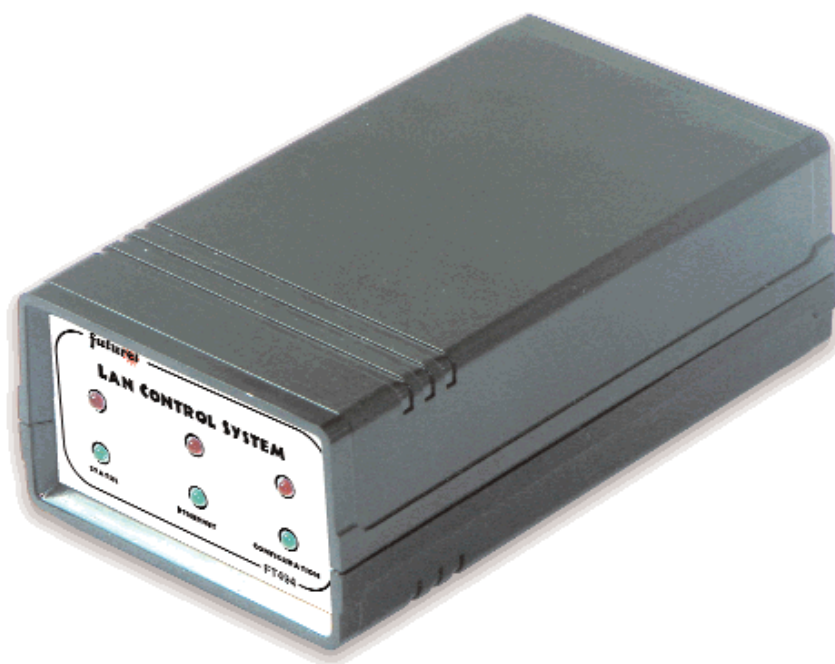


# ACCEDERE DA REMOTO TRAMITE INTERNET AD UN COMPUTER

## FT494K

Se la vostra occupazione vi richiede di dover lavorare su due o più differenti PC posizionati in diverse località, probabilmente saprete che uno dei maggiori problemi che bisogna affrontare è quello della sincronizzazione dei dati tra le due postazioni. Un esempio tipico può essere il caso in cui la maggior parte del lavoro viene svolto in ufficio, ma una certa percentuale viene realizzata a casa, magari durante il weekend o il tempo libero la sera. Il classico problema che si può verificare in questi casi è che si passi tutto il fine settimana a lavorare da casa a un progetto, e poi il lunedì successivo ci si dimentica di copiare il lavoro su un dischetto o su un CD in modo da aggiornare la versione presente in ufficio. Una soluzione al problema sarebbe quella di accedere, tramite internet, al computer di casa, in modo da scaricare la nuova versione; se infatti il PC è collegato a internet mediante un collegamento veloce di tipo ADSL scaricarsi il lavoro richiederebbe pochi istanti. Peccato però che il computer risulta spento e (quasi sicuramente...) a casa non c'è nessuno a cui telefonare per chiedere di accenderci il PC. In questa situazione risulterebbe quindi molto utile un dispositivo che ci consenta, tramite internet, di collegarci e di



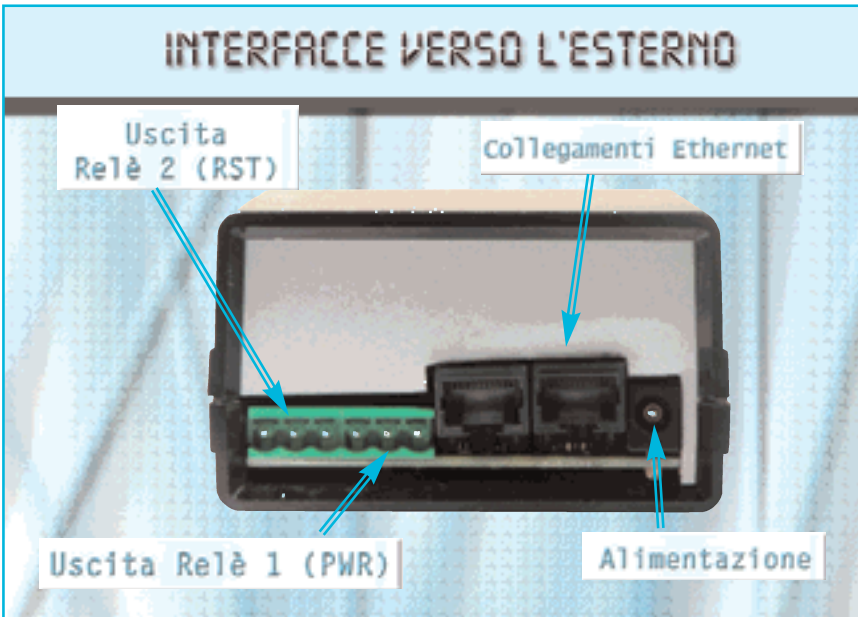
comandare da remoto l'accensione del PC.

