

# Rapporti tecnici

# INGV

**GSM DC Rebooter**

**Sistema di reboot remoto via GSM per  
l'osservatorio geomagnetico di  
Lampedusa**

# 269



## **Editorial Board**

Andrea Tertulliani - Editor in Chief (INGV - RM1)  
Luigi Cucci (INGV - RM1)  
Nicola Pagliuca (INGV - RM1)  
Umberto Sciacca (INGV - RM1)  
Alessandro Settimi (INGV - RM2)  
Aldo Winkler (INGV - RM2)  
Salvatore Stramondo (INGV - CNT)  
Gaetano Zonno (INGV - MI)  
Viviana Castelli (INGV - BO)  
Marcello Vichi (INGV - BO)  
Sara Barsotti (INGV - PI)  
Mario Castellano (INGV - NA)  
Mauro Di Vito (INGV - NA)  
Raffaele Azzaro (INGV - CT)  
Rosa Anna Corsaro (INGV - CT)  
Mario Mattia (INGV - CT)  
Marcello Liotta (Seconda Università di Napoli, INGV - PA)

## **Segreteria di Redazione**

Francesca Di Stefano  
Tel. +39 06 51860068  
Fax +39 06 36915617  
Rossella Celi  
Tel. +39 095 7165851  
[redazionecen@ingv.it](mailto:redazionecen@ingv.it)



# Rapporti tecnici INGV

## **GSM DC REBOOTER SISTEMA DI REBOOT REMOTO VIA GSM PER L'OSSERVATORIO GEOMAGNETICO DI LAMPEDUSA**

Giovanni Benedetti, James A. Baskaradas

INGV (Istituto Nazionale di Geofisica e Vulcanologia, Sezione di Geomagnetismo, Aeronomia e Geofisica Ambientale)

# 269



## **Indice**

Introduzione	7
1. L'osservatorio geomagnetico di Lampedusa	7
2. Descrizione generale	8
3. Schema a blocchi	9
4. Schema elettrico e layout	10
5. Conclusioni	12
Appendice A	13
Appendice B	16



## Introduzione

Il dispositivo è un rebooter in corrente continua (fig. 1) che permette di compiere dei riavvii e dei controlli sulle macchine a esso collegate. È un dispositivo molto utile in tutti i casi in cui si debbano utilizzare strumenti in stazioni di misure dislocate in zone remote non presidiate. Una versione prototipale dello strumento è in funzione presso l'osservatorio geomagnetico di Lampedusa dal mese di Settembre 2010. Si può accedere allo strumento attraverso un collegamento telefonico (modem) e utilizzando un'interfaccia utente su una pagina web è possibile verificare lo stato (On-Off) della strumentazione e se necessario togliere le alimentazioni per permettere il reset di tutti gli strumenti collegati.

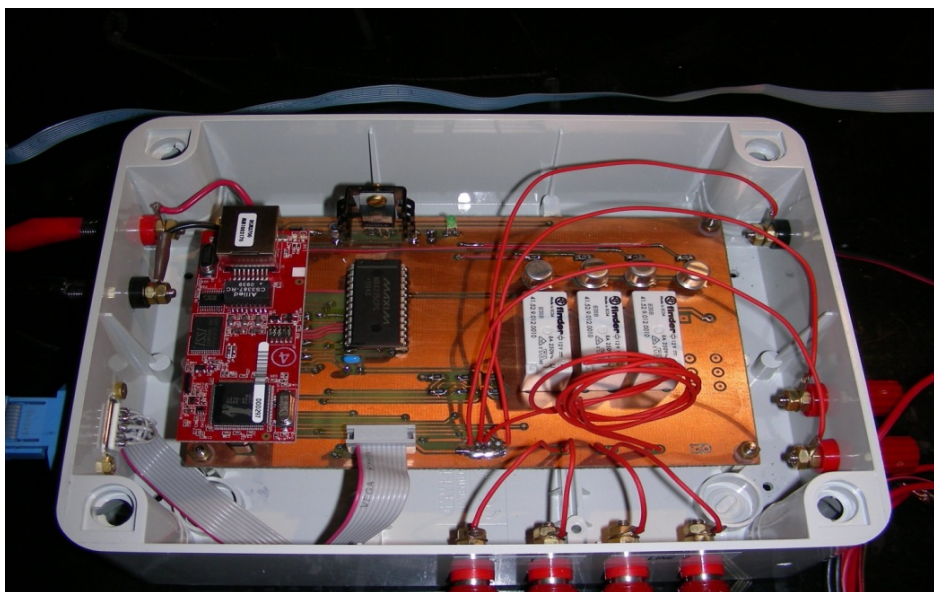


Figura 1. Il rebooter.

