

GMM AM08

grifo® Mini Modulo AT mega 8L

MANUALE TECNICO



grifo®

ITALIAN TECHNOLOGY

Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY

E-mail: grifo@grifo.it

<http://www.grifo.it>

<http://www.grifo.com>

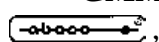
Tel. +39 051 892.052 (r.a.) FAX: +39 051 893.661



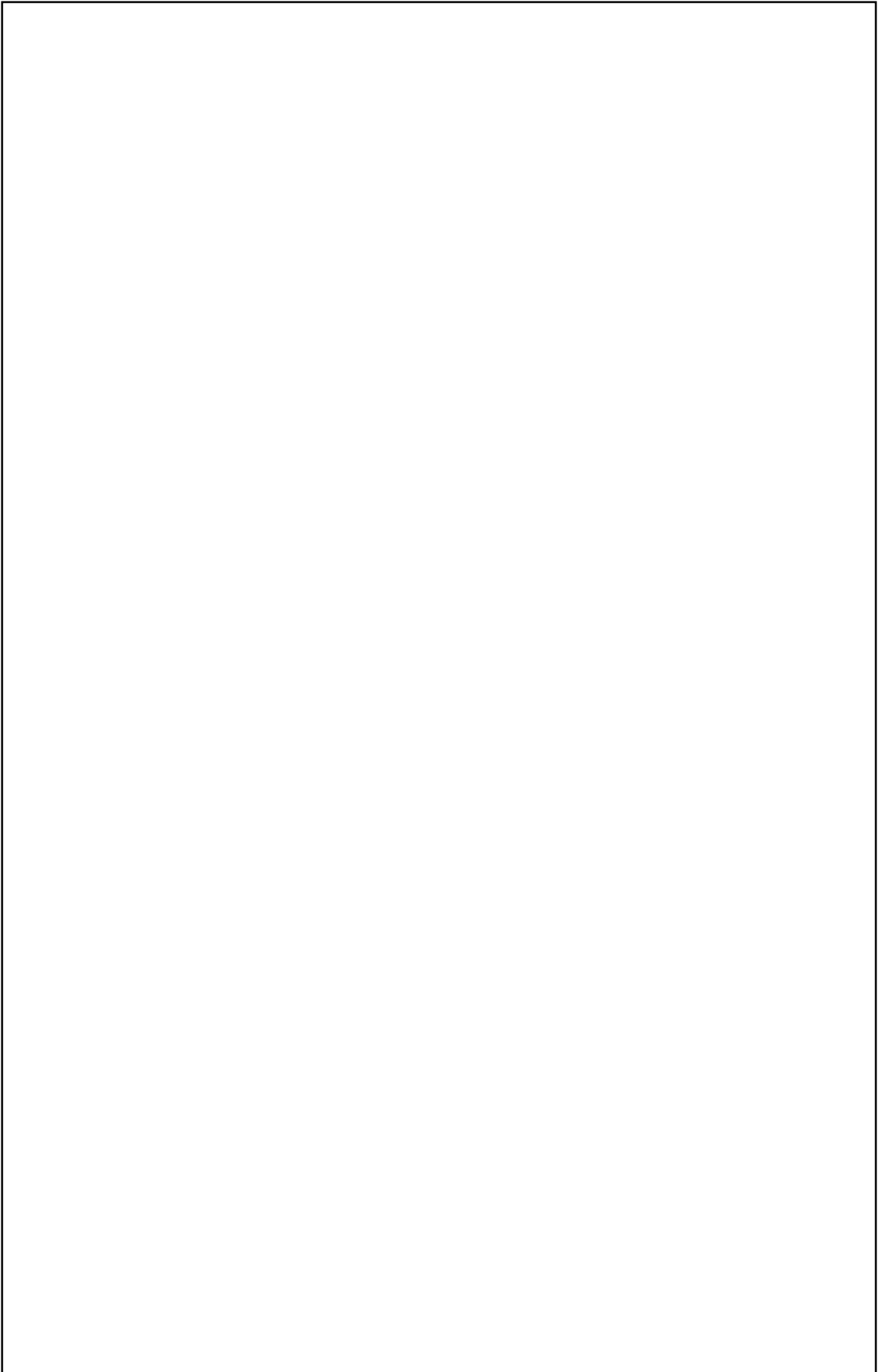
GMM AM08

Rel. 3.10

Edizione 15 Febbraio 2006



, GPC®, **grifo®**, sono marchi registrati della ditta **grifo®**



GMM AM08

grifo® Mini Modulo AT mega 8L

MANUALE TECNICO

Zoccolo maschio 28 piedini dual in line a passo 100 mils, largo 600 mils; ridottissimo ingombro: 20,7 x 38,7 x 12,8 mm; Unica alimentazione da +3,0 a +5,0 Vdc (l'assorbimento puo' variare in base ai collegamenti del modulo); disponibilita' di **Idle Mode** e **Power Down mode**; microcontrollore **Atmel ATmega 8L** (AVR RISC) con quarzo da 7,3728 MHz; 8K bytes FLASH per codice, fino a 1K bytes FLASH per **Boot Loader** opzionale, 1K bytes RAM per dati, 512 bytes EEPROM per dati; 6 canali di A/D converter con 10 bit di risoluzione e 2 canali di A/D converter con 8 bit di risoluzione a 5µsec per conversione; 18 sorgenti di **Interrupt**; 3 **Timer Counter** a 8 o 16 bits con funzionalita' di **PWM**, comparazione, cattura, ecc.; 23 linee di I/O digitale collegate al connettore; linea seriale hardware con **Baud Rate** programmabile fino a 115200 Baud, bufferata in RS 232 o TTL; interfaccia di comunicazione sincrona a due fili; interfaccia **SPI**; circuiteria di **Reset**; **Watch Dog** programmabile da 16 ms fino a 2 s; dip switch di configurazione a 4 vie; LED di segnalazione gestito via software tramite una linea di I/O digitale; possibilita' di gestione della FLASH ed EEPROM interna in modalita' **In System Programming**, ovvero con modulo gia' montato, sfruttando la modalita' di programmazione **ISP**; software gratuito per PC, di supporto alla programmazione **ISP** con cui scaricare il codice generato nella FLASH di bordo; vasta disponibilita' di software di sviluppo quali: **Assemblatori**; compilatori C (ICC AVR, DDS Micro C51); compilatori BASIC (BASCOM AVR); ecc.; ricca serie di programmi dimostrativi ed esempi di utilizzo forniti sotto forma di sorgenti ampiamente commentati, per i vari ambienti di sviluppo.

grifo®

ITALIAN TECHNOLOGY

Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY

E-mail: grifo@grifo.it

<http://www.grifo.it>

Tel. +39 051 892.052 (r.a.) FAX: +39 051 893.661

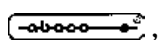
<http://www.grifo.com>



GMM AM08

Rel. 3.10

Edizione 15 Febbraio 2006



, GPC®, grifo®, sono marchi registrati della ditta grifo®

Nessuna parte del presente manuale può essere riprodotta, trasmessa, trascritta, memorizzata in un archivio o tradotta in altre lingue, con qualunque forma o mezzo, sia esso elettronico, meccanico, magnetico ottico, chimico, manuale, senza il permesso scritto della **grifo®**.