Il progetto descritto in queste pagine realizza proprio ciò di cui abbiamo bisogno per risolvere questo genere di problema; utilizzando infatti un modulo ethernet EM100 prodotto dalla Tibbo Technology consente di gestire due relè che, come vedremo in seguito, saranno opportunamente collegati in modo da permetterci di accendere o spegnere il PC da un qualsiasi altro computer connesso a internet; inoltre nella malaugurata sorte che durante le operazioni il sistema operativo dovesse bloccarsi, è presente la possibilità di resettarlo.

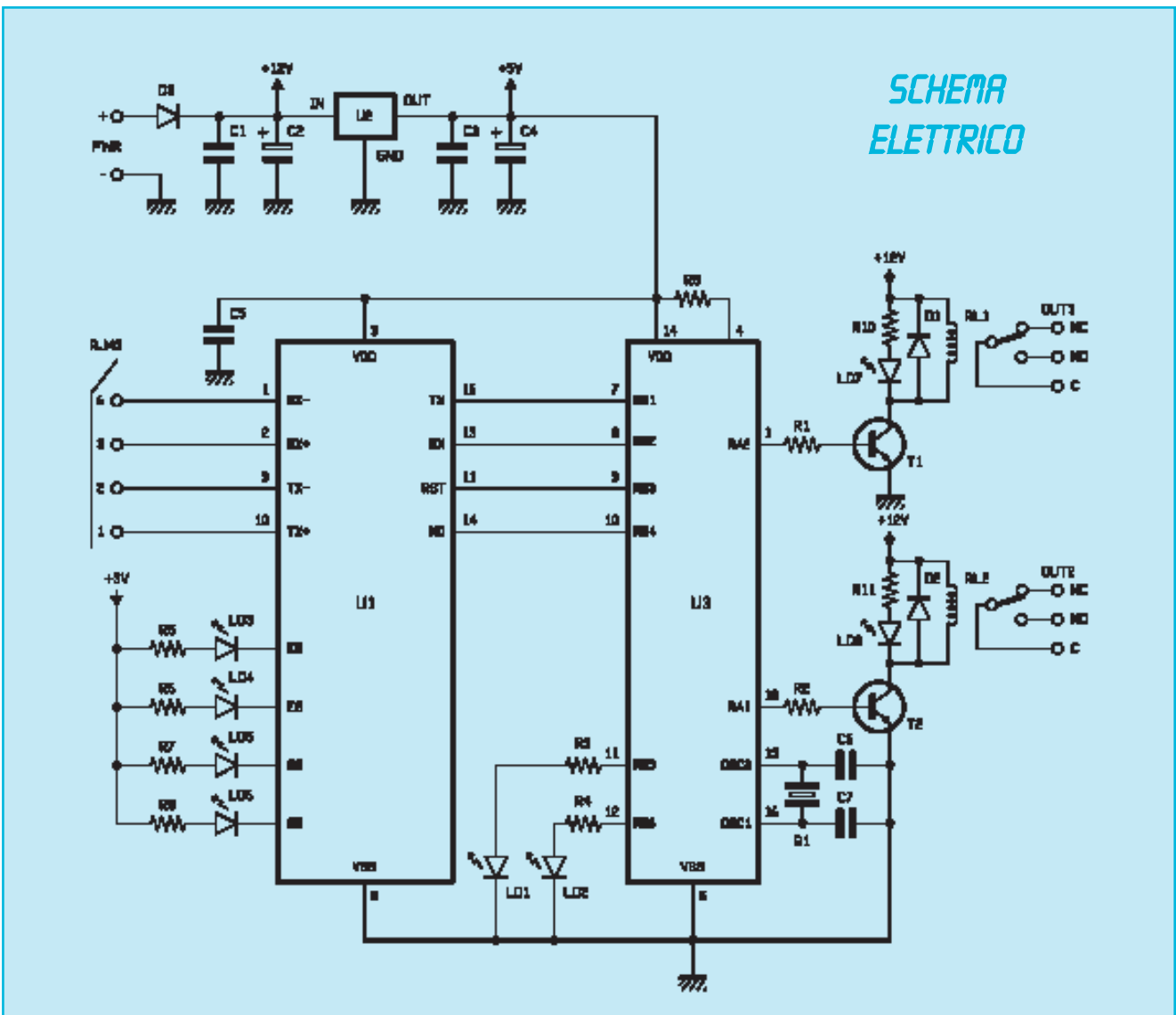
Chiaramente, per poter accedere alle informazioni memorizzate, non