### 1. L'osservatorio geomagnetico di Lampedusa

L'osservatorio geomagnetico si trova nella punta occidentale dell'isola all'interno di una riserva naturale, dove il rumore elettromagnetico è molto basso ed ideale per le misure del campo geomagnetico terrestre. Consiste in una piccola struttura in pietra e legno (fig. 2), dove la corrente elettrica è fornita da un sistema ad energia solare e tutte le comunicazioni avvengono su rete GSM (unica rete disponibile dopo il satellitare). Le misure delle componenti X, Y, Z e F del campo magnetico terrestre vengono eseguite da un magnetometro scalare Overhauser (GEM-GSM90) e da un magnetometro vettoriale fluxgate (LEMI-018) mentre l'esatta temporizzazione dei dati avviene attraverso il sistema GPS. L'acquisizione è gestita da un software in grado di leggere i dati in modo continuo per entrambi gli strumenti. I dati sono poi memorizzati in due file giornalieri, il primo file contiene i dati al secondo mentre l'altro le medie al minuto delle componenti del campo geomagnetico.

Il software gira su un computer compatto di tipo "fitPC2" alimentato a corrente continua, dotato di tutte le porte necessarie alla lettura dei dati seriali provenienti dalla strumentazione. Quando funziona a regime ha un consumo di soli 500mA e, grazie alla sua struttura in alluminio, è adatto per lavorare vicino i sensori magnetici ([www.fit-pc.com/](http://www.fit-pc.com/)). Attraverso la rete GSM, i dati sono inviati quotidianamente alla sede romana dell'INGV e caricati sul server di geomagnetismo per la successiva analisi.

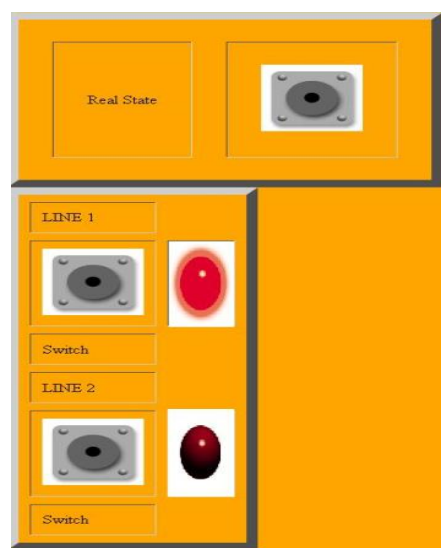
Tutti i dispositivi dell'osservatorio sono alimentati a 12 V forniti da delle batterie tampone e pannelli solari. Durante l'attività dell'osservatorio ci può essere la necessità di riavviare uno o più apparati di misura, infatti, sia il PC a corrente continua che i magnetometri (in particolare il magnetometro vettoriale) possono avere dei "crash" di sistema risolvibili solo attraverso un riavvio del dispositivo. Per questo si è pensato di realizzare un "rebooter" come quello descritto di seguito, in grado di gestire questi compiti da remoto a un costo di realizzazione molto basso.



**Figura 2.** La struttura dell'osservatorio.

## 2. Descrizione generale

Il “rebooter” è stato realizzato per funzionare in un osservatorio geomagnetico come quello di Lampedusa, dove esistono condizioni particolari per la vicinanza agli strumenti magnetici e il loro controllo. Attraverso di esso è possibile alimentare gli apparati di controllo e la strumentazione di misura di cui si necessita fare dei riavvii e dei controlli da remoto. Per la costruzione sono stati utilizzati materiali amagnetici come l'ottone, la plastica e l'alluminio. Il dispositivo consiste in un modem GSM esterno ed un cabinet avente i connettori di collegamento sulla parte laterale. All'interno c'è una scheda madre, dove alloggia un modulo programmabile compatto della *Rabbit Semiconductor* “Rabbit 3710”, programmato in modo da gestire diverse funzionalità come: il server http per il controllo remoto, la comunicazione attraverso un modem GSM e le porte digitali necessarie al controllo della strumentazione. Il modulo è una scheda a basso consumo e l'ambiente software e le librerie fornite permettono di costruire una piattaforma adatta allo sviluppo di applicazioni web server per il monitoraggio e il controllo remoto. Con 1 MB di memoria serial flash può immagazzinare pagine web, gestire account per utenti e diverse altre applicazioni web server. Il codice, scritto in un linguaggio simile al C, è visibile in appendice A e svolge tutte le funzioni necessarie al funzionamento del rebooter. Stabilendo quindi una connessione con il modem e inserendo l'indirizzo IP e le credenziali di accesso nel browser, si può accedere alla pagina web di controllo visibile in figura 3.



**Figura 3.** La pagina web di controllo.



I led indicano lo stato della strumentazione e i pulsanti sulla sinistra corrispondenti a “line1” e “line2” permettono di attivare o disattivare le alimentazioni e di eseguire quindi un reset sullo strumento qualora desiderato. Il pulsante “real state” fa un aggiornamento dello stato delle alimentazioni, infatti attraverso delle linee di feedback provenienti dagli strumenti è possibile controllare l’effettiva presenza della tensione elettrica di alimentazione, molto utile in condizioni remote per stabilire il funzionamento della stazione.

