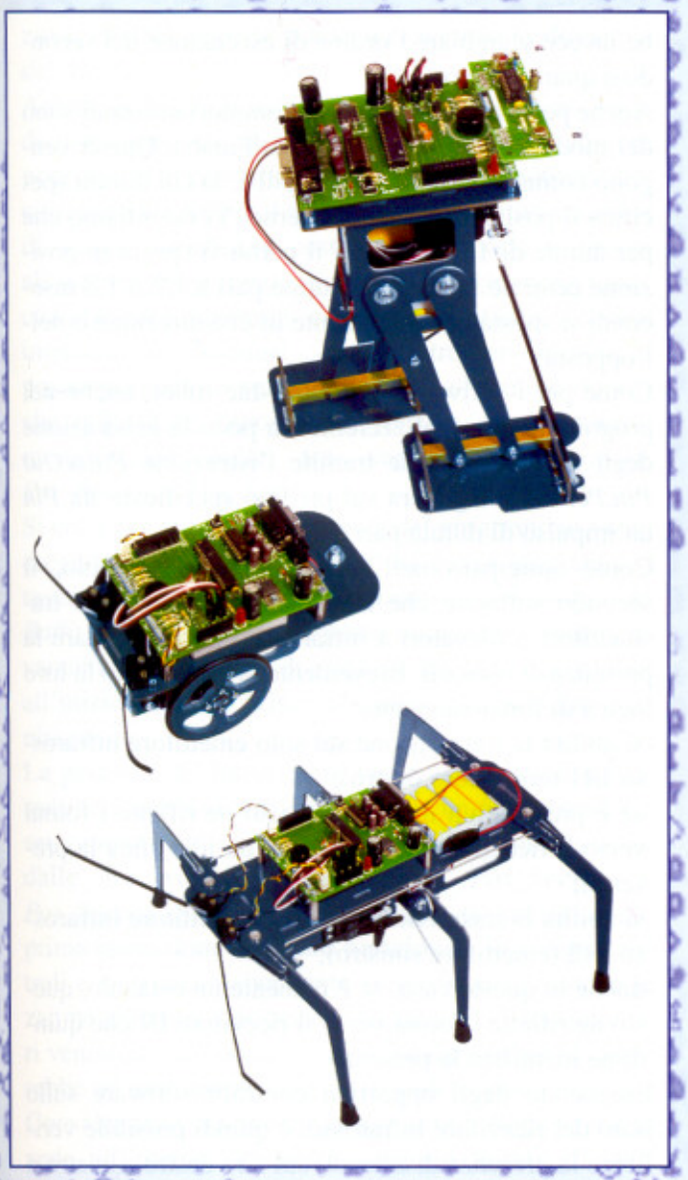
www.tommesani.it - Via San Pio V, 5A - 40131
Bologna - Tel.051-550761 - Fax 051-550591

COSTRUIRE E PROGRAMMARE I ROBOT: SPIDER



Terminiamo questo mese la descrizione del terzo robot: Spider, il ragno a sei zampe. Analizzeremo insieme tutti i segreti dei software che gli permettono di muoversi e riconoscere gli ostacoli.

In questa puntata studieremo insieme i programmi software che governano le periferiche e i dispositivi montati su Spider e che permettono quindi al robot di muoversi, riconoscere la presenza di eventuali ostacoli e rispondere correttamente agli eventi che gli provengono dal mondo esterno. Analizzeremo due esempi di listati, come sempre scritti utilizzando il linguaggio Basic: entrambi permetteranno a Spider di muoversi nello spazio circostante ma mentre il primo utilizzerà i baffi collegati agli microswitch per la rilevazione degli ostacoli, il secondo farà invece ricorso al kit dei rilevatori ad infrarosso (presentato all'interno dell'articolo della rivista numero 78) per individuare eventuali impedimenti al moto. Come per gli articoli relativi al software degli altri due robot, anche per Spider scopo del seguente testo non è tanto quello di mostrare tutto quello che il nostro robot è in grado di fare, ma semplicemente spiegare quali sono i concetti base che stanno dietro alla sua programmazione in modo da permettervi di realizzare in proprio le vostre applicazioni o ampliare con nuove funzionalità i listati qui presentati. A questo proposito vi ricordiamo che la motherboard presente sul robot è munita di due connettori di espansione (il primo SV1 composto da 20 pin porta all'esterno le 19 porte di I/O più la massa; il secondo JP13, JP14, JP15 composto da 8 pin mette a



di *Andrea Martini*

disposizione 3 masse, 3 livelli +5V e 2 livelli di tensione provenienti direttamente dalle batterie d'alimentazione), in modo da poter sovrapporre una scheda supplementare sulla quale aggiungere altri componenti o circuiti (per esempio dei sensori, delle telecamere, dei display LCD, ecc.).

Prima di addentrarci nello studio dei software, rivediamo brevemente alcune caratteristiche dell'hardware del robot: tutta la meccanica e l'elettronica viene gestita dalla motherboard in cui è installato un microcontrollore PIC16F876 funzionante alla frequenza di clock di 20 MHz. Questo dispone di tre porte di I/O (denominate A, B e C) per un totale di 22 pin; nel robot Spider alcuni di questi sono utilizzati per comandare i dispositivi esterni. Per esempio al pin 1 della porta A è collegato lo speaker; ai pin 2, 1 e 0 della porta B sono collegati i servomotori, rispettivamente di destra, sinistra e centrale; ai pin 4 e 5 della porta A sono collegati i baffi, rispettivamente di sinistra e di destra; ecc. Per un elenco completo e dettagliato vi invitiamo a consultare il box "Collegamento dei Dispositivi Esterni" presente in queste pagine.

A questo punto passiamo ad analizzare come viene generato il movimento del robot. Questo è ottenuto mediante tre servomotori: il primo comanda il movimento in avanti o verso l'indietro delle gambe anteriore e posteriore di destra; il secondo gestisce invece sempre il movimento in avanti o indietro delle gambe anteriore e posteriore, ma di sinistra; infine il terzo motore comanda l'abbassamento o il sollevamento di una delle gambe centrali. Per quest'ultimo caso vi facciamo notare che le due zampe centrali si muovono una in opposizione alla seconda; in pratica sono collegate mediante due aste di rimando in modo che quando una gamba viene alzata (per esempio la sinistra) l'altra viene abbassata (quella di destra), e viceversa.

L'avanzamento è quindi ottenuto nel seguente modo:

-si abbassa la gamba centrale di sinistra, in modo da sollevare le altre due gambe sinistre;
-si portano in avanti le gambe anteriore e posteriore di sinistra e indietro quelle di destra;
-si abbassa la gamba centrale di destra, in modo da sollevare le altre due gambe destre;
-si portano in avanti le gambe anteriore e posteriore di destra e indietro quelle di sinistra.

Eseguendo ciclicamente queste quattro operazioni si ottiene il movimento in avanti desiderato. Se invece si volesse fare retrocedere Spider, sostanzialmente le operazioni sarebbero le stesse; una volta alzate o abbassate le zampe centrali, bisognerebbe muovere in modo duale rispetto al caso appena visto le zampe anteriori e posteriori.

Chiaramente, per variare la direzione del proprio moto, il robot è anche in grado di compiere delle rotazioni rispetto al suo asse verticale; per esempio per eseguirle verso sinistra è necessario eseguire i seguenti passi:

-abbassare la gamba centrale di sinistra, in modo da sollevare le altre due gambe sinistre;
-mandare indietro le gambe anteriori e posteriori, sia di destra che di sinistra;
-abbassare la gamba centrale di destra, in modo da sollevare le altre due gambe destre;
-mandare avanti le gambe anteriori e posteriori, sia di destra che di sinistra.

Nel caso invece in cui si voglia ruotare verso destra, il primo e terzo punto rimarrebbero uguali; bisognereb-

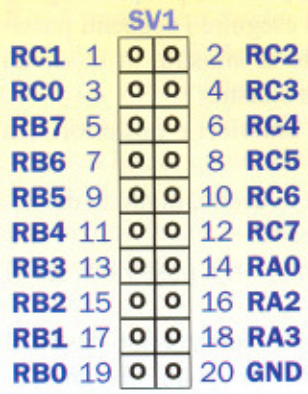
Collegamento dei Dispositivi Esterni

Tabella delle verità

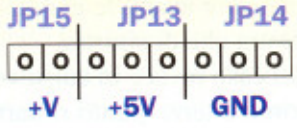
Pin	Porta	Stato logico	Significato
IR_1	Port C.1	1	Nessun ostacolo rilevato
IR_1	Port C.1	0	Ostacolo rilevato
IFR_1	Port C.5	1	Attivo trx IR di dx
IFR_1	Port C.5	0	Disattivo trx IR di dx
IFR_2	Port C.2	1	Attivo trx IR di sx
IFR_2	Port C.2	0	Disattivo trx IR di sx
Speaker	Port A.1	0	Nessun suono emesso
Speaker	Port A.1	1	Emissione di un suono
Servo1	Port B.2	-	Servomotore Destra
Servo2	Port B.1	-	Servomotore Sinistra
Servo3	Port B.0	-	Servomotore Centro
Baffo_1	Port A.4	0	Ostacolo rilevato a sx
Baffo_1	Port A.4	1	Nessun ostacolo a sx
Baffo_2	Port A.5	0	Ostacolo rilevato a dx
Baffo_2	Port A.5	1	Nessun ostacolo a dx
Led_1	Port C.3	0	Led di sinistra spento
Led_1	Port C.3	1	Led di sinistra acceso
Led_2	Port C.4	0	Led di destra spento
Led_2	Port C.4	1	Led di destra acceso

Variabile	Valore	Operazione eseguita
Servo1	600	Gambe di destra indietro
Servo1	750	Gambe di destra centrate
Servo1	900	Gambe di destra avanti
Servo2	600	Gambe di sinistra avanti
Servo2	750	Gambe di sinistra centrate
Servo2	900	Gambe di sinistra indietro
Servo3	600	Alza gamba centrale di sx
Servo3	750	Gambe centrali uguali
Servo3	900	Alza gamba centrale di dx

Connettori di Espansione



Per permettervi di espandere le funzionalità del robot, la motherboard che lo compone è munita di due connettori di espansione (SV1 e JP13, JP14, JP15); negli schemi qui pubblicati ne vengono mostrati i dettagli.



be invece scambiare l'ordine di esecuzione del secondo e quarto punto.