è sufficiente la sola accensione del computer; in precedenza bisognerà aver impostato il PC in modo che all'accensione avvii un software (PCAnywhere della symantec, per esempio) che permetta il download e l'upload dei dati.

Un vantaggio della soluzione da noi proposta è che si massimizza la sicurezza del computer. Qualcuno sostiene che un PC veramente sicuro è soltanto quello spento, staccato da internet e dalla rete di alimentazione. Non ci sentiamo di condividere questa teoria (un po' troppo pessimista, a nostro parere); certo è però che più tempo un PC resta connesso a internet (soprattutto utilizzando un indirizzo IP fisso) più questo è a rischio di attacchi ester- ➤

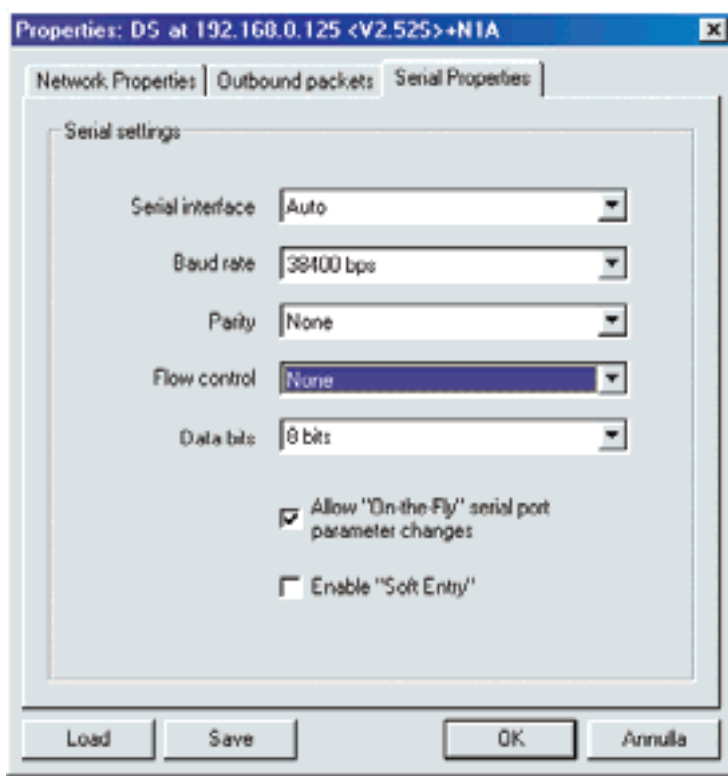


ni. Utilizzando il nostro sistema si riduce quindi al minimo tale tempo; il PC viene infatti acceso solo per il tempo necessario all'invio e allo scarico dei dati. Conclusa questa operazione, il computer può essere spento "proteggendolo" da eventuali rischi provenienti dall'esterno. Un'ultima nota riguarda la possibilità di accesso sia al nostro progetto che al PC: entrambi devono infatti essere raggiungibili dall'esterno tramite internet. Se quindi avete una connessione ADSL realizzata tramite un router (con magari l'aggiunta di un firewall, sia hardware che software), dovrete impostarli in modo da accettare tentativi di con-