La pagina è un semplice file html (il codice è visibile in appendice B) con il minimo indispensabile al funzionamento per l’osservatorio, perché la connessione dati GSM è molto lenta e dei controlli supplementari potrebbero rendere difficoltoso il loading della pagina ad ogni set up.

La scheda madre prevede la possibilità di aggiungere altre due linee di controllo nel caso ce ne sia bisogno (figs. 5, 6).

### 3. Schema a blocchi

Com’è possibile vedere dallo schema (fig. 4) la tensione di alimentazione passa attraverso il rebooter al quale sono collegati tutti gli strumenti, come il PC a 12V per l’acquisizione dati e i magnetometri. Controllando la porta digitale si possono, in output pilotare i relays che effettuano lo switch delle linee mentre, in input si possono leggere i feedback. I relays sono normalmente “a riposo” e la corrente fluisce liberamente nei contatti portando l’alimentazione alle apparecchiature anche in caso di un malfunzionamento del rebooter. Quando si lancia il comando (premendo il pulsante sulla pagina web) il relay interessato apre i contatti e la tensione si blocca; premendo il pulsante una seconda volta il relay chiude i contatti e la corrente fluisce di nuovo.

Premendo il pulsante “real state” non si fa altro che leggere la porta digitale in ingresso, per cui se è presente una linea di tensione attiva proveniente dallo strumento, il led sarà acceso e ci possiamo assicurare di aver alimentato l’apparecchiatura.

La comunicazione modem avviene su RS232 attraverso un convertitore di livello.

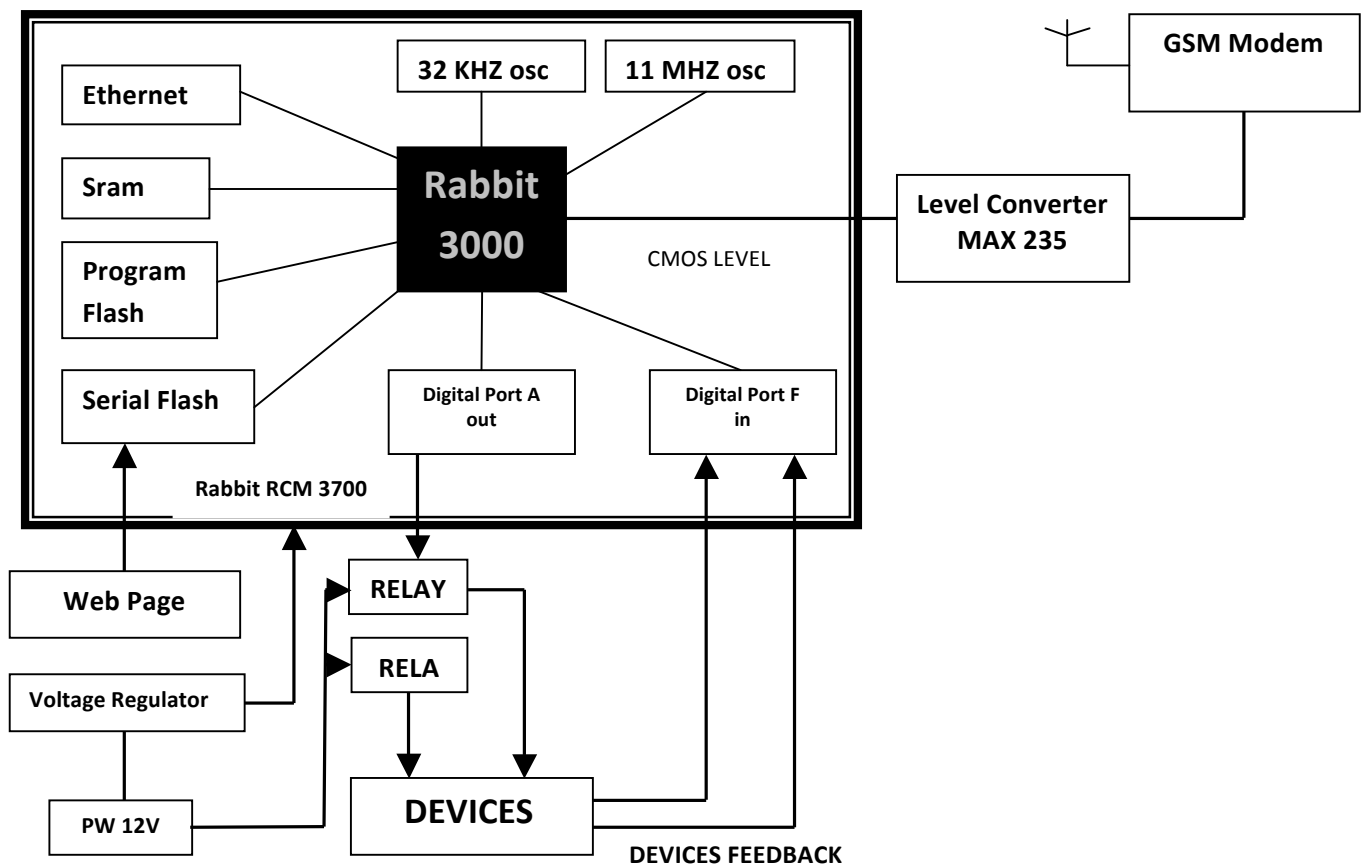


Figura 4. Schema a blocchi.