Anche per il robot Spider, i servomotori utilizzati sono dei modelli S3003 prodotti dalla Futaba. Questi vengono comandati da treni di impulsi, la cui durata specifica il posizionamento del perno. Vi ricordiamo che per durate di 1,5 msecondi il perno si sposta in posizione centrale mentre per durate pari a 1,2 o 1,8 msecondi si sposta completamente in una direzione o nell'opposta.

Come per i software degli altri due robot, anche nei programmi che analizzeremo tra poco la generazione degli impulsi avviene tramite l'istruzione *PulseOut Pin,Period* che genera sul piedino specificato da *Pin* un impulso di durata pari a $2 \cdot 10^{-6} \cdot Period$.

Come anticipato nell'introduzione dell'articolo, il secondo software che analizzeremo utilizzerà i trasmettitori e rilevatori a infrarosso per determinare la presenza di ostacoli. Brevemente vi ricordiamo la loro logica di funzionamento:

-si abilita la trasmissione sul solo emettitore infrarosso IR1 (emettitore destro);

-se è presente un ostacolo, questo ne riflette i fotoni verso il ricevitore IR che quindi ne identifica la presenza;

-si abilita la trasmissione sul solo emettitore infrarosso IR2 (emettitore sinistro);

-anche in questo caso, se è presente un ostacolo, questo ne riflette i fotoni verso il ricevitore IR che quindi ne identifica la presenza.

Eseguendo degli opportuni controlli software sullo stato del ricevitore infrarosso, è quindi possibile verificare la presenza di ostacoli sul lato destro, sinistro o su entrambi; quest'ultima condizione identifica la presenza di ostacoli davanti al robot.

SOFTWARE 1:

Movimento con controllo degli ostacoli tramite i baffi

Passiamo a questo punto a considerare il primo software. Come già accennato si tratta di un programma che permette a Spider di muoversi e identificare gli eventuali ostacoli tramite i baffi. Il ciclo principale gestisce l'avanzamento del robot e testa la presenza di impedimenti al moto; nel caso in cui venga rilevato un ostacolo sul lato destro, il robot retrocede di 3 passi, esegue una rotazione di circa 90 gradi verso sinistra e successivamente continua l'avanzamento.

Nel caso invece in cui l'ostacolo sia posizionato sulla sinistra, Spider retrocede ancora di 3 passi, ma la rotazione che esegue sarà verso destra.

Infine, se l'ostacolo viene rilevato da entrambi i baffi, dopo essere retrocessi di 3 passi, si esegue una rota-

zione di 180 gradi in modo da invertire la direzione del moto.

Analizziamo ora il listato: il programma inizia con la dichiarazione di alcune impostazioni di carattere generale, del verso delle porte (input o output); dei collegamenti tra pin delle porte e dispositivi esterni e di alcune variabili e costanti di comodo utilizzate all'interno del software. Successivamente inizia l'esecuzione vera e propria: come prima azione vengono impostati in posizione centrale i tre servomotori; il posizionamento è realizzato dalla subroutine *Azzera* che utilizza le impostazioni relative ai servomotori memorizzate nelle tre variabili *Pos_Servo1*, *Pos_Servo2* e *Pos_Servo3*.

Si entra poi nel ciclo principale del programma: viene eseguito il primo movimento del passo in avanti; si testa la presenza di ostacoli richiamando la subroutine *Baffi*; viene completato il passo in avanti; si testa nuovamente la presenza di ostacoli ed infine si ritorna all'inizio del ciclo principale, che verrà eseguito infinitamente.

La gestione del movimento viene eseguita dalla subroutine *Cammina* che non fa altro che posizionare i 3 servomotori nelle posizioni che le vengono indicate dalle tre variabili *Pos_Servo1*, *Pos_Servo2* e *Pos_Servo3*. Si noti che, all'interno di questa, come prima operazione vengono posizionate le zampe centrali e solo successivamente vengono mosse anche le zampe anteriori e posteriori. Se invece i 3 servomotori venissero posizionati contemporaneamente, si otterrebbe un movimento non corretto del robot.

Consideriamo ora invece la subroutine *Baffi*: come si vede all'interno di questa vengono testati gli stati assunti dai due dipswitch (vi ricordiamo che la condizione *Baffo=0* indica che il relativo dipswitch è chiuso, ed è presente quindi un ostacolo), ed eventualmente si richiamano le subroutine etichettate con *Sinistra*, *Destra* e *Inverti*. Nel caso in cui venga richiamata la prima di queste, significa che è presente un ostacolo sul lato sinistro: bisogna quindi retrocedere di alcuni passi (3 nel nostro esempio) e girare a destra. Il primo punto è eseguito richiamando la subroutine *Indietro*; la rotazione è invece ottenuta all'interno di un ciclo *for* in cui vengono impostate le posizioni dei 3 servomotori secondo la logica spiegata nell'introduzione dell'articolo.

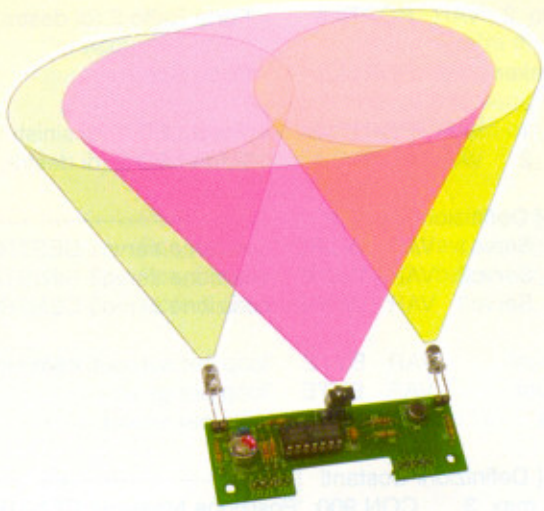
La subroutine *Destra* lavora in modo duale rispetto alla *Sinistra* appena vista: dopo essere retrocessi di 3 passi (gestito ancora dalla subroutine *Indietro*) si esegue una rotazione verso sinistra; rotazione eseguita all'interno di un ciclo *for* che lavora in modo del tutto simile al precedente.

L'ultimo caso (subroutine *Inverti*) gestisce invece la possibilità in cui vengano rilevati ostacoli contemporaneamente sia sul lato destro che sinistro: quindi, dopo essere retrocessi di 3 passi (si richiama la solita *Indietro*), si esegue una rotazione di 180°. La subroutine lavora come la procedura *Destra*; in questo caso però il ciclo *for* viene eseguito un numero maggiore di volte, in modo da eseguire una rotazione più ampia.

Prima di concludere l'analisi, facciamo due piccole note. La prima riguarda, all'interno del listato, la presenza di due variabili *Led_1* e *Led_2*; queste sono utilizzate per accendere o spegnere due led, montati rispettivamente sul lato sinistro e destro, utilizzati come "occhi" del robot. La seconda riguarda invece la

Trasmettitori e ricevitore ad infrarosso

La figura mostra la logica del funzionamento del rilevatore ad infrarosso. I due trasmettitori IR (destro e sinistro) inviano, alternativamente, il segnale luminoso: se è presente un ostacolo, il cono di luce viene riflesso e quindi rilevato dal ricevitore posto in posizione centrale. Andando a leggere lo stato del ricevitore è quindi possibile sia rilevare la presenza di eventuali ostacoli, sia distinguere se questi si trovano sul lato destro, su quello sinistro oppure se sono posizionati di fronte al robot.