## SCHERMAE DEI SOFTWARE

Sotto a sinistra viene mostrata la finestra di impostazione delle proprietà della porta seriale del programma Connection Wizard. È necessario impostare un baud rate di 38.400 bps, nessuna parità, 8 bit di dati e nessun controllo di flusso. Sotto a destra viene invece mostrata una schermata del software di gestione Controllo PC. Per collegarsi bisogna specificare l'IP dell'EM100 (o dell'eventuale router) e la porta TCP da utilizzare.



nessioni provenienti dall'esterno. Come vedremo anche in seguito il nostro dispositivo, per comunicare, utilizza il protocollo TCP/IP e una relativa porta; richiede inoltre che gli venga assegnato un indirizzo IP valido all'interno della rete locale. Per questo motivo è necessario aprire sul router ed eventuale firewall la porta di comunicazione utilizzata; inoltre dovrete aggiornare le tabelle di instradamento del router in modo che questo sappia come indirizzare il traffico.

Prima di concludere l'introduzione al dispositivo, vi facciamo notare che il nostro progetto è costituito da due parti: una prima parte hardware rappresentata dal dispositivo elettronico, e una seconda parte software costituita da un programma

(scritto in Delphi) che realizza e gestisce il collegamento al circuito tramite internet. Il software può essere scaricato gratuitamente dal sito [www.futuranet.it](http://www.futuranet.it) in modo da permettervi di accedere al circuito da qualunque PC connesso a internet, senza la necessità di dover avere a disposizione un dischetto o un computer in cui il software è già stato installato.

### Schema elettrico

Passiamo, a questo punto, ad analizzare lo schema elettrico del dispositivo. La parte più importante è sicuramente rappresentata dal modulo Ethernet EM100 la cui presentazione, vi ricordiamo, è apparsa sul numero 77 della rivista. Questo

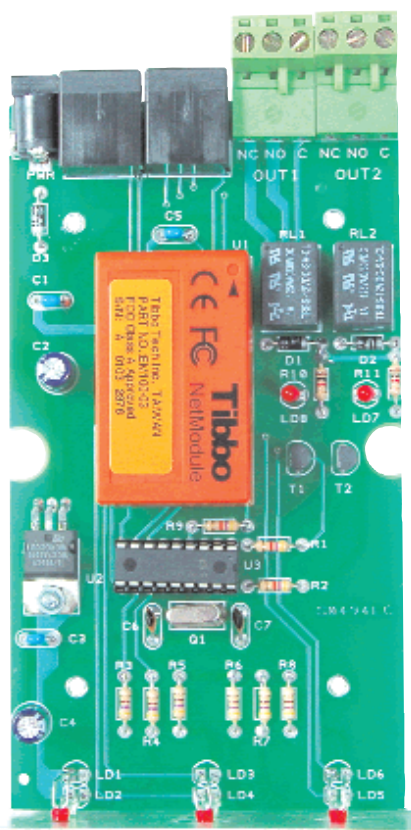
modulo è in grado di interfacciare una rete di tipo Ethernet con una periferica di tipo seriale. Oltre che ricevere i dati provenienti dalla LAN e renderli disponibili in formato seriale (e viceversa, ossia ricevere le informazioni in formato RS232 e trasmetterle in formato RJ45 10BaseT sulla rete Ethernet) il chip è anche in grado di gestire correttamente i protocolli TCP/IP, UDP/IP, ICMP e ARP. Ciò significa che l'EM100 andrà programmato in modo da specificare un indirizzo IP valido che lo individuerà univocamente; è in grado di accettare, stabilire e comunicare mediante connessioni UDP (connection less) o TCP (connection oriented); in quest'ultimo caso riesce a gestire gli eventuali errori che si verificano ➤

durante la trasmissione, riesce a ordinare eventuali pacchetti che arrivano fuori sequenza ed è in grado di gestire il meccanismo della finestra della trasmissione dei dati tipica appunto del TCP.