#### 4. Schema elettrico e layout

Lo schema elettrico (fig. 6) e layout (fig. 5) sono stati realizzati con ORCAD 9 e rappresentano la scheda madre del rebooter. Nella zona di alimentazione è stata inserita una modifica che sullo schema elettrico non compare, portando la 12V direttamente sui connettori esterni rosso/nero, dunque è possibile alimentare la strumentazione attraverso il rebooter. La comunicazione con la RS232 del modem, è adattata tramite un MAX235, per regolare i livelli di tensione della porta seriale ai livelli CMOS della scheda Rabbit. Il connettore dove convergono le linee di feedback, presenta dei partitori di tensione, che dividono il livello in ingresso, rendendolo adeguato agli ingressi tollerati dalla scheda Rabbit (fig. 6). Ci sono poi dei pull down resistors che evitano il floating dei bit utilizzati nella porta digitale quando nessun feedback è collegato. In fine gli switch elettronici sono formati da transistor 2N1711 che, entrando in saturazione con il livello logico inviato dalla porta digitale, pilotano i rispettivi relays a 12V. La scheda prevede alloggi e connessioni per altre due linee di controllo e feedback. Il PCB è stato realizzato con il metodo fotografico e sviluppo in bagni chimici nei laboratori INGV.

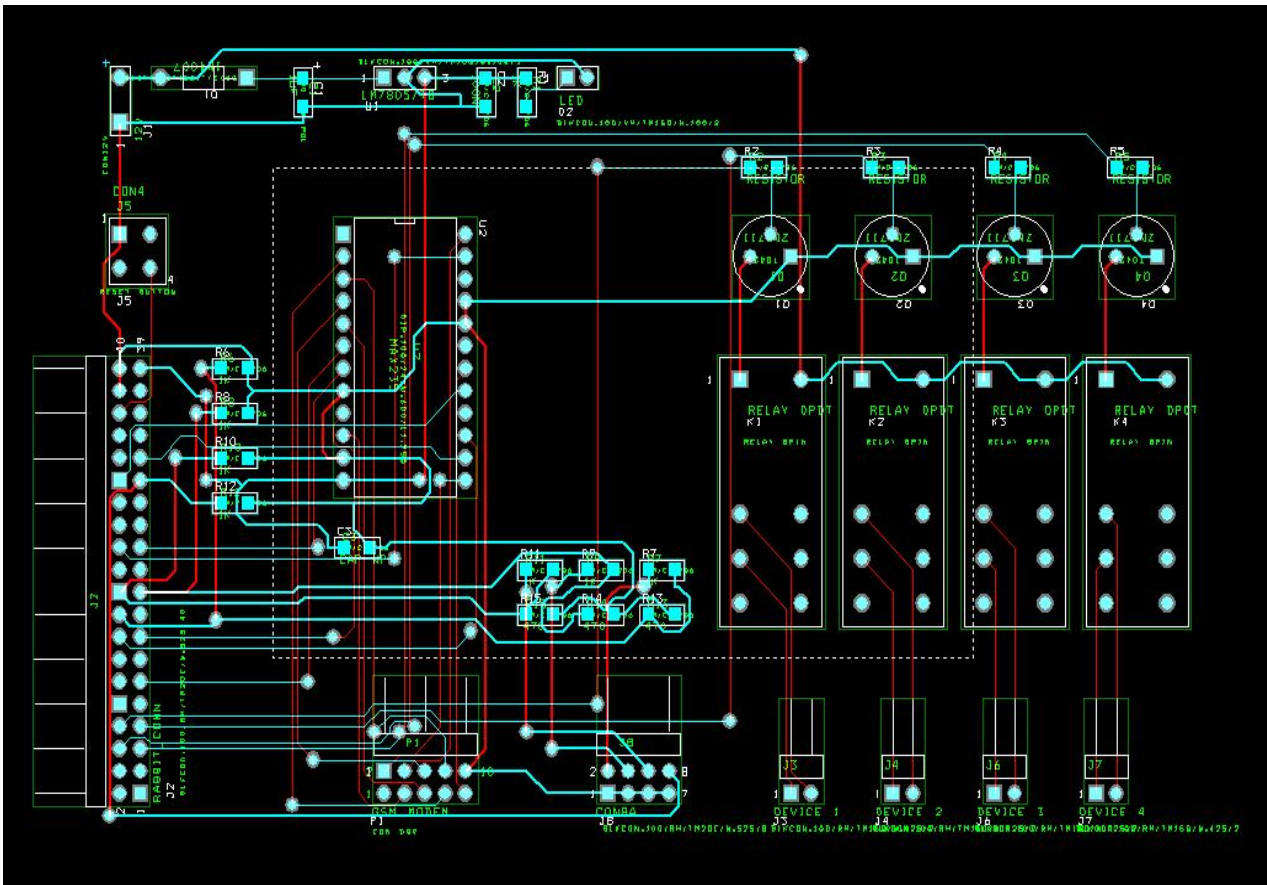


Figura 5. Layout.