IMPORTANTE

Tutte le informazioni contenute sul presente manuale sono state accuratamente verificate, ciononostante **grifo®** non si assume nessuna responsabilità per danni, diretti o indiretti, a cose e/o persone derivanti da errori, omissioni o dall'uso del presente manuale, del software o dell' hardware ad esso associato.

grifo® altresì si riserva il diritto di modificare il contenuto e la veste di questo manuale senza alcun preavviso, con l' intento di offrire un prodotto sempre migliore, senza che questo rappresenti un obbligo per **grifo®**.

Per le informazioni specifiche dei componenti utilizzati sui nostri prodotti, l'utente deve fare riferimento agli specifici Data Book delle case costruttrici o delle seconde sorgenti.

LEGENDA SIMBOLI

Nel presente manuale possono comparire i seguenti simboli:



Attenzione: Pericolo generico



Attenzione: Pericolo di alta tensione



Attenzione: Dispositivo sensibile alle cariche elettrostatiche

Marchi Registrati



, GPC®, **grifo®** : sono marchi registrati della **grifo®**.

Altre marche o nomi di prodotti sono marchi registrati dei rispettivi proprietari.

INDICE GENERALE

INTRODUZIONE	1
VERSIONE SCHEDA.....	3
INFORMAZIONI GENERALI	4
LINEE DI I/O DIGITALE TTL	6
A/D CONVERTER	6
DIP SWITCH	6
WATCH DOG	6
INTERFACCIA SPI	6
LINEA I ² C BUS	8
COMUNICAZIONE SERIALE	8
TIMER COUNTER E PWM	8
DISPOSITIVI DI MEMORIA	9
CLOCK	9
CARATTERISTICHE TECNICHE	10
CARATTERISTICHE GENERALI	10
CARATTERISTICHE FISICHE	10
CARATTERISTICHE ELETTRICHE	11
INSTALLAZIONE	12
SEGNALAZIONI VISIVE	12
CONNESSIONI	12
CN1 - CONNETTORE CON SEGNALI DEL MINI MODULO	12
CONFIGURAZIONE MINI MODULO	14
SELEZIONE COMUNICAZIONE SERIALE	16
INTERFACCIAMENTO CONNETTORI CON IL CAMPO	18
ALIMENTAZIONE	18
INTERRUPTS	19
SCHEDE DI SUPPORTO	20
UTILIZZO CON IL MODULO BLOCK GMB HR84	20
UTILIZZO CON LA SCHEDA GMM TST 2	22
COME INIZIARE	24
RICOMPILAZIONE CON BASCOM AVR	28
RICOMPILAZIONE CON ICC AVR	30
DESCRIZIONE SOFTWARE	32
DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO	34
LED DI ATTIVITA'	34
PERIFERICHE DELLA CPU	34

BIBLIOGRAFIA	36
APPENDICE A: DESCRIZIONE COMPONENTI DI BORDO	A-1
ATmega8L	A-1
APPENDICE B: PROGRAMMAZIONE ISP CON GMM TST 2	B-1
OPERAZIONI COMUNI PER TUTTI I PROGRAMMATORI	B-1
PROGRAMMAZIONE CON grifo® MP-AVR 51+ ; grifo® UEP 49	B-3
PROGRAMMAZIONE CON EQUINOX EPSILON5	B-5
PROGRAMMAZIONE CON ATMEL AVR ISP	B-7
PROGRAMMAZIONE CON PONY PROG	B-9
APPENDICE C: SCHEMA ELETTRICO GMM TST 2	C-1
APPENDICE D: INDICE ANALITICO	D-1

INDICE DELLE FIGURE

FIGURA 1: POSIZIONE DEL NUMERO DI REVISIONE	3
FIGURA 2: SCHEMA A BLOCCHI	7
FIGURA 3: FOTO GMM AM08	9
FIGURA 4: TABELLA DELLE SEGNALAZIONI VISIVE	12
FIGURA 5: CN1 - ZOCCOLO CON SEGNALI DEL MINI MODULO	13
FIGURA 6: TABELLA DIP SWITCH DSW1 A 4 VIE	14
FIGURA 7: PIANTA COMPONENTI (LATO COMPONENTI)	15
FIGURA 8: DISPOSIZIONE LEDs, DIP SWITCHES, ECC.	15
FIGURA 9: ESEMPIO COLLEGAMENTO SERIALE RS 232	17
FIGURA 10: ESEMPIO COLLEGAMENTO SERIALE TTL	17
FIGURA 11: FOTO DI GMB HR84 E GMM AM08 CON E SENZA CONTENITORE	21
FIGURA 12: SCHEDA DI SPERIMENTAZIONE GMM TST 2 CON GMM AM08 E AVR ISP	23
FIGURA 13: COLLEGAMENTO SERIALE RS 232 TRA GMM AM08 E PC	24
FIGURA 14: TABELLA ESEMPI.....	25
FIGURA 15: BOOT LOADER UTILITY	27
FIGURA 16: CARICAMENTO SORGENTE CON BASCOM AVR	28
FIGURA 17: CONFIGURAZIONE COMPILATORE BASCOM AVR	29
FIGURA 18: COMPILAZIONE CON BASCOM AVR	29
FIGURA 19: CARICAMENTO FILE DI PROGETTO CON ICC AVR	30
FIGURA 20: COMPILAZIONE CON ICC AVR	30
FIGURA 21: VISTA DALL'ALTO DELLA GMM AM08	33
FIGURA 22: SCHEMA DELLE POSSIBILI CONNESSIONI	35
FIGURA 23: SCHEMA DELLE CONNESSIONI A GMM TST 2 CON GMM AM08	37
FIGURA B-1: SCHEMA DELL'INTERFACCIA TRA GMM TST 2 E MP-AVR 51+	B-2
FIGURA B-2: SCHEMA DELL'INTERFACCIA TRA GMM TST 2 E UEP 49	B-2
FIGURA B-3: SCHEMA DI CONNESSIONE DIRETTA TRA GMM AM08 E SEP 40	B-3
FIGURA B-4: SCHEMA DI CONNESSIONE DIRETTA TRA GMM AM08 E UEP 49	B-3
FIGURA B-5: SELEZIONE DEL COMPONENTE CON PG4UW	B-4
FIGURA B-6: SCHEMA DI CONNESSIONE DIRETTA TRA GMM AM08 ED EPSILON5 o AVR ISP .	B-5
FIGURA B-7: SCHEMA DI CONNESSIONE DIRETTA TRA GMM AM08 ED MP AVR-51+	B-5
FIGURA B-8: CONFIGURAZIONE DEL COMPONENTE CON PG4UW	B-6
FIGURA B-9: CONFIGURAZIONE DEL PROGRAMMATORE PG4UW	B-6
FIGURA B-10: CONFIGURAZIONE PROGETTO CON EQTOOLS	B-8
FIGURA B-11: SELEZIONE CPU CON AVR STUDIO	B-10
FIGURA B-12: CONFIGURAZIONE CPU CON AVR STUDIO	B-10
FIGURA B-13: CONFIGURAZIONE AVR ISP	B-10
FIGURA B-14: LOGO DEL PONYPROG	B-11
FIGURA B-15: SELEZIONE MICROCONTROLLORE CON PONYPROG	B-12
FIGURA B-16: CONFIGURAZIONE MICROCONTROLLORE CON PONYPROG	B-12
FIGURA B-17: AVVIO DELLA PROGRAMMAZIONE CON PONYPROG	B-12
FIGURA C-1: SCHEMA ELETTRICO GMM TST 2 (1 DI 3)	C-1
FIGURA C-2: SCHEMA ELETTRICO GMM TST 2 (2 DI 3)	C-2
FIGURA C-3: SCHEMA ELETTRICO GMM TST 2 (3 DI 3)	C-3