Movimento con controllo degli ostacoli tramite i baffi

```
'*****************************************************************
'* Nome:        Spider_Baffi                                     *
'* Note:        Movimento con controllo sui baffi                *
'*****************************************************************
'-----[ Definizioni          ]-----------------------------------
DEFINE LOADER_USED 1        'Usato per boot-loader
DEFINE OSC 20               'Imposta Clock a 20MHz
ADCON1 = %00000111          'Port A = Digitale

'-----[ Verso Porte          ]-----------------------------------
TRISA = %00110000           'Imposta pin Port A in
                            'Input e/o Output
TRISB = %00000000           'Imposta pin Port B in
                            'Input e/o Output
TRISC = %00100100           'Imposta pin Port C in
                            'Input e/o Output

'-----[ Definizioni I/O      ]-----------------------------------
Servo1  VAR  PORTB.2        'Porta Servo 1 (DESTRA)
Servo2  VAR  PORTB.1        'Porta Servo 2 (SINISTRA)
Servo3  VAR  PORTB.0        'Porta Servo 3 (CENTRO)

Baffo_1 VAR  PORTA.4        'Porta Baffo 1 (di sinistra)
Baffo_2 VAR  PORTA.5        'Porta Baffo 2 (di destra)

Speaker VAR  PORTA.1        'Porta Speaker

Led_1   VAR  PORTC.3        'Porta LED 1 (di sinistra)
Led_2   VAR  PORTC.4        'Porta LED 2 (di destra)

'-----[ Definizioni Variabili ]-------------------------------
Pos_Servo1  VAR  WORD  'posizione servo1 DESTRA
Pos_Servo2  VAR  WORD  'posizione servo2 SINISTRA
Pos_Servo3  VAR  WORD  'posizione servo3 CENTRO

mcount      VAR  BYTE  'loop per subrout. cammina
ncount      VAR  BYTE  'loop per gli altri
Nota        VAR  BYTE  'nota per sound

'-----[ Definizioni Costanti ]-------------------------------
pos_max_3   CON 900  'Posizione Massima CENTRO
pos_mid_3   CON 750  'Posizione Centrale CENTRO
pos_min_3   CON 600  'Posizione Minima CENTRO
pos_max_1   CON 900  'Posizione Massima DESTRA
pos_mid_1   CON 750  'Posizione Centrale DESTRA
pos_min_1   CON 600  'Posizione Minima DESTRA
pos_max_2   CON 900  'Posizione Massima SINISTRA
pos_mid_2   CON 750  'Posizione Centrale SINISTRA
pos_min_2   CON 600  'Posizione Minima SINISTRA

Ritardo     CON 20   'Pausa per Pulsout in µsecondi
Passo       CON 5    'Passo per ciclo Azzera
Passo_1     CON 7    'Passo per ciclo Camminata
Durata      CON 50   'Durata nota per Sound

'-----[ Inizializzazione     ]-----------------------------------
PORTA =0
PORTB =0
PORTC =0

'-----[ Inizio programma      ]-----------------------------------
      ncount = 0
      mcount = 0
      Nota = 0

      GoSub Occhi           'lampeggio occhi

      Nota = 50
      GoSub Suono           'Emetti suono

      Pos_Servo1 = pos_mid_1 'posizione inizale...
      Pos_Servo2 = pos_mid_2 '...dei tre servomotori...
      Pos_Servo3 = pos_mid_3 '...al centro
      GoSub Azzera

Inizio:

      Pos_Servo1 = pos_min_1 'dietro zampe destra
      Pos_Servo2 = pos_min_2 'avanti zampe sinistra
      Pos_Servo3 = pos_max_3 'basso centrale sinistra
      GoSub Cammina          'movimento in avanti

      GoSub Baffi            'test se contatto su baffi

      Pos_Servo1 = pos_max_1 'avanti zampe destra
      Pos_Servo2 = pos_max_2 'dietro zampe sinistra
      Pos_Servo3 = pos_min_3 'basso centrale destra
      GoSub Cammina          'movimento in avanti

      GoSub Baffi            'test se contatto su baffi

      GoTo Inizio            'ripeti ciclo

'-----[ Subroutine           ]-----------------------------------
Cammina:

      'Posiziona le zampe centrali
      For mcount=1 to 100 step Passo_1
            PulsOut Servo3,Pos_Servo3
            Pause Ritardo
      Next

      'Muove le zampe di destra e sinistra
      For mcount=1 to 100 step Passo_1
            PulsOut Servo1,Pos_Servo1  'destra
            PulsOut Servo2,Pos_Servo2  'sinistra
            PulsOut Servo3,Pos_Servo3  'centro
            Pause Ritardo
      Next

      Return

Baffi:
      'Testa presenza di ostacoli sui baffi
      IF Baffo_1 = 0 Then Sinistra 'contatto su baffo 1
      IF Baffo_2 = 0 Then Destra  'contatto su baffo 2
      IF Baffo_1 AND Baffo_2 = 0 Then Inverti  'contatto
                                    'su entrambi i baffi
      Return
```

```
Occhi:
        Led_1 = 0               'Spegne gli occhi
        Led_2 = 0

        For ncount=1 to 5       '5 Lampeggi degli occhi
                Toggle Led_1
                Toggle Led_2
                Pause 100
        Next

        Return

Suono:
        Sound Speaker, [Nota,Durata]    'Emette suono
        Low Speaker             'Spegne speaker
        Return

Azzera:
        'Posiziona i tre servomotori in posizione centrale
        For ncount = 1 to 100 step Passo
                PulsOut Servo1,Pos_Servo1
                PulsOut Servo2,Pos_Servo2
                PulsOut Servo3,Pos_Servo3
                Pause Ritardo
        Next
        Return

Sinistra:
'Bisogna tornare indietro di tre passi e girare a destra...
'...per evitare l'ostacolo individuato dal baffo_1

        Nota = 10               'Emetti suono
        GoSub Suono

        Led_1 = 0               'Spegni led di sinistra
        GoSub Indietro          'Torna indietro di 3 passi

        For ncount=1 to 5                       'Gira a destra
                Pos_Servo1 = pos_max_1 'av. zampe dx
                Pos_Servo2 = pos_min_2  'av. zampe sx
                Pos_Servo3 = pos_max_3 'basso cent. sx
                GoSub Cammina

                Pos_Servo1 = pos_min_1  'ind. zampe dx
                Pos_Servo2 = pos_max_2 'ind. zampe sx
                Pos_Servo3 = pos_min_3  'basso cent. dx
                GoSub Cammina
        Next

        Led_1 = 1               'Riaccendi led di sinistra
        Return

Destra:
'Bisogna tornare indietro di tre passi e girare a sinistra...
'...per evitare l'ostacolo individuato dal baffo_2

        Nota = 100              'Emetti suono
        GoSub Suono

        Led_2 = 0               'Spegni led di destra
        GoSub Indietro          'Torna indietro di 3 passi

        For ncount=1 to 5                       'Gira a sinistra
                Pos_Servo1 = pos_min_1  'ind. zampe dx
                Pos_Servo2 = pos_max_2 'ind. zampe sx
                Pos_Servo3 = pos_max_3 'basso cent. sx
                GoSub Cammina

                Pos_Servo1 = pos_max_1 'av. zampe dx
                Pos_Servo2 = pos_min_2  'av. zampe sx
                Pos_Servo3 = pos_min_3  'basso cent. dx
                GoSub Cammina

        Next

        Led_2 = 1               'Riaccendi led di destra
        Return

Inverti:
'Bisogna tornare indietro di tre passi e girare a 180°...
'...per evitare l'ostacolo individuato da entrabi i baffi

        Nota = 50               'Emetti suono
        GoSub Suono

        Led_1 = 0               'Spegni entrambi i led
        Led_2 = 0
        GoSub Indietro          'Torna indietro di 3 passi

        For ncount=1 to 7               'Gira a sinistra di 180°
                Pos_Servo1 = pos_min_1  'ind. zampe dx
                Pos_Servo2 = pos_max_2 'ind. zampe sx
                Pos_Servo3 = pos_max_3 'basso cent. sx
                GoSub Cammina

                Pos_Servo1 = pos_max_1 'av. zampe dx
                Pos_Servo2 = pos_min_2  'av. zampe sx
                Pos_Servo3 = pos_min_3  'basso cent. dx
                GoSub Cammina

        Next

        Led_1 = 1               'Riaccendi entrambi i led
        Led_2 = 1
        Return

Indietro:
        For ncount=1 to 4               'Torna indietro di 3 passi
                Pos_Servo1 = pos_max_1 'av. zampe dx
                Pos_Servo2 = pos_max_2 'ind. zampe sx
                Pos_Servo3 = pos_max_3 'basso cent. sx
                GoSub Cammina

                Pos_Servo1 = pos_min_1  'ind. zampe dx
                Pos_Servo2 = pos_min_2  'av. zampe sx
                Pos_Servo3 = pos_min_3  'basso cent. dx
                GoSub Cammina

        Next

        Return

End
```

Il Bootloader

Per far funzionare tutta la serie dei nostri robot è necessario scrivere un programma che gli faccia fare ciò che desideriamo, nel limite delle risorse disponibili. Questo programma può essere scritto in qualsiasi linguaggio, dal Basic al C, all'assembler, ecc.; esso deve essere successivamente compilato in modo da ottenere il file .HEX adatto ad essere memorizzato nel micro. In questo box vi spieghiamo come questo file può essere caricato nel microcontrollore.

Normalmente questa operazione si effettua utilizzando un idoneo programmatore hardware nel quale viene fisicamente inserito il micro. Tuttavia, per rendere più agevole questa operazione, la nostra motherboard prevede un sistema di programmazione in-circuit che consente di non rimuovere il microcontrollore dalla basetta.