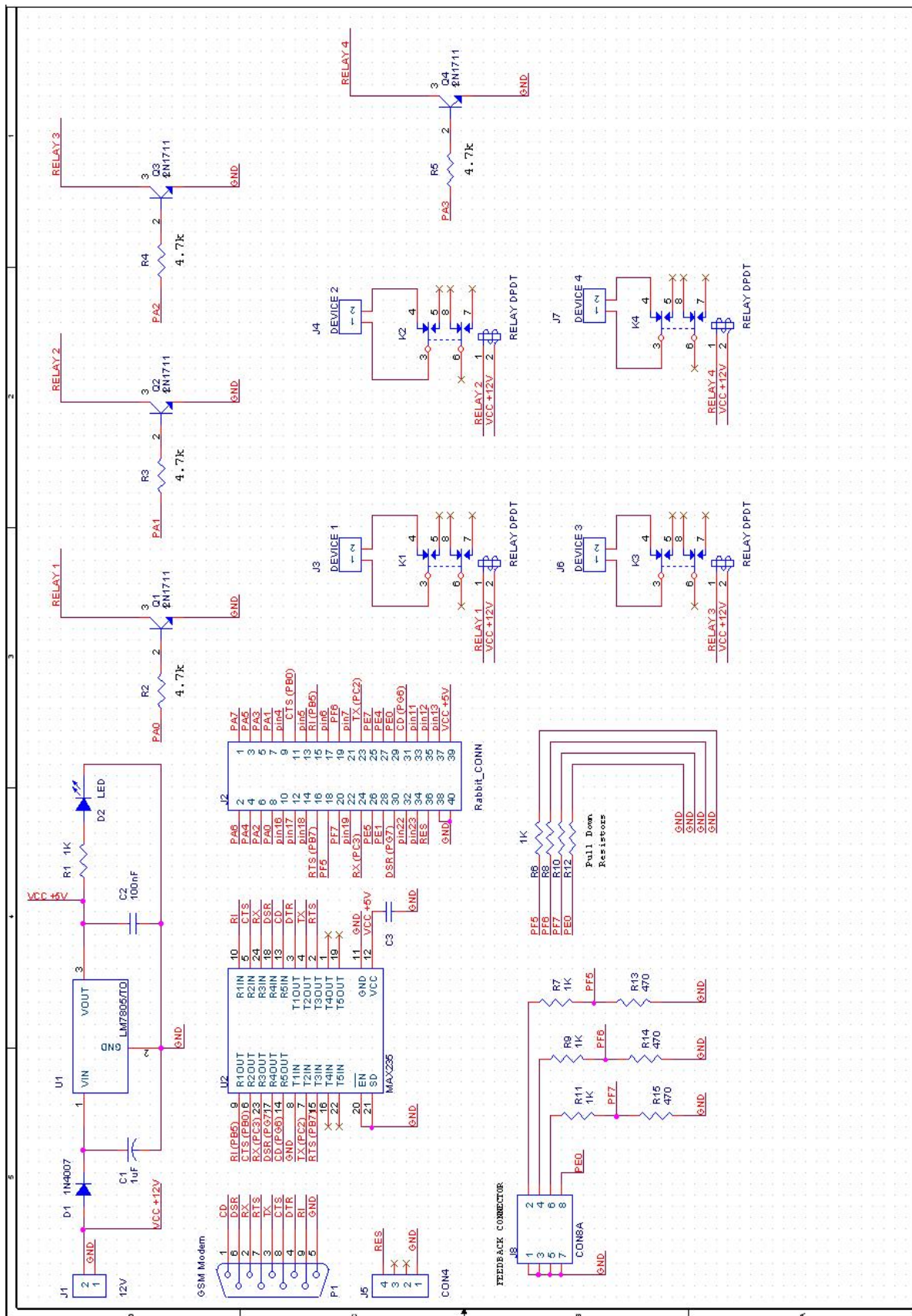


Figura 6. Schema elettrico.

## 5. Conclusioni

Il rebooter è in funzione nell'osservatorio di Lampedusa da circa tre anni. Si è dimostrata l'efficienza con il basso consumo energetico e l'affidabilità ad ogni collegamento remoto, che permette ogni qual volta ce ne sia la necessità, di riavviare tutti gli strumenti dell'osservatorio.

Per l'installazione è stata scritta un quick start guide (visibile in basso) che spiega all'utente come avviare il dispositivo e come effettuare i collegamenti da remoto attraverso il web server.

La scheda presenta la possibilità di montare un'espansione per altre due linee di switch con i rispettivi feedback, come illustrato nello schema elettrico.

I relays utilizzati sono predisposti per lavorare con la tensione di rete 220V, quindi l'intero apparato può essere configurato anche come switch remoto AC.