INTRODUZIONE

L'uso di questi dispositivi è rivolto - IN VIA ESCLUSIVA - a personale specializzato.
Questo prodotto non è un **componente di sicurezza** così come definito dalla direttiva **98-73/CE**.



I pin del Mini Modulo non sono dotati di protezione contro le cariche elettrostatiche. Esiste un collegamento diretto tra i pin del Mini Modulo e i rispettivi pin del microcontrollore. Il Mini Modulo è sensibile ai fenomeni ESD.

Il personale che maneggia i Mini Moduli è invitato a prendere tutte le precauzioni necessarie per evitare i possibili danni che potrebbero derivare dalle cariche elettrostatiche.

Scopo di questo manuale è la trasmissione delle informazioni necessarie all'uso competente e sicuro dei prodotti. Esse sono il frutto di un'elaborazione continua e sistematica di dati e prove tecniche registrate e validate dal Costruttore, in attuazione alle procedure interne di sicurezza e qualità dell'informazione.

I dati di seguito riportati sono destinati - IN VIA ESCLUSIVA - ad un utenza specializzata, in grado di interagire con i prodotti in condizioni di sicurezza per le persone, per la macchina e per l'ambiente, interpretando un'elementare diagnostica dei guasti e delle condizioni di funzionamento anomale e compiendo semplici operazioni di verifica funzionale, nel pieno rispetto delle norme di sicurezza e salute vigenti.

Le informazioni riguardanti installazione, montaggio, smontaggio, manutenzione, aggiustaggio, riparazione ed installazione di eventuali accessori, dispositivi ed attrezzature, sono destinate - e quindi eseguibili - sempre ed in via esclusiva da personale specializzato avvertito ed istruito, o direttamente dall'ASSISTENZA TECNICA AUTORIZZATA, nel pieno rispetto delle raccomandazioni trasmesse dal costruttore e delle norme di sicurezza e salute vigenti.

I dispositivi non possono essere utilizzati all'aperto. Si deve sempre provvedere ad inserire i moduli all'interno di un contenitore a norme di sicurezza che rispetti le vigenti normative. La protezione di questo contenitore non si deve limitare ai soli agenti atmosferici, bensì anche a quelli meccanici, elettrici, magnetici, ecc.

Per un corretto rapporto coi prodotti, é necessario garantire leggibilità e conservazione del manuale, anche per futuri riferimenti. In caso di deterioramento o più semplicemente per ragioni di approfondimento tecnico ed operativo, consultare direttamente l'Assistenza Tecnica autorizzata.

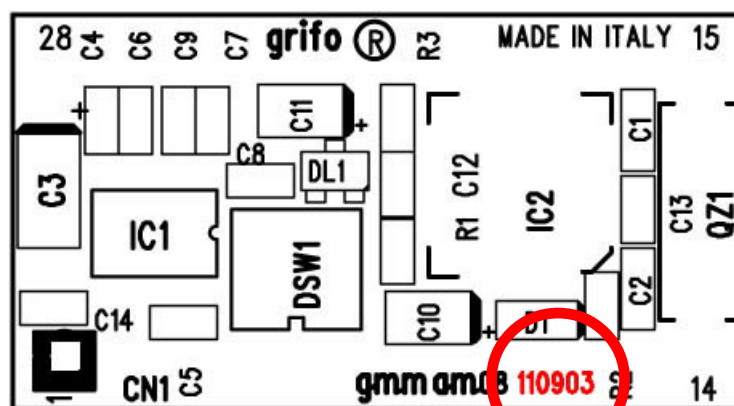
Al fine di non incontrare problemi nell'uso di tali dispositivi, é conveniente che l'utente - PRIMA DI COMINCIARE AD OPERARE - legga con attenzione tutte le informazioni contenute in questo manuale. In una seconda fase, per rintracciare più facilmente le informazioni necessarie, si può fare riferimento all'indice generale e all'indice analitico, posti rispettivamente all'inizio ed alla fine del manuale.

VERSIONE SCHEDA

Il presente manuale è riferito alla scheda **GMM AM08** revisione **110903**.

La validità delle informazioni riportate è quindi subordinata al numero di revisione della scheda in uso e l'utente deve quindi sempre verificarne la giusta corrispondenza.

Tali versioni sono sempre riportate sullo stampato in più punti e la seguente figura illustra la posizione più facilmente accessibile.



NUMERO DI REVISIONE DELLO STAMPATO

FIGURA 1: POSIZIONE DEL NUMERO DI REVISIONE

INFORMAZIONI GENERALI

Il modulo **GMM AM08** (**grifo®** Mini Modulo con ATmega8L), è basato sul microcontrollore **Atmel ATmega8L**, un potente e completo sistema on-chip dotato di **CPU**, **memoria integrata** sia per il codice da eseguire sia per i dati, **A/D converter**, **watch dog**, **interrupts**, linee di I/O digitali TTL, una linea seriale hardware, **timer/counter** dedicati con capacità di capture/compare e **PWM**, ecc. Il modulo ha già montati nella sua ridottissima area alcuni componenti che servono a valorizzare le principali caratteristiche del microcontrollore e, oltre a questo, monta ulteriori circuiterie che ne integrano le funzionalità, come quella che genera il segnale di reset.

Le **possibili applicazioni** dei moduli **GMM AM08** sono innumerevoli.

Si può citare, ad esempio, il **funzionamento** come piccoli **nodi intelligenti** con funzionalità locali come il controllo con algoritmi PID di temperature, motori, valvole o come **sistemi a logica distribuita** tipo robot, automazioni su macchine di produzione in linea, automazioni di fabbriche di grosse dimensioni. Infine la **teleacquisizione** e il **telecontrollo** su medio brevi distanze, la **conversione** tra **SPI** e **seriale asincrona** o linea **SPI** e **I²C BUS** (interfaccia seriale sincrona a due fili) e l'**automazione domestica** (accensione e spegnimento luci, controllo riscaldamento e condizionamento, supervisione elettrodomestici e servizi elettrici, sistemi di sorveglianza e controllo accesso).