Il supporto al protocollo ICMP (Internet Control Message Protocol) permette invece la corretta trasmissione e ricezione delle informazioni di controllo (gli esempi più noti sono l'echo reply utilizzato per il "ping" o il destination unreachable); infine il supporto a ARP (Address Resolution Protocol) permette al modulo EM100 di gestire autonomamente la risoluzione degli indirizzi IP in indirizzi fisici MAC.

Come si nota dallo schema elettrico il modulo si interfaccia direttamente alla rete ethernet tramite i piedini 1 e 2 (linea di input negativa e positiva); dall'altro lato i dati vengono invece resi disponibili sui pin 15 e 16, rispettivamente input e output della linea seriale. Le informazioni possono quindi essere trasmesse direttamente ai pin che realizzano le porte seriali di molti microcontrollori (nel nostro progetto abbiamo utilizzato un PIC16F628, chip U2). Il formato dei livelli delle tensioni in uscita è TTL; non è quindi necessario dover utilizzare un integrato tipo il MAX232 per convertirli. Inoltre nel nostro caso il microcontrollore scelto è già munito di una porta UART interna, che ci permette quindi di gestire l'invio e la ricezione dei dati tra il PIC e l'EM100 senza dover ricorrere a ulteriori dispositivi esterni.

Il modulo EM100 e il microcontrollore sono inoltre collegati tramite due ulteriori pin, indicati con RST e MD su U1. Il primo piedino indica il reset del modulo, mentre il secondo indica la selezione del modo di funzionamento. Infatti, sempre nell'articolo pubblicato sulla rivista 77, era stato sottolineato che l'EM100 è in grado di funzionare in



tre diverse modalità: *Normal mode* nella quale realizza la conversione Ethernet/seriale; *Serial programming mode* utilizzata per programmarlo e *Firmware download mode* utilizzata per l'upgrade del firmware presente nel modulo. La selezione della modalità di funzionamento avviene tramite i due piedini appena indicati; nel nostro circuito si utilizza la prima modalità la cui selezione viene gestita automaticamente dal software presente nel microcontrollore.

Il modulo ethernet è inoltre provvisto di 4 pin (predisposti al collegamento verso 4 led) che indicano lo stato sia del modulo che della connessione verso la LAN: le linee contrassegnate come SG e SR indicano la modalità di funzionamento (*configuration*) dell'EM100 ed eventuali condizioni di errore; invece le linee EG e ER danno informazioni sulle operazioni compiute dalla porta ethernet. In particolare

il led LD4 è normalmente acceso; viene spento momentaneamente proprio quando l'EM100 riceve dei pacchetti dalla porta ethernet; il led LD3 invece è normalmente spento e viene acceso momentaneamente quando è rilevata una collisione sempre sulla rete ethernet.

All'interno del circuito sono inoltre presenti due led aggiuntivi (LD1 e LD2, indicati con *status*) gestiti dal microcontrollore e utilizzati per visualizzare informazioni sullo stato di quest'ultimo.

Il controllo dei due relè RL1 e RL2, utilizzati per accendere/spengere o resettare il PC, viene invece gestito dai pin RA1 e RA2 tramite i due transistor TI e T2. A questo punto è necessario fare una piccola nota: tutti i moderni alimentatori che rispettano il formato ATX offrono la possibilità di essere accesi o spenti sia via software sia utilizzando un pulsante di accensione. Inoltre, in molti casi, la scheda madre presente all'interno del PC (anch'essa in formato ATX) prevede due connettori indicati con PWR e RST. Il primo è collegato al pulsante di accensione e serve appunto per accendere o spegnere il PC; il secondo è invece utilizzato per il reset del computer e risulta collegato al relativo pulsante.

In particolare è previsto che corto circuitando il primo connettore per un breve periodo (nel nostro caso circa 2 secondi) se il computer è collegato all'alimentazione e spento, questo viene acceso. Se invece il PC è acceso, chiudendo lo stesso connettore per un tempo più lungo (nel nostro esempio circa 10 secondi) il computer viene spento.

Cortocircuitando invece il secondo connettore per un tempo breve (nel nostro caso sempre circa 2 secondi) viene comandato un reset del PC.