In pratica la programmazione avviene direttamente dal PC tramite la porta seriale connessa alla presa DB9 presente sulla motherboard. Per fare ciò è necessario utilizzare un particolare sistema di programmazione denominato Bootloader.

Questo sistema prevede l'utilizzo di uno specifico software (PICdownloader.exe) che permette di caricare nel micro i programmi da noi sviluppati (in formato .HEX) tramite la porta seriale. Ciò è possibile unicamente se nei microcontrollori è stato in precedenza caricato un breve programma di supporto (bootldr20Mhz-19200bps.hex) che viene allocato nelle prime celle di memoria; questo software, ovviamente, deve essere caricato con un normale programmatore. Comunque, nelle scatole di montaggio dei robot fornite, tutti i microcontrollori sono già programmati con questo firmware per cui, coloro che acquistano i kit potranno caricare i file .HEX sviluppati senza dover utilizzare alcun programmatore esterno.

A questo punto vediamo come utilizzare il Bootloader. Innanzitutto è necessario scaricare il programma PICdownloader.exe dal sito www.futuranet.it; successivamente tale programma deve essere installato sul vostro PC.

A questo punto, tramite un opportuno compilatore, generate il file in formato .HEX del vostro programma e, grazie al PICdownloader, scaricatelo nella EEPROM del PIC presente sulla motherboard.

Per maggiori informazioni sulle operazioni da eseguire vi rimandiamo all'articolo relativo ai robot presente nella rivista numero 76. All'interno del box "Come caricare i programmi" vengono mostrati nel dettaglio tutti i passi necessari per l'installazione del programma sul PC e il caricamento dei file .HEX all'interno del microcontrollore.

subroutine *Suono* e le relative chiamate posizionate all'interno del listato. Questa procedura, che è già stata ampiamente analizzata nelle puntate riguardanti il software sia di CarBot che di Filippo, altro non fa che emettere un segnale sonoro dallo speaker installato sulla motherboard. L'emissione del suono viene gestita dall'istruzione *Sound Speaker, [Nota,Durata]* che genera sul pin specificato da *Speaker* un'onda quadra caratterizzata dai livelli di tensione TTL. Il parametro *Nota* indica la frequenza del tono sonoro generato, mentre *Durata* indica per quanto tempo l'emissione verrà generata (il parametro *Durata* può assumere valori compresi nell'intervallo 0÷255, mentre il tempo di generazione è circa pari a *Durata·12millisecondi*). Sebbene all'interno dei listati non sia necessario, vi facciamo notare che è possibile generare più note utilizzando una singola istruzione: per esempio l'istruzione *Sound Speaker [Nota1, Durata1, Nota2, Durata2, ..., NotaN, DurataN]* genera *N* segnali sonori di frequenza e durate specificate; grazie a una sola istruzione è quindi possibile generare segnali sonori multitono.

La subroutine *Suono* termina con l'istruzione *Low Speaker* che altro non fa che portare a livello logico basso il pin della porta specificata da *Speaker*.

SOFTWARE 2:

Movimento con controllo degli ostacoli tramite infrarossi

Passiamo, a questo punto, a considerare il secondo software; si tratta anch'esso di un programma che permette a Spider di muoversi all'interno di uno spazio, avanzare, retrocedere o compiere rotazioni. La differenza rispetto all'esempio precedente è che in questo caso la rilevazione degli ostacoli non è più effettuata mediante i dipswitch collegati ai baffi; bensì si utilizza il kit composto dai trasmettitori e ricevitore a infrarosso presentato nel numero 78 della rivista.

Analizziamo quindi, passo dopo passo, il listato: come sempre si inizia con la definizione di alcune impostazioni generali necessarie per la corretta esecuzione del programma, del verso dei pin delle tre porte, dei collegamenti tra porte del microcontrollore e dispositivi esterni (servomotori, speaker, trasmettitori e ricevitore ad infrarosso, ecc.). Successivamente vengono definite e inizializzate alcune variabili e costanti utilizzate all'interno del programma.

A questo punto si entra nel ciclo principale: inizialmente si centrano tutti i servomotori (subroutine *Azzera*); successivamente si eseguono i test sugli infrarossi. Il funzionamento è simile al software che abbiamo visto nella puntata dedicata a Filippo: si atti-

va il trasmettitore di sinistra e si memorizza all'interno della variabile *Sinist_IR_det* lo stato del ricevitore infrarosso. In seguito si attiva il trasmettitore di destra e si memorizza in *Destr_IR_det* lo stato del ricevitore (vi ricordiamo che la presenza di ostacoli su un singolo lato viene indicata con *Sinist_IR_det* o *Destr_IR_det* uguali a 0).

A questo punto, sulla base dello stato delle due variabili si determina quale subroutine eseguire (la selezione viene scritta nella variabile *action*); se non viene individuato nessun ostacolo *action* viene posta uguale a 0; se è presente un ostacolo sul solo lato sinistro viene posta uguale a 1; se è presente solo sul lato destro è pari a 2 e infine se l'ostacolo è di fronte viene imposto *action=3*.

A questo punto i due bit meno significativi di *action* vengono scritti nella variabile *VarAnd* (utilizzando un'operazione di *And*, rappresentata dal simbolo &).

L'istruzione che determina il passaggio alla corretta subroutine è la *Branch VarAnd, [Avanti, Destra, Sinistra, Inverti]*; l'avevamo già incontrata nella puntata dedicata al software di Filippo. Riassumendo brevemente il suo funzionamento, possiamo dire che se *VarAnd=0* si salta a *Avanti*; se *VarAnd=1* si esegue la subroutine *Destra* e così via. Si noti che i salti sono equivalenti a delle istruzioni *GoTo* (e non *Gosub*): per questo le 4 subroutine *Avanti*, *Destra*, *Sinistra* e *Inverti* ritornano nel ciclo principale (etichettato con la label *Inizio*) mediante l'istruzione *GoTo Inizio* e non con una *Return*.

Per quanto riguarda queste 4 subroutine, praticamente sono le stesse dell'esempio appena visto: *Avanti* gestisce l'avanzamento di un passo (richiamando la *Cammina*), sia delle gambe di destra che di sinistra.

La subroutine *Sinistra* viene richiamata quando è presente un ostacolo sulla destra; tramite *Indietro* si retrocede quindi di 3 passi e tramite il ciclo *for* si esegue una rotazione verso destra.

La subroutine *Destra* è la duale della *Sinistra*: dopo essere retrocessi si esegue una rotazione verso sinistra.

Infine *Inverti* richiama *Indietro* per retrocedere di 3 passi ed inverte il verso del moto eseguendo una rotazione di 180 gradi verso sinistra.

Conclusioni:

Con questa puntata termina la rassegna di Elettronica In dedicata alla robotica.

Se ci avete seguito con attenzione per tutti questi mesi sicuramente ora sarete in grado di costruire e conoscerete tutti i segreti (sia hardware che software) dei nostri tre robot.

Non solo; giunti a questo punto del corso ci sembra di >

- Utilizzo pratico -

Come visto nel testo dell'articolo, i 3 servomotori vengono comandati mediante il parametro Period della funzione PulsOut. Abbiamo già spiegato che per valori di Period pari a 900 o a 600 il perno si sposta tutto in un senso o nell'altro; mentre per Period pari a 750 il perno si sposta in posizione centrale. Questi valori sono stati calcolati teoricamente partendo dalla formula T=2·10-6·Period; sostituendo T con le durate desiderate si ottengono i relativi valori di Period. In realtà, nell'utilizzo pratico, intervengono sempre delle tolleranze. Per esempio, in alcuni dei nostri test, ci è capitato che per Period pari a 750 non tutti e tre i servomotori risultassero in posizione centrale, ma quello delle zampe di sinistra risultava ruotato di un'angolo minimo. Abbiamo quindi dovuto centrare i due servomotori (mediante il listato in figura), trovando che il motore delle zampe centrali e delle zampe di destra era centrato per Period pari a 750; quello delle zampe di sinistra lo era invece per 749. Come si vede sono variazioni minime, che fanno parte delle tolleranze delle normali apparecchiature elettroniche, e che in questo caso possono essere risolte mediante opportuni accorgimenti software.