Da non dimenticare il **settore didattico**; infatti la **GMM AM08** offre la possibilità di apprendere il funzionamento di un microcontrollore **RISC** con core famiglia AVR e di sviluppare le sue applicazioni canoniche ad un costo veramente basso.

A questo scopo è ideale la scheda di supporto **GMM TST 2**, che permette di collegare immediatamente una porta seriale RS 232 per il collegamento con un PC ed una tensione di riferimento per l'A/D Converter.

Inoltre fornisce di serie una tastiera a matrice da 16 tasti ed un display LCD da 2 righe di 20 caratteri l'una, permettendo l'implementazione di un emulatore terminale e lo studio di soluzioni per interfaccia utente.

La **GMM TST 2** è inoltre predisposta per la programmazione ISP del mini modulo tramite la comoda e veloce interfaccia **AVR ISP**, che permette di automatizzare e quindi di velocizzare al massimo il ciclo di riprogrammazione della FLASH interna, ed anche tramite il software **PonyProg** della italiana **LancOS**, che permette di riprogrammare la FLASH tramite una semplice porta seriale RS 232 (ad esempio, quella del PC).

Il modulo **GMM AM08** viene fornito con un programma di Boot Loader preinstallato, che permette di riprogrammare la FLASH e la EEPROM tramite una semplice porta seriale RS 232 (ad esempio, quella del PC).

In tutti i casi di scarso tempo di sviluppo: **l'utente può avere il suo prototipo** o addirittura il prodotto finito **nel giro di una settimana**.

Le caratteristiche di massima del modulo **GMM AM08** sono:

- **Zoccolo** maschio **28 piedini** dual in line a passo 100 mils, largo 600 mils
- **Ridottissimo ingombro**: 20,7 x 38,7 x 12,8 mm
- **Unica alimentazione** da +3,0 a +5,0 Vdc (l'assorbimento può variare in base ai collegamenti del modulo)
- Disponibilità di **idle mode e power down mode**
- Microcontrollore **Atmel ATmega8L** (AVR RISC) con quarzo da **7,3728 Mhz**
- **8K bytes FLASH** per codice, **fino a 1K bytes FLASH per boot loader** opzionale, **1K bytes RAM** per dati, **512 bytes EEPROM** per dati

- **6 canali di A/D converter** con **10 bit** di risoluzione e **2 canali di A/D converter** con **8 bit** di risoluzione a **5 μ s** per conversione
- 18 sorgenti di **interrupt**
- **3 Timer Counter** a 8/16 bits con funzionalita' di PWM, comparazione, cattura, ecc.
- **23 linee di I/O digitale** collegate al connettore
- **Linea seriale hardware** con Baud Rate programmabile fino a 115200 Baud, bufferata in **RS 232** o TTL
- Interfaccia di **comunicazione sincrona a due fili**
- Interfaccia **SPI**
- Circuiteria di **Reset**
- **Watch dog** programmabile da 16 ms fino a 2 s
- Dip switch di configurazione a 4 vie
- **LED** di segnalazione gestito via software tramite una linea di I/O digitale
- Possibilità di gestione della FLASH ed EEPROM interna in modalita' **In System Programming**, ovvero con modulo gia' montato, sfruttando la modalità di programmazione **ISP**
- **Software gratuito per PC**, di supporto alla programmazione ISP con cui scaricare il codice generato nella FLASH di bordo
- Vasta disponibilità di **software di sviluppo** quali: Assemblatori; compilatori C (ImageCraft ICC AVR, DDS Micro C); compilatori BASIC (BASCOS AVR); ecc.
- Ricca serie di programmi dimostrativi ed esempi di utilizzo forniti sotto forma di sorgenti ampiamente commentati, per i vari ambienti di sviluppo.

Viene di seguito riportata una descrizione dei blocchi funzionali della scheda, con indicate le operazioni effettuate da ciascuno di essi. Per una più facile individuazione di tali blocchi e per una verifica delle loro connessioni, fare riferimento alla figura 2.

LINEE DI I/O DIGITALE TTL

Il Mini Modulo **GMM AM08** mette a disposizione 23 linee di I/O digitale TTL del microcontrollore Atmel ATmega8L, ovvero i segnali PB0÷PB7, PC0÷PC6, e PD0÷PD7.

Tali linee si comandano direttamente programmando gli appositi registri interni del micro oppure usando le istruzioni ad alto livello dei linguaggi di programmazione.

Via software è definibile ed acquisibile la funzionalità e lo stato di queste linee, con possibilità di associarle anche alle periferiche della scheda (Timer Counter, Interrupt, I²C BUS, SPI, ecc.), tramite una semplice programmazione di alcuni registri interni della CPU.

Per maggiori informazioni fare riferimento ai paragrafi CONNESSIONI e DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO.

A/D CONVERTER

Il Mini Modulo **GMM AM08** mette a disposizione 8 linee analogiche di A/D converter del microcontrollore Atmel ATmega8L, ovvero i segnali ADC0÷ADC7 multiplexati sulle linee PC0÷PC5 più due segnali non multiplexati (ADC6 e ADC7).

La conversione A/D si esegue tramite l'opportuna manipolazione degli appositi registri interni del micro. Per ulteriori informazioni si vedano i datasheet nell'appendice A di questo manuale o si consultino i commenti nei programmi di esempio ad alto livello.

DIP SWITCH

Il Mini Modulo **GMM AM08** è dotato di un di un dip switch di bordo il cui scopo è l'impostazione del protocollo elettrico di comunicazione del Mini Modulo.

Infatti gli switch decidono se devono arrivare ai pin dello zoccolo i segnali RS 232 generati dal MAX 3222E o direttamente i segnali TTL generati dall'USART di bordo del microcontrollore.

WATCH DOG

Il microcontrollore Atmel ATmega8L incorpora un watch dog hardware programmabile in grado di resettare la CPU se il programma utente non riesce a retriggerarlo entro il tempo di intervento selezionato.

Il range di tempi di intervento è piuttosto ampio, spaziando da circa 16 millisecondi a 2 secondi.

INTERFACCIA SPI

Il microcontrollore Atmel ATmega8L usa l'interfaccia SPI, multiplexata con alcune linee di I/O TTL, per la programmazione ISP della propria FLASH ed EEPROM interne, permette inoltre all'utente di utilizzare la stessa nota interfaccia sincrona per la comunicazione della propria applicazione con altri dispositivi dotati della stessa interfaccia.