Il nostro dispositivo utilizza proprio queste funzionalità delle moderne mainboard, per accendere/spengere ➤

o resettare il PC, chiudendo per tempi differenti i due relè RL1 e RL2 che, come vedremo nella parte relativa alla realizzazione pratica, andranno collegati ai due connettori presenti sulla scheda madre.

All'interno del PC, l'accensione o il reset viene gestito direttamente dal BIOS (e non dal sistema operativo): ciò permette di resettare o spegnere il PC anche in caso di blocco del S.O. (se usate Windows, sicuramente saprete che è un'eventualità tutt'altro che rara...); per il corretto funzionamento è necessario quindi impostare il BIOS della scheda madre affinché sia in grado di gestire il tutto. Generalmente bisogna selezionare, dopo essere entrati nel BIOS, la voce "Soft-Off by PWRBTN" e indicare il valore "Delay 5 sec". Esistono però anche alcune varianti, dipendenti dal produttore della scheda madre e del BIOS. Per i dettagli vi suggeriamo quindi di riferirvi al manuale della vostra motherboard.

L'ultima parte del circuito che ci resta da analizzare è la sezione di alimentazione, composta dal chip U3, un integrato LM7805 che converte la tensione +12V ai +5V utilizzati dai dispositivi TTL.

Un particolare che non si nota dall'analisi dello schema elettrico ma che risulta molto importante è che sul circuito sono presenti due porte ethernet che risultano collegate in parallelo: tutti i dati che entrano da

una porta vengono trasmessi anche sull'altra; questo significa che una porta del dispositivo può essere collegata a un cavo proveniente dall'hub della rete LAN (o dal router ADSL, ecc.), mentre l'altra può essere collegata alla porta ethernet presente nel PC. In questo modo i due dispositivi risultano indirizzabili singolarmente.

### Protocollo di comunicazione

Dopo aver visto il funzionamento dello schema elettrico del circuito, vediamo ora il formato del protocollo utilizzato per la comunicazione tra il dispositivo e il programma di gestione remoto.

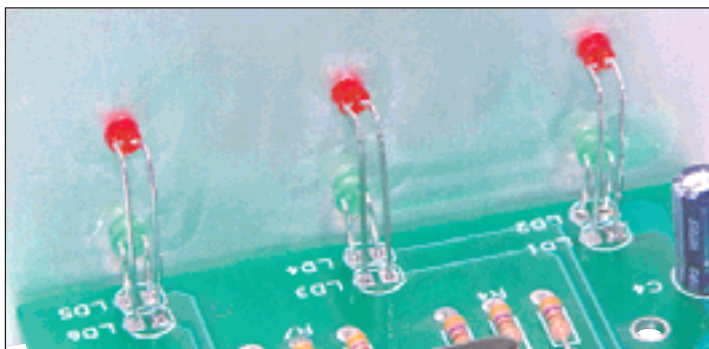
Le operazioni che il software può richiedere al circuito sono l'attivazione dei due relè; la richiesta dello stato logico e l'impostazione del tempo di attivazione degli stessi.

Come abbiamo appena visto, l'accensione del computer avviene attivando per circa 2 secondi il relè RL1: questo è ottenuto inviando al circuito il comando `"*/S"`. Lo spegnimento del computer è invece ottenuto attivando per circa 10 secondi lo stesso RL1; questo si ottiene inviando il comando `"*/E"`. Infine, il reset del PC, viene eseguito attivando il relè RL2 per circa 2 secondi (comando `"*/R"`). Per ogni comando inviato, il dispositivo risponde con una stringa che conferma l'esecuzione dell'operazione.

Per richiedere lo stato corrente dei relè si utilizza il comando `"*/I"`; il circuito risponde con le stringhe `"RELÈ i ON"` o `"RELÈ i OFF"` in cui *i* indica il relè (può quindi valere 1 o 2) mentre `"ON"` e `"OFF"` indicano rispettivamente che i relè sono attivati o meno.