### **GSM DC Rebooter – Quick start guide**

Il dispositivo è già programmato e basterà seguire alcuni passi per un'installazione "plug and play".

Lo strumento consente di controllare da remoto due linee di alimentazione a 12V attraverso una pagina web.

Ogni linea ha un rispettivo feedback di tensione, in altre parole è sempre possibile avere un riscontro sulla presenza di alimentazione dal dispositivo da monitorare da remoto.

#### **Installazione:**

1. Alimentare il rebooter con la 12V, attraverso le boccole rosso/nero presenti sul pannello frontale denominate "12V IN".
2. Alimentare il modem GSM esterno con la 12V e collegare il cavo seriale del modem al connettore denominato "RS232" sul pannello frontale del rebooter; posizionare l'antenna del modem possibilmente all'esterno.
3. Collegare gli strumenti che s'intendono controllare alle prese di alimentazione a 12V presenti sul pannello laterale del rebooter, denominate rispettivamente "Line 1" e "Line 2".
4. Utilizzare una linea di tensione dallo strumento che s'intende controllare, e collegarla all'ingresso feedback presente sul pannello posteriore denominato "1" (feedback corrispondente alla Line 1) e "2" (feedback corrispondente alla Line 2).

A questo punto lo strumento è pronto per la prova di collegamento remoto.

## Appendice A

Codice sorgente:

```
////REMOTE CONTROL 12V ////
#class auto
////////GSM modem define////////
#define TCPCONFIG 2
#define USE_PPP_SERIAL 0x04
#define PPP_DEBUG
//Uncomment for PPP detail
#define PPP_VERBOSE
#define MODEM_DEBUG 1
#define CINBUFSIZE 31
#define COUTBUFSIZE 31
////////
#define TCP_BUF_SIZE 2048
#define HTTP_MAXSERVERS 2
#define MAX_TCP_SOCKET_BUFFERS 2
#define REDIRECTHOST PRIMARY_STATIC_IP
#define REDIRECTTO "http:"
#mmap xmem
#use "dertcp.lib"
#use "http.lib"
////////
#use "modem.lib"
#define TIMEZONE -8
#define LCP_TIMEOUT 5000
////////directory di collocazione web pages//
#ximport
"I:\BENEDETTI_GIOVANNI_LAB_WRK\RemoteControl 220V\newCODE 12V switch\webPage\webPage.shtml"
index_shtml
#ximport
"I:\BENEDETTI_GIOVANNI_LAB_WRK\RemoteControl 220V\newCODE 12V switch\webPage\rabbit1.gif"
rabbit1_gif
#ximport
"I:\BENEDETTI_GIOVANNI_LAB_WRK\RemoteControl 220V\newCODE 12V switch\webPage\ledon.gif"
ledon_gif
#ximport
"I:\BENEDETTI_GIOVANNI_LAB_WRK\RemoteControl 220V\newCODE 12V switch\webPage\ledoff.gif"
ledoff_gif
#ximport
"I:\BENEDETTI_GIOVANNI_LAB_WRK\RemoteControl 220V\newCODE 12V switch\webPage\button.gif"
button_gif
#ximport
"I:\BENEDETTI_GIOVANNI_LAB_WRK\RemoteControl 220V\newCODE 12V switch\remoteControl 12V switch.c" browseled_c
//////// MIMETABLE //////////
SSPEC_MIMETABLE_START
SSPEC_MIME_FUNC(".shtml", "text/html",
shtml_handler),
SSPEC_MIME(".html", "text/html"),
SSPEC_MIME(".gif", "image/gif"),
SSPEC_MIME(".cgi", "")
SSPEC_MIMETABLE_END
//variabili led per il feedback degli strumenti
char led1[15];
char led2[15];
//variabili alias x lo switch dei relays
char alias1[15];
char alias2[15];

//Feedback////////
int pf5,pf6
int ledtoggle(HttpState* state)
{
////////LINE 1////////
pf5=BitRdPortI(0x038 , 5); //data register

if(pf5==0)
strcpy(led1,"ledoff.gif");
else
strcpy(led1,"ledon.gif");
////////LINE 2////////
pf6=BitRdPortI(0x038 , 6);
if(pf6==0)
strcpy(led2,"ledoff.gif");
else
strcpy(led2,"ledon.gif");

cgi_redirectto(state,REDIRECTTO); //aggiornameto
pagina
return 0;
}
////////RELAYS CONTROL////////
////////LINE 1////////PAO
int led12toggle(HttpState* state) {
if (strcmp(alias1,"ledon.gif")==0){
strcpy(alias1,"ledoff.gif");
BitWrPortI(0x030, &PFDRShadow, 1, 0) }
else{
strcpy(alias1,"ledon.gif");
BitWrPortI(0x030, &PFDRShadow, 0, 0);
}

cgi_redirectto(state,REDIRECTTO);
//aggiornamento pagina..
return 0;
}
////////LINE 2////////PAI
int led22toggle(HttpState* state)
{
if (strcmp(alias2,"ledon.gif")==0){
strcpy(alias2,"ledoff.gif");
BitWrPortI(0x030, &PFDRShadow, 1, 1);
}
}
else{
strcpy(alias2,"ledon.gif");
BitWrPortI(0x030, &PFDRShadow, 0, 1);
}

cgi_redirectto(state,REDIRECTTO);
return 0;
}

SSPEC_RESOURCETABLE_START
SSPEC_RESOURCE_XMEMFILE("",
index_shtml),
SSPEC_RESOURCE_XMEMFILE("/index.shtml", index_shtml),
SSPEC_RESOURCE_XMEMFILE("/rabbit1.gif", rabbit1_gif),
SSPEC_RESOURCE_XMEMFILE("/ledon.gif", ledon_gif),
SSPEC_RESOURCE_XMEMFILE("/ledoff.gif", ledoff_gif),
SSPEC_RESOURCE_XMEMFILE("/button.gif", button_gif),
SSPEC_RESOURCE_XMEMFILE("browseled.c", browseled_c),
SSPEC_RESOURCE_ROOTVAR("led1",
led1, PTR16, "%s"),
SSPEC_RESOURCE_ROOTVAR("led2",
led2, PTR16, "%s"),
//FEEDBACK-lettura su parallel port F ed E in ingresso
(BitRead) per la presenza di un valore di tensione in
ingresso...vedere digiIN..
```

```

        SSPEC_RESOURCE_FUNCTION("/ledtog.cg
i", ledtoggle), //sono state aggiunte tutte le funzioni
necessarie nella resource table, e nella pagina.shtml
//RELAYS-CONTROL-scrittura su parallel port A per lo
switch dei relays
        SSPEC_RESOURCE_FUNCTION("/led12tog.cgi",
led12toggle),
        SSPEC_RESOURCE_FUNCTION("/led22tog.cgi",
led22toggle),
SSPEC_RESOURCE_END
//////////MAIN//////////
main()
{
int user;
int page1;
int page2;
//////////
auto unsigned long t;
auto char buffer[100];
//////////
//Chiamo le funzioni sui led per il feedback degli
strumenti all'inizio del main per un controllo iniziale
//sullo stato degli strumenti ON/OFF...quindi all'apertura
della pagina i led indicheranno lo stato attuale della
strumentazione
ledtoggle();
//Paralle Port A INIZIALIZZATION...senza boardInit
//stringhe prese da "boardInit"...inizializzo solo ciò che
necessario
WrPortI(PADR, &PADRShadow, 0x00); //set to
output all low
WrPortI(SPCR, &SPCRShadow, 0x84); //sets all bits
to output
sock_init();
http_init();
//--Autenticazione su webserver!--
http_setauthentication(HTTP_DIGEST_AUTH);
user = sauth_adduser("giovanni", "2345",
SERVER_HTTP);
page1 = sspec_addxmemfile("/", index.shtml,
SERVER_HTTP);
sspec_adduser(page1, user);
sspec_setrealm(page1, "Admin");

page2 = sspec_addxmemfile("index.shtml",
index.shtml, SERVER_HTTP);
sspec_adduser(page2, user);
sspec_setrealm(page2, "Admin");

sspec_addxmemfile("rabbit1.gif", rabbit1.gif,
SERVER_HTTP);
sspec_addxmemfile("ledon.gif", ledon.gif,
SERVER_HTTP);
sspec_addxmemfile("ledoff.gif", ledoff.gif,
SERVER_HTTP);
sspec_addxmemfile("button.gif", button.gif,
SERVER_HTTP);
sspec_addxmemfile("browseled.c", browseled_c,
SERVER_HTTP);