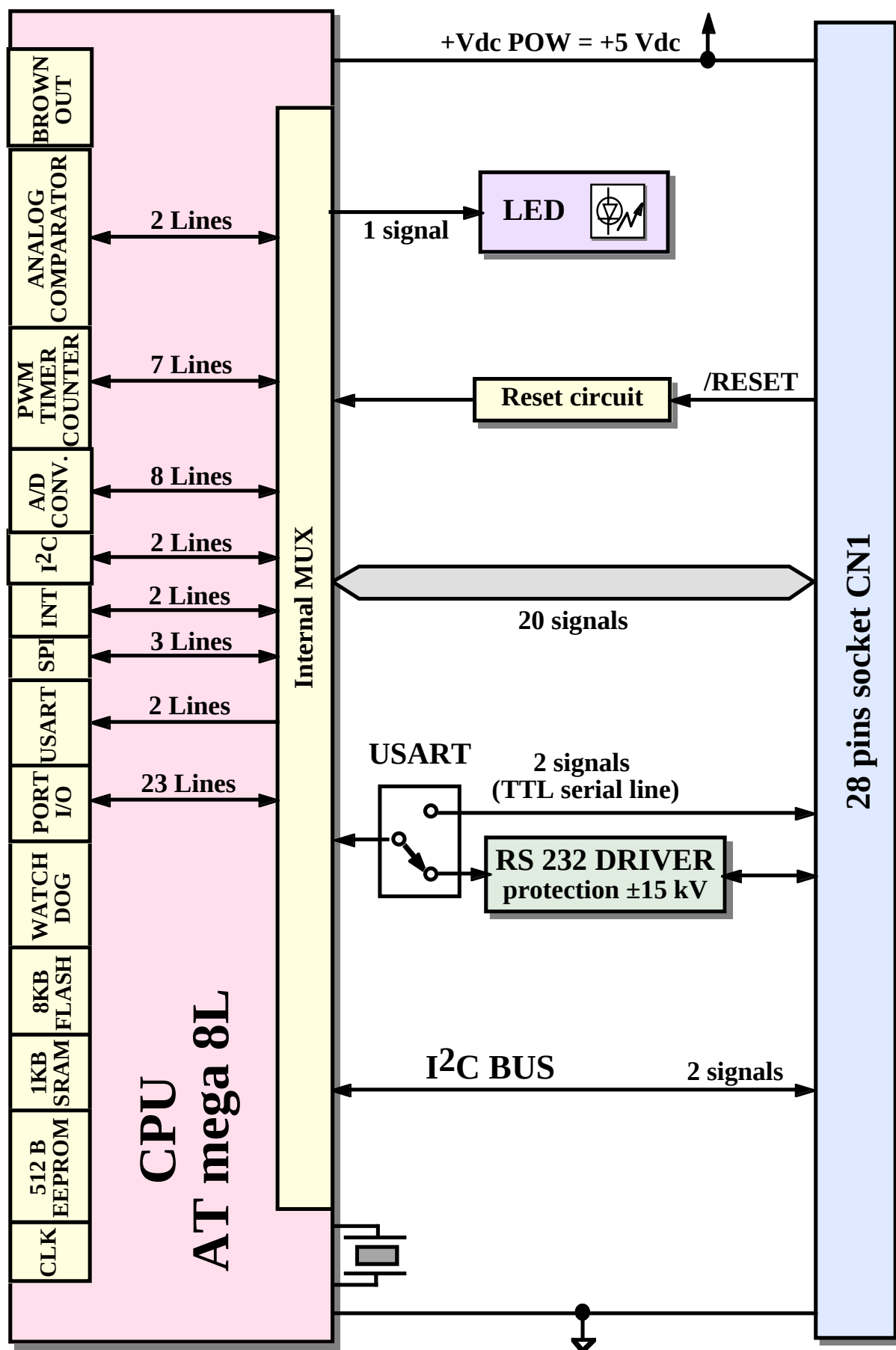


FIGURA 2: SCHEMA A BLOCCHI

LINEA I²C BUS

Il pin out standard **grifo[®] Mini Modulo** del connettore a 28 vie riserva due segnali, il 6 ed il 7, all'interfaccia I²C BUS. Questi segnali sono dotati di un pull-up da 4,7 k Ω che si trova a bordo del Mini Modulo.

Nella **GMM AM08** viene usata l'interfaccia hardware della CPU utilizzabile mediante i suoi registri interni. Questa può funzionare sia come master che come slave, in ricezione e trasmissione.

Grazie a questa interfaccia possono essere collegati dispositivi dotati dello stesso standard di comunicazione in modo da espandere localmente le potenzialità del Mini Modulo.

Una ricca serie di esempi software prevede la gestione delle più comuni e diffuse interfacce I²C BUS come A/D e D/A converter, memorie, sensori di temperatura, ecc.

Notevoli tra le periferiche connettabili in I²C BUS sono i pannelli operatore **grifo[®]** della serie **QTP**, in grado di gestire display sia grafici che alfanumerici e tastiere di varie dimensioni, a seconda del modello.

Le schede di supporto ai Mini Moduli prodotte dalla **grifo[®]** (come la **GMB HR84**) prevedono, tra le altre cose, anche un connettore dedicato all'I²C BUS, in modo da facilitare la massimo le connessioni con il campo.

COMUNICAZIONE SERIALE

La scheda dispone di una linea seriale hardware in cui il protocollo fisico (baud rate, stop bit, bit x chr, ecc.) è completamente settabile via software tramite la programmazione dei registri interni.

Quindi per ulteriori informazioni si faccia riferimento alla documentazione tecnica del microcontrollore della casa costruttrice o alle appendici di questo manuale.

La linea seriale è collegata al connettore CN1 a livello TTL o RS 232, grazie alla configurazione di alcuni dip switch di bordo quindi, quando la scheda deve essere collegata in una rete, collegata a distanza, o collegata ad altri dispositivi che usano diversi protocolli elettrici, si deve interporre un apposito driver seriale esterno (RS 422, RS 485, Current loop, ecc.).

Sul connettore CN1 oltre alle linee di ricezione e trasmissione sono disponibili anche altre linee di I/O gestibili via software che possono essere usate per definire la direzione della linea in caso di RS 485, per abilitare il driver di trasmissione in caso di RS 422 oppure come handshake hardware in caso di RS 232. Ad esempio può essere utilizzato il modulo **MSI 01** che è in grado di convertire la linea seriale TTL in qualsiasi altro standard elettrico in modo comodo ed economico.

Per maggiori informazioni contattare direttamente la **grifo[®]** e leggere il paragrafo SELEZIONE COMUNICAZIONE SERIALE.

TIMER COUNTER E PWM

Il microcontrollore mette a disposizione tre Timer/Counter (due ad otto bit, uno a sedici bit), in grado di contare gli impulsi di clock (con prescaler programmabile), transizioni di livello su opportuni pin e generare interrupt.

Inoltre possono essere usati in modalità PWM, per generare dei segnali di frequenza e duty cycle definibile via software con risoluzione otto o sedici bit.

Le applicazioni tipiche di tali segnali sono il controllo della velocità dei motori, infatti molti azionamenti dispongono di azionamenti compatibili. Oppure la generazione di segnali analogici, tramite facilmente ottenibili aggiungendo un semplice circuito integratore.

DISPOSITIVI DI MEMORIA

La scheda è dotata di un massimo di 9,5KByte di memoria variamente suddivisi con un massimo di **8KBytes** FLASH EPROM, **1KBytes** di SRAM interna e **512 Bytes** di EEPROM.

La scelta d'uso delle memorie da utilizzare può avvenire in relazione all'applicazione da risolvere e quindi in relazione alle esigenze dell'utente.