CENTRARE I SERVOMOTORI

```
'*****************************************************************
'* Nome:          Centra ServoMotori                             *
'* Proces.:       PIC16F876                                      *
'* Note:          Permette di centrare i 3 Servo                 *
'*****************************************************************
'------[ Definizioni          ]-----------------------------------
DEFINE LOADER_USED 1          'Usato per boot-loader
DEFINE OSC 20                 'Imposta Clock a 20MHz
ADCON1 = %00000111            'Port A = Digitale
'------[ Verso Porte          ]-----------------------------------
TRISB = %00000000
'-----[ Definizioni I/O        ]-----------------------------------
Servo1  VAR     PORTB.2       'Porta Servo 1
Servo2  VAR     PORTB.1       'Porta Servo 2
Servo2  VAR     PORTB.0       'Porta Servo 3

'------[ Inizializzazione     ]-----------------------------------
PORTB = 0

'------[ Inizio programma     ]-----------------------------------
Inizio:
        PulsOut Servo1, 750     'Centra il Servo 1 con
                                '1500 µs
        PulsOut Servo2, 749     'Centra il Servo 2 con
                                '1498 µs
        PulsOut Servo3, 750     'Centra il Servo 2 con
                                '1500 µs
        Pause 20                'Aspetta 20 ms
        GoTo Inizio

End
```

Movimento con controllo degli ostacoli tramite infrarossi

```
'*****************************************************************
'* Nome:        Spider_IR                                        *
'* Note:        Movimento con controllo sui...                   *
'*              ...rilevatori a IR                               *
'*****************************************************************
'-----[ Definizioni          ]-----------------------------------------
DEFINE LOADER_USED 1        'Usato per boot-loader
DEFINE OSC 20               'Imposta Clock a 20MHz
ADCON1 = %00000111          'Port A = Digitale
'-----[ Verso Porte          ]-----------------------------------------
TRISA = %00110000           'Imposta pin Port A in
                            'Input e/o Output
TRISB = %00000000           'Imposta pin Port B in
                            'Input e/o Output
TRISC = %00000011           'Imposta pin Port C in
                            'Input e/o Output
'-----[ Definizioni I/O      ]-----------------------------------------
Servo1  VAR  PORTB.2        'Porta Servo 1 (DESTRA)
Servo2  VAR  PORTB.1        'Porta Servo 2 (SINISTRA)
Servo3  VAR  PORTB.0        'Porta Servo 3 (CENTRO)

Speaker VAR  PORTA.1        'Porta Speaker

IR_1    VAR  PORTC.1        'Porta Ricevitore Infrarosso

IFR_1   VAR  PORTC.5        'Porta Trx. Infrarosso destro
IFR_2   VAR  PORTC.2        'Porta Trx. Infrarosso sinistro
'-----[ Definizioni Variabili ]-----------------------------------------
Pos_Servo1  VAR  WORD       'posizione servo1 DESTRA
Pos_Servo2  VAR  WORD       'posizione servo2 SINISTRA
Pos_Servo3  VAR  WORD       'posizione servo3 CENTRO

mcount      VAR  BYTE       'loop per subrout. cammina
ncount      VAR  BYTE       'loop per gli altri
Nota        VAR  BYTE       'nota per sound

action      VAR  BYTE
VarAnd      VAR  BYTE       'variabile per and logico
Sinist_IR_det VAR BIT       'variabili per salvare lo...
Destr_IR_det VAR BIT        '...stato del ricevitore IR
'-----[ Definizioni Costanti  ]-----------------------------------------
pos_max_3   CON 900         'Posizione Massima CENTRO
pos_mid_3   CON 750         'Posizione Centrale CENTRO
pos_min_3   CON 600         'Posizione Minima CENTRO
pos_max_1   CON 900         'Posizione Massima DESTRA
pos_mid_1   CON 750         'Posizione Centrale DESTRA
pos_min_1   CON 600         'Posizione Minima DESTRA
pos_max_2   CON 900         'Posizione Massima SINISTRA
pos_mid_2   CON 750         'Posizione Centrale SINISTRA
pos_min_2   CON 600         'Posizione Minima SINISTRA

Ritardo     CON 20          'Pausa per Pulsout in µsecondi
Passo       CON 5           'Passo per ciclo Azzera
Passo_1     CON 7           'Passo per ciclo Camminata
Durata      CON 50          'Durata nota per Sound
'-----[ Inizializzazione     ]-----------------------------------------
PORTA =0
PORTB =0
PORTC =0
        Sinist_IR_det = 0
        Destr_IR_det = 0
        ncount = 0
        mcount = 0
        Nota = 0
        VarAnd = 0

'-----[ Inizio programma      ]-----------------------------------------
        Nota = 50
        GoSub Suono             'emetti suono

        Pos_Servo1 = pos_mid_1  'posizione inizale...
        Pos_Servo2 = pos_mid_2  '...dei tre servomotori...
        Pos_Servo3 = pos_mid_3  '...al centro
        GoSub Azzera

Inizio:
        'Verifica presenza ostacolo sulla sinistra.
        High IFR_2              'Attiva Trx. IR di sinistra
        Pause 1
        Sinist_IR_det = IR_1    'Legge stato Rx. IR
        Low IFR_2               'Disattiva Trx. IR
        Pause 1

        IF Sinist_IR_det = 0 Then  'Se presente ostacolo...
                Nota = 10          '...emetti suono
                GoSub Suono
        EndIF

        'Verifica presenza ostacolo sulla destra.
        High IFR_1              'Attiva Trx. IR di destra
        Pause 1
        Destr_IR_det = IR_1     'Legge stato Rx. IR
        Low IFR_1               'Disattiva Trx. IR
        Pause 1

        IF Destr_IR_det = 0 Then  'Se presente ostacolo...
                Nota = 100        '...emetti suono
                GoSub Suono
        EndIF

        'Determina quale azione eseguire
        action = 0
        'se 1 (ostacolo non presente) non fare nulla
        IF Sinist_IR_det = 1 Then cont1
        action = action +1
cont1:
        'se 1 (ostacolo non presente) non fare nulla
        IF Destr_IR_det = 1 Then cont2
        action = action +2
cont2:
'Action = 0, nessun ostacolo
'Action = 1, ostacolo sulla sinistra
'Action = 2, ostacolo sulla destra
'Action = 3, ostacolo di fronte

'Esegui l'azione indicata da Action
VarAnd = action & %00000011
Branch VarAnd, [Avanti, Destra, Sinistra, Inverti]
```

```
'-----[ Subroutine      ]----------------------------------------
Avanti:
        Pos_Servo1 = pos_min_1  'dietro zampe destra
        Pos_Servo2 = pos_min_2  'avanti zampe sinistra
        Pos_Servo3 = pos_max_3 'basso centrale sinistra
        GoSub Cammina

        Pos_Servo1 = pos_max_1 'avanti zampe destra
        Pos_Servo2 = pos_max_2 'dietro zampe sinistra
        Pos_Servo3 = pos_min_3  'basso centrale destra
        GoSub Cammina

        GoTo Inizio                 'Ritorna al ciclo main

Cammina:
        'Posiziona le zampe centrali
        For mcount=1 to 100 step Passo_1
                PulsOut Servo3,Pos_Servo3
                Pause Ritardo
        Next

        'Muove le zampe di destra e sinistra
        For mcount=1 to 100 step Passo_1
                PulsOut Servo1,Pos_Servo1   'destra
                PulsOut Servo2,Pos_Servo2   'sinistra
                PulsOut Servo3,Pos_Servo3   'centro
                Pause Ritardo
        Next
        Return

Suono:
        Sound Speaker, [Nota,Durata]        'Emette suono
        Low Speaker                         'Spegne speaker
        Return

Azzera:
        'Posiziona i tre servomotori in posizione centrale
        For ncount = 1 to 100 step Passo
                PulsOut Servo1,Pos_Servo1
                PulsOut Servo2,Pos_Servo2
                PulsOut Servo3,Pos_Servo3
                Pause Ritardo
        Next
        Return

Sinistra:
'Bisogna tornare indietro di tre passi e girare a destra...
'...per evitare l'ostacolo individuato dal baffo_1

        Nota = 10           'Emetti suono
        GoSub Suono

        GoSub Indietro              'Torna indietro di 3 passi

        For ncount=1 to 5                   'Gira a destra
                Pos_Servo1 = pos_max_1 'av. zampe dx
                Pos_Servo2 = pos_min_2  'av. zampe sx
                Pos_Servo3 = pos_max_3 'basso cent. sx
                GoSub Cammina

                Pos_Servo1 = pos_min_1  'ind. zampe dx
                Pos_Servo2 = pos_max_2 'ind. zampe sx
                Pos_Servo3 = pos_min_3  'basso cent. dx
                GoSub Cammina
        Next
        Goto Inizio

Destra:
'Bisogna tornare indietro di tre passi e girare a sinistra...
'...per evitare l'ostacolo individuato dal baffo_2

        Nota = 100          'Emetti suono
        GoSub Suono

        GoSub Indietro              'Torna indietro di 3 passi
        For ncount=1 to 5                   'Gira a sinistra
                Pos_Servo1 = pos_min_1  'ind. zampe dx
                Pos_Servo2 = pos_max_2 'ind. zampe sx
                Pos_Servo3 = pos_max_3 'basso cent. sx
                GoSub Cammina

                Pos_Servo1 = pos_max_1 'av. zampe dx
                Pos_Servo2 = pos_min_2  'av. zampe sx
                Pos_Servo3 = pos_min_3  'basso cent. dx
                GoSub Cammina
        Next
        GoTo Inizio

Inverti:
'Bisogna tornare indietro di tre passi e girare a 180°...
'...per evitare l'ostacolo individuato da entrabi i baffi

        Nota = 50           'Emetti suono
        GoSub Suono

        GoSub Indietro              'Torna indietro di 3 passi

        For ncount=1 to 7                   'Gira a sinistra di 180°
                Pos_Servo1 = pos_min_1  'ind. zampe dx
                Pos_Servo2 = pos_max_2 'ind. zampe sx
                Pos_Servo3 = pos_max_3 'basso cent. sx
                GoSub Cammina

                Pos_Servo1 = pos_max_1 'av. zampe dx
                Pos_Servo2 = pos_min_2  'av. zampe sx
                Pos_Servo3 = pos_min_3  'basso cent. dx
                GoSub Cammina
        Next
        GoTo Inizio

Indietro:
        For ncount=1 to 4           'Torna indietro di 3 passi
                Pos_Servo1 = pos_max_1 'av. zampe dx
                Pos_Servo2 = pos_max_2 'ind. zampe sx
                Pos_Servo3 = pos_max_3 'basso cent. sx
                GoSub Cammina

                Pos_Servo1 = pos_min_1  'ind. zampe dx
                Pos_Servo2 = pos_min_2  'av. zampe sx
                Pos_Servo3 = pos_min_3  'basso cent. dx
                GoSub Cammina
        Next
        Return
End
```

avervi dato anche tutte le informazioni necessarie per permettervi di realizzare quella che, a nostro parere, è la parte più interessante, divertente e appassionante della guida: ossia espandere con nuove funzionalità i nostri tre progetti, in modo che ognuno di voi possa personalizzare secondo i propri gusti e preferenze il proprio robot.