```

```

//--Fine Autenticazione!--

tcp_reserveport(80);

while(1)
{
//make sure modem RTS
is active low
BitWrPortI(PBDR, &PBDRShadow, 0, 7);

ifconfig(IF_PPP2,
IFS_PPP_INIT,
IFS_PPP_SPEED, 9600L, //19200L
IFS_PPP_USEMODEM, 1,
IFS_PPP_MODEMESCAPE, 1,
IFS_PPP_SENDEXPECT,
"ATS0=1
&CONNECT", IFS_PPP_HANGUP,
"ATH", IFS_IPADDR, aton("10.1.10.1"),
IFS_PPP_ACCEPTIP, 0,
IFS_PPP_SETREMOTEIP, aton("10.1.10.2"),
IFS_PPP_ACCEPTDNS, 0,
IFS_PPP_LOCALAUTH, "rabbit", "carrots",
IFS_UP, IFS_END);
while(ifpending(IF_PPP2) % 2)
{
tcp_tick(NULL); //wait for PPP to come up
}

if(ifstatus(IF_PPP2))
{
printf("PPP established\n");
printf("IP address is %s\n", inet_ntoa( buffer,
gethostid()));

http_init();
tcp_reserveport(80);

while (ifstatus(IF_PPP2) && ModemConnected()) {
http_handler();
}
else
{
printf("PPP failed\n");
}
}

ifconfig(IF_PPP2, IFS_DOWN, IFS_END);
while(ifstatus(IF_PPP2))
{
tcp_tick(NULL); //wait for PPP to terminate
}
}

}

#nodebug

```

## Descrizione del codice e funzionamento

Il codice è scritto in Dynamic C, linguaggio proprietario della Rabbit Semiconductor, fondamentalmente molto simile al C classico con l'aggiunta di funzioni dedicate al dispositivo e la possibilità all'accesso e modifica dei suoi registri.