Grazie alla EEPROM di bordo c'è inoltre la possibilità di mantenere i dati anche in assenza di alimentazione. Questa caratteristica fornisce alla scheda la possibilità di ricordare in ogni condizione, una serie di parametri come ad esempio la configurazione o lo stato del sistema.

Qualora la quantità di memoria per dati risulti insufficiente (ad esempio per sistemi di data login) si possono sempre collegare dei dispositivi esterni di memoria nelle tecnologie SRAM, EEPROM e FLASH tramite le comode ed efficienti interfacce SPI ed I²C BUS della scheda.

Il mappaggio e la gestione delle risorse di memoria avviene direttamente all'interno del microcontrollore come descritto nella documentazione del componente o nell'APPENDICE A di questo manuale.

CLOCK

Nel modulo **GMM AM08** c'è una circuiteria che genera una frequenza di 7,3728 MHz per il funzionamento del microcontrollore.

Dal punto di vista delle prestazioni si ricorda che **GMM AM08** ha installato un microprocessore di tipo RISC, in grado di eseguire, mediamente, circa una istruzione ogni ciclo di clock.

Pertanto, tenendo conto del quarzo montato, la velocità di esecuzione può arrivare a circa 7 MIPS.



FIGURA 3: FOTO GMM AM08

CARATTERISTICHE TECNICHE

CARATTERISTICHE GENERALI

Risorse della scheda:	23 linee di I/O digitale TTL 8 ingressi analogici A/D converter 1 sezione Watch Dog 3 Timer/Counter programmabili 18 sorgenti di interrupt 1 circuiteria di reset 1 linea seriale RS 232 con MAX 3222 1 Dip Switch a 4 vie 1 LED di stato rosso	
Memorie:	8 Kbyte FLASH fino a 1 K byte FLASH 512 Bytes EEPROM 1024 Bytes SRAM	programma utente boot loader dati utente dati utente
CPU di bordo:	Atmel ATmega8L	
Frequenza di clock:	7,3728 MHz	
Frequenza massima contatori:	frequenza clock I/O	
Tempo di power on:	79 ms	
Risoluzione A/D:	6 linee a 10 bit 2 linee a 8 bit	
Tempo di conversione A/D:	5 μ s	

CARATTERISTICHE FISICHE

Dimensioni (L x A x P):	20,7 x 38,7 x 12,8 mm	
Peso:	6,8 g	
Connettori:	zoccolo maschio da 28 piedini	
Range di temperatura:	da 0 a 50 gradi Centigradi	
Umidità relativa:	20% fino a 90%	(senza condensa)

CARATTERISTICHE ELETTRICHE

Tensione di alimentazione:	chiamata +Vdc POW, da +3,0 a +5,0 Vdc	
Consumo di corrente a +5 Vdc:	3 mA	(power down mode)
	15 mA	(normale)
	18 mA	(massimo)
Consumo di corrente a +3, 0 Vdc:	2 mA	(power down mode)
	9 mA	(normale)
	10 mA	(massimo)
Impedenza generatori segnali analogici:	<10 K Ω	
Protezione RS 232:	± 15 kV	
Resistenza di pull-up su I²C BUS:	4,7 k Ω	
Soglie del brown out:	2, 7 o 4, 0 Vdc con isteresi	

INSTALLAZIONE

In questo capitolo saranno illustrate tutte le operazioni da effettuare per il corretto utilizzo della scheda. A questo scopo viene riportata l'ubicazione e la funzione dei connettori, dei LEDs, dei dip switch, ecc. presenti sulla **GMM AM08**.

SEGNALAZIONI VISIVE

La scheda **GMM AM08** è dotata delle segnalazioni visive descritte nella seguente tabella:

LED	SIGNIFICATO
DL1	Viene pilotato dalla linea PB.5 , SCK del mini modulo e può essere usato come LED di attività, gestito via software.

FIGURA 4: TABELLA DELLE SEGNALAZIONI VISIVE

La funzione principale di questo LED é quella di fornire un'indicazione visiva dello stato della scheda, facilitando quindi le operazioni di debug e di verifica di funzionamento di tutto il sistema. Per una più facile individuazione di tali segnalazioni visive, si faccia riferimento alla figura 8, mentre per ulteriori informazioni sui LEDs si faccia riferimento al paragrafo LED DI ATTIVITA'.

CONNESSIONI

Il modulo **GMM AM08** è provvisto di 1 connettore con cui vengono effettuati tutti i collegamenti con il campo e con le altre schede del sistema di controllo da realizzare. Di seguito viene riportato il suo pin out ed il significato dei segnali collegati; per una facile individuazione di tale connettore, si faccia riferimento alla figura 8, mentre per ulteriori informazioni a riguardo del tipo di connessioni, fare riferimento alle figure successive che illustrano il tipo di collegamento effettuato a bordo scheda e presentano alcuni dei collegamenti più frequentemente richiesti.

CN1 - CONNETTORE CON SEGNALI DEL MINI MODULO

Il connettore CN1 é uno zoccolo maschio da 28 piedini con passo 100 mils e larghezza 600 mils. Su questo connettore sono presenti tutti i segnali d'interfacciamento del Mini Modulo come l'alimentazione, le linee di I/O, le linee di comunicazione seriale sincrona ed asincrona, i segnali delle periferiche hardware di bordo, le linee di selezione del modo operativo, ecc.

Alcuni piedini di questo connettore hanno una duplice o triplice funzione infatti, via software, alcune sezioni interne della CPU possono essere multiplexate con i segnali di I/O e per completezza la seguente figura li riporta tutti. I segnali presenti su CN1 sono quindi di diversa natura, come descritto nel successivo paragrafo INTERFACCIAMENTO CONNETTORI CON IL CAMPO e seguono il pin out standardizzato dei Mini Moduli grifo®.

Al fine di evitare problemi di conteggio e numerazione la figura 5 descrive i segnali direttamente sulla vista dall'alto della **GMM AM08**, inoltre la serigrafia riporta la numerazione sui 4 angoli della scheda sia sul lato superiore che inferiore