Ormai infatti dovreste avere una conoscenza approfondita di tutti i concetti fondamentali che riguardano la meccanica, l'elettronica e l'informatica che stanno dietro alla realizzazione di ciascun progetto che vi abbiamo presentato e, per esempio, non vi dovrebbe risultare difficile modificarne leggermente l'elettronica in modo da aggiungere nuove periferiche o dispositivi.

Un'idea che si potrebbe sviluppare potrebbe essere >

- Il compilatore Basic per PIC -

Per poter far funzionare ciascuno dei tre robot disponibili è necessario scrivere un programma che gli faccia eseguire ciò che desideriamo. Il programma può essere scritto in qualsiasi linguaggio, ma per poter essere trasferito al PIC16F876 è necessario che questo venga compilato nel formato .HEX, che è comprensibile dal micro.

Per i nostri esempi abbiamo utilizzato il linguaggio di programmazione Basic; ci siamo quindi dovuti munire di un compilatore Basic per PIC in grado di convertire i listati scritti in Basic in istruzioni del codice macchina.

All'interno dell'articolo abbiamo già fatto riferimento al Bootloader e al trasferimento da PC dei file .HEX; in questo box vediamo invece come ottenere dai listati scritti in Basic i relativi file .HEX.

Il nostro consiglio è di utilizzare il pacchetto **PicBasic Compiler** sviluppato dalla **microEngineering Labs, Inc.**

Con questo software è possibile scrivere un programma direttamente in Basic; sarà poi il compilatore a trasformarlo in un file scritto in codice macchina che potrà essere memorizzato nel micro tramite il Bootloader.

L'utilizzo di questo software rende la programmazione molto più semplice e veloce, consentendo di realizzare in poche righe di Basic ciò che invece richiederebbe molte più istruzioni in Assembler. Del software sono disponibili due versioni: una **Base** che permette di utilizzare funzioni avanzate di programmazione, comandi di salto, di interazione, ecc.; e una **Pro** che in più aggiunge la gestione degli interrupt, la possibilità di utilizzare array, una migliore gestione delle seriali sia a livello hardware che software, ecc. Per i nostri scopi è comunque sufficiente la versione Base del compilatore.

Entrambe le versioni sono distribuite in Italia dall'azienda **Futura Elettronica** di Rescaldina (MI). Il costo è di Euro 128,00 per la versione Base (codice PBC) e di Euro 284,00 per la versione Pro (codice PBC PRO).

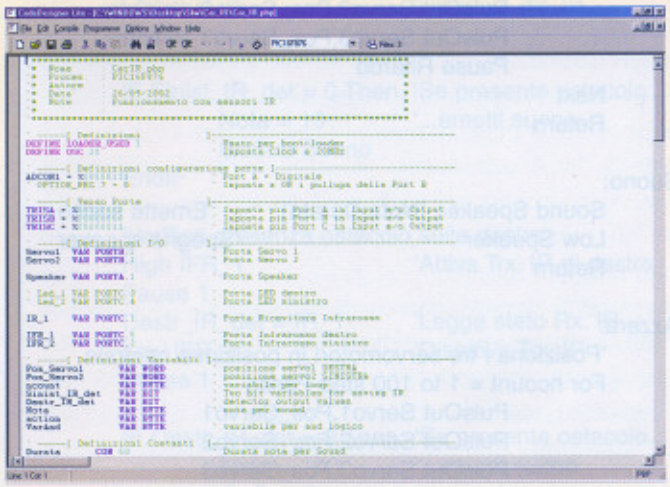


FIG. 1
Schermata di PicBasic Compiler. È possibile scrivere direttamente nella finestra il listato in Basic. Con un click si otterrà il codice compilato in formato .Hex.

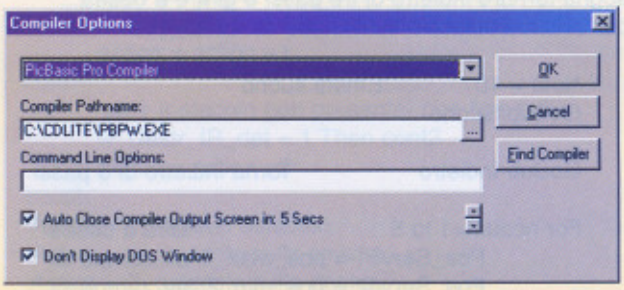


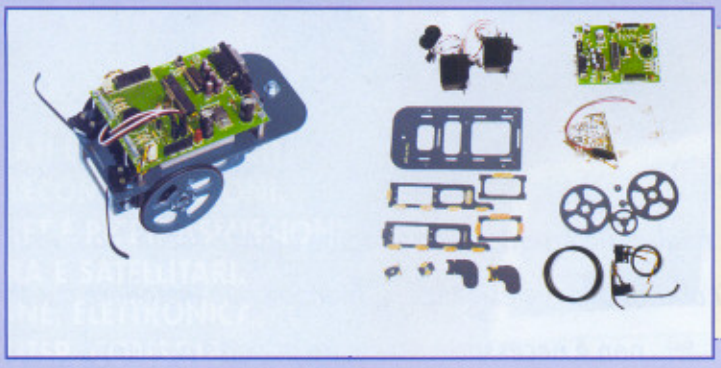
FIG. 2
Finestra di impostazione di PicBasic Compiler. è possibile specificare se utilizzare la versione Base o Pro e la directory in cui si trova il compilatore.

quella di cambiare leggermente e posizionare un secondo kit dei trasmettitori e ricevitore ad infrarosso nella parte anteriore dei robot ma, invece di utilizzarlo come rilevatore di ostacoli, si potrebbe realizzare un sistema che consenta di individuare ed aggirare eventuali buche o gradini.
Ancora si potrebbe pensare di collegare la motherboard dei robot ad un modulo ricevitore, in modo da realizzare una sorta di "telecomando" a distanza senza fili.
Allo stesso modo, se avete compreso il funzionamento dei listati Basic che vi abbiamo mostrato e descritto in questi mesi, dovreste ora essere in grado di modificarli in modo da adattarli alle vostre necessità, per esempio per gestire e comandare le nuove periferiche che aggiungerete.

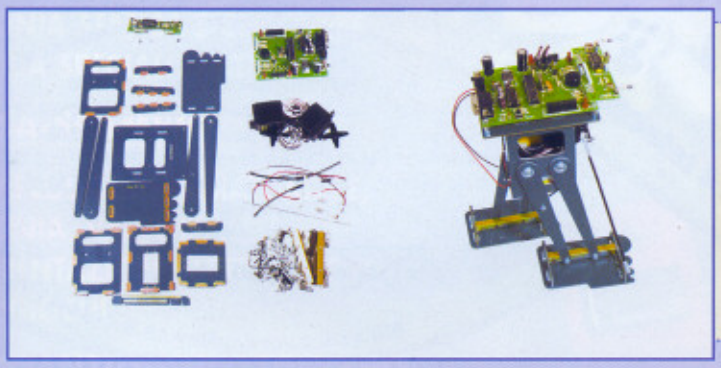
Per il

MATERIALE

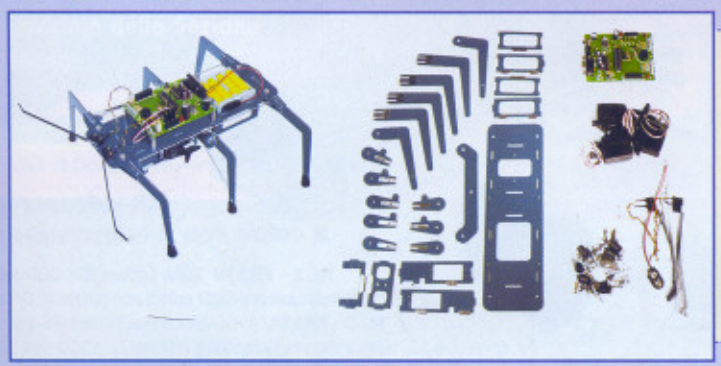
I tre robot sono disponibili in scatola di montaggio e possono essere richiesti all'azienda Futura Elettronica (Rescaldina-MI, V.le Kennedy 96) chiamando il numero di telefono 0331/576139 oppure inviando un fax allo 0331/466686.
E' anche possibile acquistare i prodotti ON-LINE collegandosi al sito *www.futuranet.it*