Inizialmente sono dichiarate delle "define" indicando i tipi di protocolli da usare, la porta seriale B per il modem, l'uso di librerie come quella per il funzionamento dell'http server e del modem. Vengono fatte delle "import" di codice html ed immagini .gif per la visualizzazione della pagina web di controllo. C'è poi una "mimetable" dove vengono dichiarati i tipi di file usati nell'interazione con l'http server. Si dichiarano le variabili per il funzionamento dei led, per pilotare i pulsanti e per i feedback.

C'è poi la funzione "ledtoggle" per leggere i feedback degli strumenti collegati. Quando viene premuto il pulsante "Real State" nella web page, si richiama questa funzione in cui sono stati scelti due bit della porta digitale F per la lettura, configurata come input (pf5, pf6). A questo punto viene letto il dataRegister della porta: se è presente un livello di tensione alto negli ingressi il relativo led sarà acceso, altrimenti sarà spento.

Ci sono ora le funzioni per il reset degli strumenti. Qui sono stati scelti due bit della porta digitale A (pa0, pa1) configurati come output. Quando nella web Page di controllo viene premuto il pulsante relativo ad una linea da switchare, si richiama la funzione "led12toggle" oppure "led22toggle" che invierà in uscita sul bit in questione un livello alto o basso. Questi due bit sono collegati a degli interruttori elettronici che provvederanno a switchare le linee 1 e 2 degli strumenti. Premendo di nuovo il pulsante "real state" si farà un refresh sui led che indicano lo stato attuale della strumentazione collegata.

Nella "Resource Table" vengono inseriti tutti i parametri e le funzioni usate per l'interazione del codice con la web page.

La funzione "main" inizia con alcune dichiarazioni e la chiamata di funzione da codice per "ledtoggle"; quindi all'apertura della web page, viene fatto un check sui feedback collegati agli ingressi che indicheranno lo stato della strumentazione al primo accesso al dispositivo. È poi inizializzata la porta digitale A come output, impostando tutti i bit a zero, in modo da avere livelli bassi che non eccitino i relays senza che gli venga dato il comando. Viene poi definita un'autenticazione sull'accesso alla web page. All'interno del loop c'è tutto il funzionamento del modem GSM attraverso la libreria PPP caricata; viene impostato un nome utente e password per stabilire la connessione, è definito l'indirizzo IP al quale collegarsi per aprire la web page di controllo e l'IP dato all'utente remoto. Viene quindi avviato l'http server e la comunicazione è attiva.

## Appendice B

Codice pagina HTML:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD W3 HTML//EN">
<HTML>
<HEAD>

    <TITLE>Remote Contol 220V</TITLE>
</HEAD>
<body topmargin="1" leftmargin="0" marginwidth="0"
marginheight="0"
    bgcolor="#FFA500" link="#009966"
vlink="#FFCC00" alink="#006666">

<body style="
background-image: url(img.gif);
background-repeat: repeat;"
>

<img SRC="rabbit1.gif">
<BR>

<TABLE BORDER="10" CELSPACING="30"
CELLPADDING="30">
<TR>
<TD> Real State </TD>
<TD> <A HREF="/ledtog.cgi"> <img SRC="button.gif"> </A>
</TD>
</TR>
</TABLE>

<TABLE BORDER="10" CELSPACING="10"
CELLPADDING="10">

    <TR>

        <TD> LINE 1 </TD>
```

```
</TR>

<TR>

    <TD> <A HREF="/led12tog.cgi"> <img SRC="button.gif"> </A>
</TD>
    <TD> <img SRC="<!--#echo var="led1"-->"> </TD>
</TR>

<TR>
<TD> Switch </TD>
</TR>

    <TR>
<TD> LINE 2 </TD>
</TR>

<TR>

    <TD> <A HREF="/led22tog.cgi"> <img SRC="button.gif">
</A> </TD>
    <TD> <img SRC="<!--#echo var="led2"-->"> </TD>
</TR>

<TR>
<TD> Switch </TD>
</TR>

</TABLE>

</BODY>
```



**Coordinamento editoriale e impaginazione**

Centro Editoriale Nazionale | INGV

**Progetto grafico e redazionale**

Daniela Riposati | Laboratorio Grafica e Immagini | INGV

© 2014 INGV Istituto Nazionale di Geofisica e Vulcanologia

Via di Vigna Murata, 605

00143 Roma

Tel. +39 06518601 Fax +39 065041181

**<http://www.ingv.it>**



**Istituto Nazionale di Geofisica e Vulcanologia**