Fin qui abbiamo visto la parte del protocollo di comunicazione relativa all'utilizzo del dispositivo per accendere o spegnere il PC. Il circuito può però essere utilizzato anche per qualsiasi altra applicazione che richieda di comandare da remoto tramite internet l'attivazione di due relè. Per questo è stata prevista la possibilità di modificare i tempi di attivazione dei due relè; inviando `"*/1"` si indica che si vuole impostare il tempo relativo a RL1 (`"*/2"` indica invece il relè RL2). A questo punto il circuito risponde con `"->INSERIRE TEMPO RELÈ 1<-"`. Inviando, per esempio, `"*/25"` si imposta un tempo di circa 25 secondi. I tempi impostabili vanno da 1 a 60 secondi; se si invia `"*/00"` si indica la modalità bistabile.

Prima di passare ad analizzare il software utilizzato per il collegamento, notiamo che come prima operazione è necessario programmare il modulo EM100 presente nel circuito. La programmazione può essere eseguita tramite il software *Connection Wizard*, presentato nel numero 77 della rivista. In particolare andrà impostato l'indirizzo IP ➤



*Un particolare relativo alla realizzazione pratica del dispositivo. Nell'immagine viene mostrato come saldare alla basetta i 6 led di segnalazione. Notate inoltre che per quelli rossi bisogna lasciare i terminali metallici più lunghi rispetto a quelli verdi. Nella superficie metallica del contenitore realizzate 6 fori di diametro 3 mm; per fissare i led alla superficie utilizzate una colla del tipo cianoacrilica.*

da associare al dispositivo; andrà scelto il protocollo TCP e relativa porta di comunicazione da utilizzare. Inoltre bisognerà scegliere, come *routing mode*, la modalità *Slave*. Infine, nella finestra delle proprietà della comunicazione seriale, andranno disabilitati i controlli di flusso; bisognerà selezionare un baud rate di 38.400 bit/sec, 8 bit di dati e nessuna parità.

### Software di collegamento

Analizziamo a questo punto il software utilizzato per collegarsi al dispositivo. Vi ricordiamo che il programma (scritto in Delphi e compatibile con Windows 98/XP/ME) è disponibile sul sito internet [www.futuranet.it](http://www.futuranet.it); in questo modo non è necessario avere sempre a disposizione un PC o un dischetto munito di tale software, ma quando vi necessita lo potete semplicemente scaricare dalla rete ed eseguire ovunque vi troviate.

Una volta lanciato il programma, come prima operazione è necessario specificare l'indirizzo IP utilizzato per accedere dall'esterno al circuito (se è presente un router, sarà pari all'IP della sua interfaccia esterna) e la porta di comunicazione utilizzata. A questo punto il software apre un *socket* verso il dispositivo. Un *socket* altro non è che un collegamento realizzato mediante una rete internet: in generale ogni socket è caratterizzato dal tipo di protocollo utilizzato (nel nostro caso TCP); dagli indirizzi IP sorgente e destinazione e dalle porte di comunicazione utilizzate sia dalla sorgente che dalla destinazione. Nel nostro caso, per quanto riguarda la coppia IP/porta di destinazione, saranno pari a quelli specificati; l'IP di sorgente sarà invece pari all'IP del PC da cui stiamo eseguendo l'operazione (verrà specificata automaticamente dal sistema operativo); anche la porta di comunicazione della sor-

gente verrà assegnata dal S.O.

Una volta che il collegamento è stato instaurato, viene utilizzato il protocollo di comunicazione appena visto per inviare i comandi di attivazione dei relè, per ricevere gli stati degli stessi o per inviare le impostazioni sul tempo di attivazione.

L'utilizzo del programma è abbastanza semplice e intuitivo: sono presenti tre pulsanti che accendono, spengono o resettano il computer. Un ulteriore pulsante richiede lo stato dei relè che verrà visualizzato utilizzando due "led" colorati.

Il dispositivo, oltre che per accendere o spegnere un computer, può essere anche utilizzato più in generale per applicazioni in cui è richiesto di gestire da remoto i due relè: sono quindi presenti due tasti che comandano l'attivazione dei relè stessi; è inoltre possibile specificare il relativo tempo di attivazione.

Infine, è presente una finestra di testo che mostra a video i dati in arrivo dal circuito.

**L'articolo completo  
del progetto è stato  
pubblicato su:**

**Elettronica In n. 80  
Giugno 2003**