La scatola di montaggio "CarBot" comprende tutte le parti meccaniche da assemblare mediante saldatura a stagno, i due servo già modificati, le minuterie, la Motherboard, il micro programmato col bootloader, una serie di programmi demo ed un completo manuale d'istruzione.
Il kit costa **195,00 Euro** *(IVA compresa).*



Il kit del bipede "Filippo" comprende tutte le parti meccaniche da assemblare mediante saldatura a stagno, i due servomotori, le minuterie, la Motherboard, il sensore IR, il micro programmato col bootloader, una serie di programmi demo ed un completo manuale d'istruzione.
Il kit costa **220,00 Euro** *(IVA compresa).*



La scatola di montaggio del ragno "Spider" comprende tutte le parti meccaniche da assemblare mediante saldatura a stagno, i tre servomotori, le minuterie, la Motherboard, il micro programmato col bootloader, una serie di programmi demo ed un completo manuale d'istruzione.
Il kit costa **250,00 Euro** *(IVA compresa).*

CORSO DI PROGRAMMAZIONE Voice Extreme IC



Corso di utilizzo e programmazione dell'integrato Voice Extreme della Sensory. Questo chip è in pratica un microcontrollore ad 8 bit in grado anche di parlare e di comprendere comandi vocali. Impareremo a programmare il VE-IC realizzando applicazioni che utilizzano la voce come mezzo di controllo per apparecchiature o sistemi di sicurezza.

Decima e ultima puntata

a cura di *Ing. Roberto Nogarotto*

Già da alcune puntate del corso abbiamo iniziato l'analisi di piccoli programmi demo, che ci hanno permesso di capire come programmare il VE-IC e come gestire la demoboard che lo ospita. In tutti i listati che abbiamo visto e che utilizzeremo faremo uso del file *demob.veh*, presentato nella puntata di aprile (rivista 78), e che ci semplificherà la scrittura dei programmi. Brevemente vi ricordiamo che questo file è costituito in sostanza da una serie di direttive *#define*. In pratica queste direttive permettono di "sostituire" un valore numerico, o un'istruzione, con una parola.

PROGRAMMA demob_8

Questo programma effettua ancora il riconoscimento in tecnologia *Speaker Independent* (come il programma demob_7), solo che il risultato del riconoscimento, oltre ad essere visualizzato sul display, viene inviato attraverso la porta seriale al PC a cui è collegata la demoboard.

Per poter visualizzare i dati in arrivo dal VE-IC è sufficiente avviare, all'interno del programma *Sensory Voice Extreme IDE*, nel menu *Tools*, il programma *Debug Terminal*.

Questo programma non è altro che un terminale remoto che visualizza le informazioni in arrivo sulla porta seriale, una volta aperta la comunicazione pigiando il pulsante *Open Connection*. Occorre ricordare che la comunicazione sulla seriale avviene solo ed unicamente a 9600 baud.

Passiamo a questo punto ad analizzare il listato: la direttiva *#pragma Ve_APP_TEXT* ci permette di scrivere nel micro, al momento della programmazione, la frase riportata poi fra virgolette.

Il *main* è del tutto uguale a quanto visto nel programma *demob_7*, come pure parte della *Riconosci*. Unica differenza che, una volta identificato il nume- >

Listato del programma demob_8

```
//Pigiando il tasto 1, si avvia
//il riconoscimento SI.
//Se l'intervallo di confidenza è
//maggiore dell'80%, viene inviata
//alla seriale una stringa col numero
//riconosciuto.


#include "demob.veh"
#include "numeri.veh"


WEIGHTS *weight_set;


//Definizione Variabili
sint8 Tasto;
uint8 numero;
sint8 error;


//Definizione Funzioni
void  Riconosci();
void  Inviaseriale();
void  Visualizza();


#pragma VE_APP_TEXT "Numero = "

main()
{
    ConfigureIO(1,0,3);
    //Configurato D0 di 4511
    ConfigureIO(1,1,3);
    //Configurato D1 di 4511
    ConfigureIO(1,2,3);
    //Configurato D2 di 4511
    ConfigureIO(1,7,3);
    //Configurato D3 di 4511

    DelayMilliSeconds(200);
    Blank;

    do
    {
      Tasto=ScanKeypad();
      if (Tasto==1)  Riconosci();
    } while(FOREVER);
}
```

```
Riconosci()
{
 weight_set=&WTdigits;
 error=PatGenW(STANDARD,weight_set);
 if (error==0)  {
      Recog(&WTdigits);
      if (GetRecogLevel1()>80) {
           numero=GetRecogMatch1();
           Inviaseriale();
           Visualizza();
           DelaySeconds(2);
           Blank;
      }
 }
 Blank;
}


Inviaseriale() {
 Init232();
 DelayMilliSeconds(10);
 WriteString232(GetApplicationText());
 PutByte232(numero+48);
 PutByte232(13);
 Idle232();
}


Visualizza()
{
  if (numero==0)  Zero;
  if (numero==1)  Uno;
  if (numero==2)  Due;
  if (numero==3)  Tre;
  if (numero==4)  Quattro;
  if (numero==5)  Cinque;
  if (numero==6)  Sei;
  if (numero==7)  Sette;
  if (numero==8)  Otto;
  if (numero==9)  Nove;
}
```

ro pronunciato, viene richiamata una routine chiamata *Inviaseriale*. Vediamo come lavora: la funzione *Init232* serve per inizializzare la comunicazione seriale.
La funzione *WriteString* serve per inviare serialmente i codici ascii di una stringa.

Nel nostro caso, la stringa da inviare viene recuperata dalla funzione *GetApplicationText*. Poiché questa era "*Numero* =", verranno inviati alla seriale i codici ascii dei singoli caratteri, che verranno di conseguenza visualizzati nella finestra del debug terminal.

Inizio listato del programma demob_9

```
//Tenendo premuto il tasto 1 dopo un
//reset si entra nella fase di training.
//Occorre istruire due parole.
//Nella fase Continuos Listening,
//occorre pronunciare in sequenza due
//parole. Se vengono riconosciute,
//vengono emessi tre beep.

#include "demob.veh"

extern  SPEECH  frasi;

//Template temporaneo
TEMPLATE  temptemplate[1];

//TEMPLATES:
//Parole da riconoscere in
//continuos listening.
TEMPLATE  parolecl[1];

//Parole da riconoscere in
//speaker dipendent.
TEMPLATE  parolesd[1];

sint8 error;
uint8 result;
uint8 utente;

sint8 Tasto;

//Definizione nuove funzioni
void  Riconosci();
void  TrainingUtente1();
void  Chiedipassword1();
void  Ripetipassword1();
void  Chiedipassword2();
void  Ripetipassword2();


main() {
     ConfigureIO(1,0,3);
     //Configurato D0 di 4511
     ConfigureIO(1,1,3);
     //Configurato D1 di 4511
     ConfigureIO(1,2,3);
     //Configurato D2 di 4511
     ConfigureIO(1,7,3);
     //Configurato D3 di 4511
     DelayMilliSeconds(100);
     Blank;
     BEEP;
     DelayMilliSeconds(300);
     BEEP;
     DelaySeconds(1);

     //Verifica se è premuto un pulsante
     Tasto=ScanKeypad();
     if (Tasto==1) TrainingUtente1();
     do {
       //Aspetta una parola
       result=CLPatGen(STANDARD);
       //Se è arrivata correttamente
       if (result==0) {
         //Prova a riconoscere sul
         //template ParoleCL
         result=RecogSD(1,parolecl);
         if (result==0) {
           BEEP;
           //Aspetta una parola
           result=PatGen(STANDARD);
           //Se è arrivata correttamente
           if (result==0) {
             //Prova a riconoscere sul
             //template ParoleSD
             result=RecogSD(1,parolesd);
             if (result==0) {
               BEEP;
               DelayMilliSeconds(300);
               BEEP;
               DelayMilliSeconds(300);
               BEEP;
               DelayMilliSeconds(300);
             }
           }
         }
       }
     } while(FOREVER);
}
```

Con la funzione *PutByte* inviamo invece un singolo byte alla seriale.
Poiché i codici ascii dei numeri da 0 a 9 corrispondono ai codici da 48 a 57, per visualizzare il numero corretto sul *debug terminal*, dovremo inviare il numero sommato a 48.