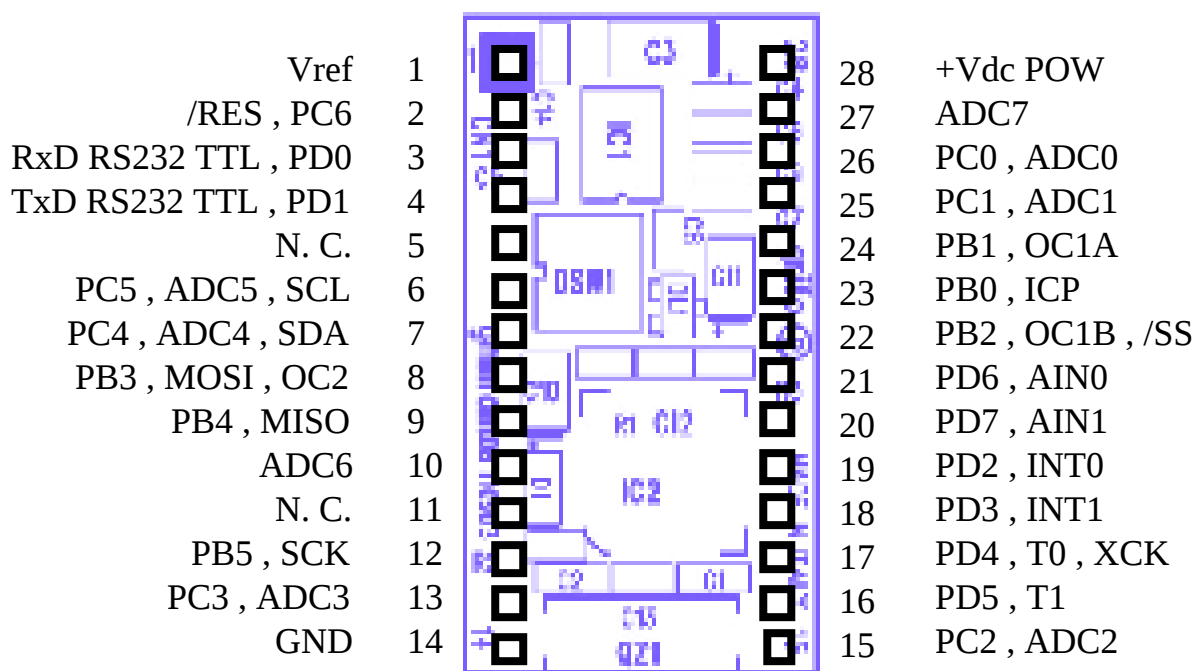


FIGURA 5: CN1 - ZOCCOLO CON SEGNALI DEL MINI MODULO

Legenda:

+Vdc POW	= I - Linea di alimentazione
GND	= - Linea di massa
RxD RS232 TTL	= I - Receive Data: linea di ricezione in RS 232 o TTL
TxD RS232 TTL	= O - Transmit Data: linea di trasmissione in RS 232 o TTL
INTn	= I - Interrupt interno della CPU (INT0 e INT1)
Tn	= I - Ingressi esterni per conteggio dei timer 0, 1 e 2
/RES	= I - Segnale di reset della CPU
PB0÷7	= I/O - Segnali del Port B di I/O TTL della CPU
PC0÷6	= I/O - Segnali del Port C di I/O TTL della CPU
PD0÷7	= I/O - Segnali del Port D di I/O TTL della CPU
ADC0÷7	= I - Ingressi analogici dell'A/D converter
AIN0÷1	= I - Ingressi analogici del comparatore analogico
OCxy	= O - Match dei compare di Timer 1 (A e B) e Timer 2
ICP	= I - Input capture pin del Timer 1
SDA	= I/O - Segnale dati dell'interfaccia seriale sincrona a due fili (I ² C BUS)
SCL	= I - Segnale clock dell'interfaccia seriale sincrona a due fili (I ² C BUS)
SCK	= I - Segnale clock dell'interfaccia seriale sincrona SPI
MOSI	= I - Segnale dati in ingresso dell'interfaccia seriale sincrona SPI
MISO	= I - Segnale dati in uscita dell'interfaccia seriale sincrona SPI
/SS	= I - Segnale Slave Select dell'interfaccia seriale sincrona SPI
Vref	= I - Tensione di riferimento dell'A/D converter
N. C.	= - Nessuna connessione

CONFIGURAZIONE MINI MODULO

A bordo del Mini Modulo **GMM AM08** è presente un dip switch ad 4 vie, con cui é possibile effettuare alcune selezioni che riguardano il modo di funzionamento dello stesso. Nelle successive figure ne è riportato l'elenco, l'ubicazione e la funzione nelle varie modalità di connessione.

Nelle seguenti tabella l'* (asterisco) indica la connessione di default, ovvero quella impostata in fase di collaudo, con cui la scheda viene fornita.

Per individuare la posizione degli elementi di configurazione si faccia riferimento alla figura 8.

Per ulteriori informazioni sulla configurazione delle linee di comunicazione seriele, si faccia riferimento al paragrafo "SELEZIONE COMUNICAZIONE SERIALE".

SWITCH	POSIZIONE	UTILIZZO	DEF.
1	ON	Collega il segnale RxD RS232 TTL , PD0 dello zoccolo CN1 al driver RS 232 di bordo. Usato in coppia con lo switch 3.	*
	OFF	Non collega il segnale RxD RS232 TTL , PD0 dello zoccolo CN1 al driver RS 232 di bordo, consentendo il collegamento diretto al microcontrollore. Usato in coppia con lo switch 3.	
2	ON	Collega il segnale TxD RS232 TTL , PD0 dello zoccolo CN1 al driver RS 232 di bordo. Usato in coppia con lo switch 4.	*
	OFF	Non collega il segnale TxD RS232 TTL , PD0 dello zoccolo CN1 al driver RS 232 di bordo, consentendo il collegamento diretto al microcontrollore. Usato in coppia con lo switch 4.	
3	ON	Collega il segnale RxD RS232 TTL , PD0 dello zoccolo CN1 direttamente al microcontrollore, eliminando il driver RS 232 di bordo. Usato in coppia con lo switch 1.	*
	OFF	Non collega il segnale RxD RS232 TTL , PD0 dello zoccolo CN1 al microcontrollore, consentendo l'uso del driver RS 232 di bordo. Usato in coppia con lo switch 1.	
4	ON	Collega il segnale TxD RS232 TTL , PD0 dello zoccolo CN1 direttamente al microcontrollore, eliminando il driver RS 232 di bordo. Usato in coppia con lo switch 2.	*
	OFF	Non collega il segnale TxD RS232 TTL , PD0 dello zoccolo CN1 al microcontrollore, consentendo l'uso del driver RS 232 di bordo. Usato in coppia con lo switch 2.	

FIGURA 6: TABELLA DIP SWITCH DSW1 A 4 VIE

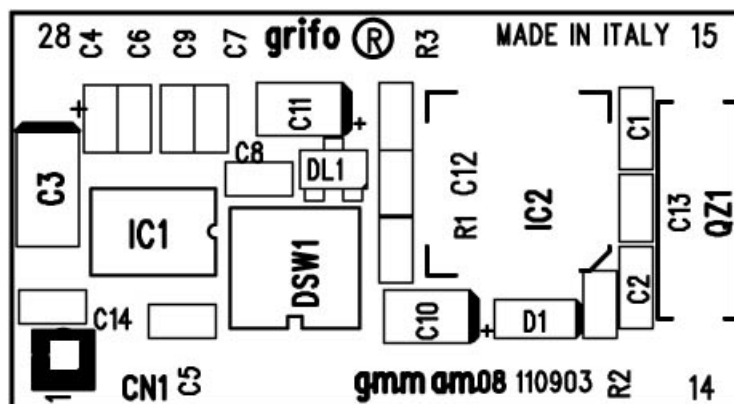


FIGURA 7: PIANTA COMPONENTI (LATO COMPONENTI)