Viene poi inviato il codice 13 che corrisponde al ritorno a capo (*invio*). In questo modo, al successivo riconoscimento di un numero, questo verrà scritto nella riga immediatamente sotto al precedente.
Con la funzione *Idle232*, infine, si disabilita la funzione di comunicazione seriale. >

Continuazione listato del programma demob_9

```
Chiedipassword1() {
  uint8 i;
  Talk(2,&frasi);  //Pronuncia
  Talk(3,&frasi);  //Password
  Talk(0,&frasi);  //Uno
  for (i=0;i<3;i++) {
    error=PatGen(STANDARD);
    if (error==0) break;
    else {
      Talk(4,&frasi);    //Ripeti
      DelayMilliSeconds(300);
    }
  }
}

Ripetipassword1() {
  uint8 i;
  Talk(4,&frasi);  //Ripeti
  Talk(3,&frasi);  //Password
  Talk(0,&frasi);  //Uno
  for (i=0;i<3;i++)  {
    error=PatGen(STANDARD);
    if (error==0) break;
    else  {
      Talk(4,&frasi);    //Ripeti
      DelayMilliSeconds(300);
    }
  }
}

Chiedipassword2() {
  uint8 i;
  Talk(2,&frasi);  //Pronuncia
  Talk(3,&frasi);  //Password
  Talk(1,&frasi);  //Due
  for (i=0;i<3;i++) {
    error=PatGen(STANDARD);
    if (error==0) break;
    else {
      Talk(4,&frasi);    //Ripeti
      DelayMilliSeconds(300);
    }
  }
}

Ripetipassword2() {
  uint8 i;
  Talk(4,&frasi);  //Ripeti
  Talk(3,&frasi);  //Password
  Talk(1,&frasi);  //Due
  for (i=0;i<3;i++) {
    error=PatGen(STANDARD);
    if (error==0) break;
    else {
      Talk(4,&frasi);    //Ripeti
      DelayMilliSeconds(300);
    }
  }
}

TrainingUtente1() {
  Uno;
  BEEP;
  DelaySeconds(1);
  Chiedipassword1();
  if (error==0) {
   PutTemplate(UNKNOWN,0,temptemplate);
   DelayMilliSeconds(500);
   Ripetipassword1();
   if (error==0) {
     GetTemplate(KNOWN,0,temptemplate);
     error=TrainSD(UNKNOWN,KNOWN,UNKNOWN);
     if (error==0) {
       PutTemplate(UNKNOWN,0,parolec1);
       BEEP;
       DelayMilliSeconds(500);
       error = 0;
     }
   }
  }
  if (error==0) {
    Due;
    Chiedipassword2();
    if (error==0) {
      PutTemplate(UNKNOWN,0,temptemplate);
      DelayMilliSeconds(500);
      Ripetipassword2();
      if (error==0) {
        GetTemplate(KNOWN,0,temptemplate);
        error=TrainSD(UNKNOWN,KNOWN,UNKNOWN);
        if (error==0) {
          PutTemplate(UNKNOWN,0,parolesd);
          BEEP;
          DelayMilliSeconds(500);
          error=0;
        }
      }
    }
  }
  Blank;
}
```

PROGRAMMA demob_9

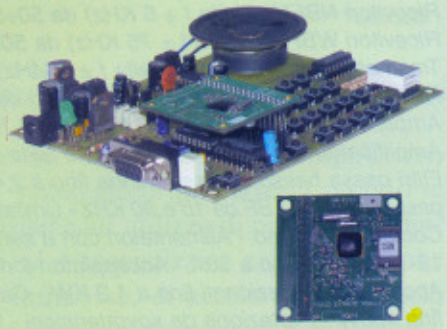
Con questo programma andiamo a studiare il funzionamento della tecnologia *Continuous Listening*. In questa modalità di funzionamento, il sistema è sempre in attesa dell'arrivo di una parola da riconoscere.
Tipicamente, una applicazione completa prevede l'utilizzo di una parola da riconoscere in *Continuous Listening* e, se questa viene riconosciuta, attende l'arrivo di una seconda parola, da riconoscere questa volta in *Speaker Independent*.
Vediamo quindi come è stato realizzato questo programma: nel *main* viene testata la pressione del pulsante 1.
Se questo viene trovato pigiato, il programma passa alle fase di training, passando alla *Trainingutente1*.
Questa funzione è quasi del tutto simile a quanto già visto nel programma *demob_4;* l'unica differenza consiste nel fatto che la prima parola viene salvata in un template chiamato *parolecl*, mentre la seconda viene salvata in un template chiamato *parolesd*. Se non si entra in fase di apprendimento, tenendo quindi premuto il pulsante 1 all'accensione o al reset della scheda, il programma entra in un ciclo costituito fondamentalmente dalla funzione *CLPatGen*.
Con questa funzione, il VE-IC resta in attesa dell'arrivo di un segnale vocale.
Se questo segnale arriva e se, come al solito, è adatto al riconoscimento (*result = 0*), con la *RecogSD* si tenta il riconoscimento vero e proprio rispetto all'array *parolecl* (questo array è costituito in realtà da un solo template).
Se il riconoscimento va a buon fine si emette un beep ed il micro attende, con la funzione *PatGen*, l'arrivo di un'altro segnale vocale.
Se anche questa va buon fine, così come il successivo riconoscimento effettuato dalla funzione *RecogSD* sull'array *parolesd*, il programma emette tre beep a breve distanza l'uno dall'altro ad indicare il corretto riconoscimento di entrambe le parole.

I TOOLS di sviluppo disponibili



E' il tool di sviluppo originale messo a disposizione dal produttore Sensory, contiene una Demoboard/Programmatore con a bordo un modulo basato sul VE-IC. La Demoboard dispone delle seguenti risorse: un microfono, un altoparlante, un'interfaccia RS232 per il collegamento al PC, dei LED e dei pulsanti per testare i programmi demo allegati. è poi presente un'area millefori nella quale è possibile realizzare prototipi di circuiti. Il software allegato comprende: l'ambiente di sviluppo (IDE) del VE-IC, attraverso il quale è possibile scrivere il programma in C e scaricarlo attraverso la seriale nella memoria flash; il programma Quick Synthesis per elaborare i file vocali e musicali; diversi file di esempio e tutta la documentazione necessaria. La confezione del Toolkit Voice Extreme (codice VET) è disponibile completa del modulo VEM al prezzo di **158,00 euro**.

La Demoboard/Programmatore presentata in queste pagine è disponibile in scatola di montaggio (cod. FT453K) al prezzo di 45,00 euro. Il kit comprende tutti i componenti, la basetta forata e serigrafata, l'altoparlante, il microfono, un cavo seriale per il collegamento al PC, le minuterie, un CD con l'ambiente di sviluppo IDE del VE-IC attraverso il quale è possibile scrivere il programma in C e scaricarlo attraverso la seriale nella memoria flash, il programma Quick Synthesis, la documentazione originale in lingua inglese e i programmi dimostrativi in italiano a livello sorgente accuratamente commentati. Il kit non comprende il Modulo Voice Extreme (cod. FT455M) che è disponibile separatamente al prezzo di 39,**00 euro.**



Tutti i prezzi si intendono IVA compresa. Il materiale va richiesto a ***Futura Elettronica****, V.le Kennedy 96, 20027 Rescaldina (MI), tel. 0331/576139, www.futuranet.it.*



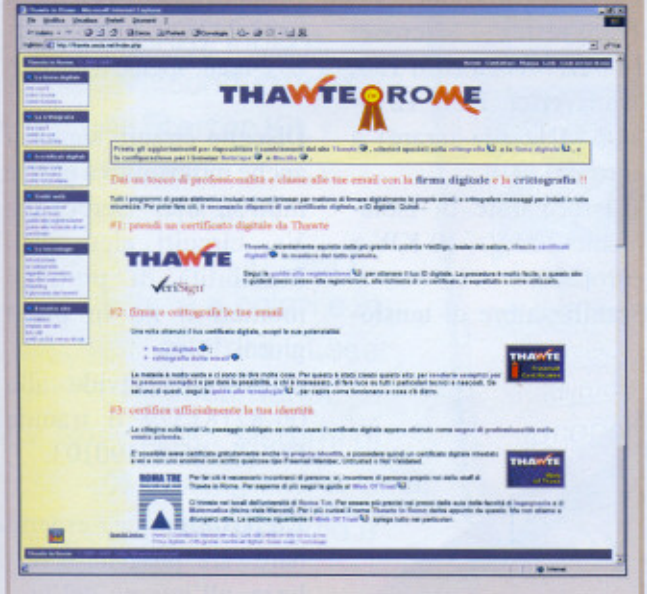
a cura della redazione

http://www.national.com



Sito della National Semiconductors, azienda americana leader nella produzione di semiconduttori che integrano, in un unico chip, sistemi analogici e tecnologie digitali. Le aree di applicazione spaziano dalle comunicazioni wireless all'elaborazione delle informazioni dall'acquisizione delle immagini all'audio, ecc.). Uno dei punti di forza dell'azienda è l'alta capacità di integrazione dei propri dispositivi.

http://thawte.ascia.net



Sito dedicato all'azienda Thawte (recentemente acquisita da VeriSign) che rilascia gratuitamente certificati digitali; questi ultimi sono l'elemento chiave di tutte le moderne tecnologie informatiche riguardanti la sicurezza. Il sito comprende anche alcune piccole sezioni didattiche in cui vengono spiegati cosa sono, come si usano e come funzionano le firme digitali, la crittografia e i certificati digitali. Segnaliamo la sezione "Le tecnologie".

http://www.murata.it



Sito dell'azienda Murata Elettronica S.p.A. che, sfruttando il vantaggio di produrre in proprio le ceramiche insieme all'utilizzo di processi produttivi esclusivi, si è specializzata nella produzione di sensori a infrarosso, di accelerazione, shock, velocità angolare, rotazione e temperatura. L'esempio più recente è il GYROSTAR, sensore di velocità angolare che riesce a ottimizzare l'effetto della forza di Coriolis.