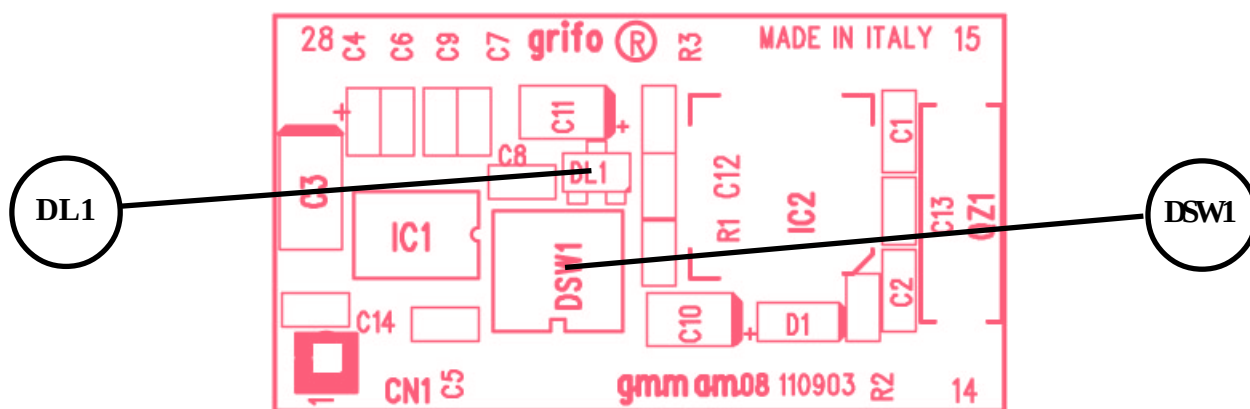


FIGURA 8: DISPOSIZIONE LEDs, DIP SWITCHES, ECC.

SELEZIONE COMUNICAZIONE SERIALE

La linea di comunicazione seriale della scheda **GMM AM08** può essere bufferate in RS 232 o TTL. Dal punto di vista software sulla linea può essere definito il protocollo fisico di comunicazione tramite la programmazione di alcuni registri interni del microprocessore.

L'interfaccia seriale possiede un gruppo indipendente di registri per la configurazione, e può funzionare in maniera totalmente indipendente rispetto alle altre periferiche.

La selezione del protocollo elettrico avviene via hardware e richiede un'opportuna configurazione del dip switch di bordo, come descritto nelle precedenti tabelle; l'utente può autonomamente passare da una configurazione all'altra seguendo le informazioni riportate di seguito:

- LINEA SERIALE SETTATA IN RS 232 (configurazione default)

DSW1.1	=	ON
DSW1.2	=	ON
DSW1.3	=	OFF
DSW1.4	=	OFF

- LINEA SERIALE SETTATA IN TTL

DSW1.1	=	OFF
DSW1.2	=	OFF
DSW1.3	=	ON
DSW1.4	=	ON

Le figure 9 e 10 illustrano come collegare un generico sistema esterno alla linea seriale della **GMM AM08** rispettivamente in configurazione RS 232 e TTL.

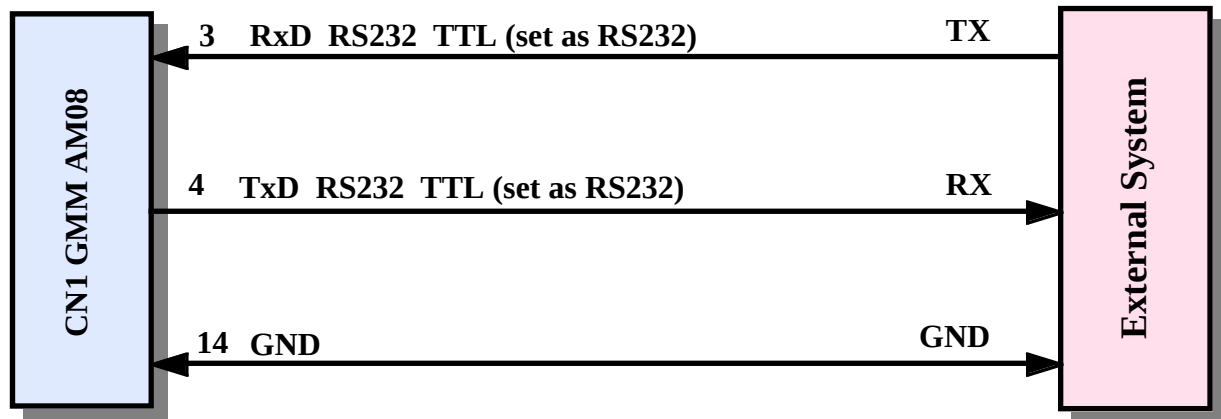


FIGURA 9: ESEMPIO COLLEGAMENTO SERIALE RS 232

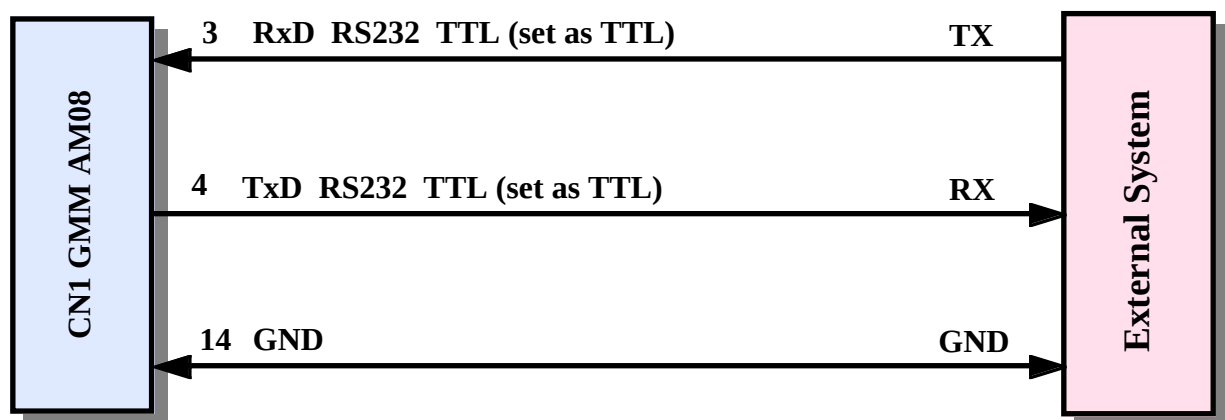


FIGURA 10: ESEMPIO COLLEGAMENTO SERIALE TTL

INTERFACCIAMENTO CONNETTORI CON IL CAMPO

Al fine di evitare eventuali problemi di collegamento della scheda con tutta l'elettronica del campo a cui **GMM AM08** si deve interfacciare, si devono seguire le informazioni riportate nei vari paragrafi.

- Per i segnali che riguardano la comunicazione seriale con il protocollo RS 232, fare riferimento alle specifiche standard degli stessi.
- Tutti i segnali a livello TTL possono essere collegati a linee dello stesso tipo riferite alla massa digitale della scheda. Il livello 0V corrisponde allo stato logico 0, mentre il livello pari alla tensione di alimentazione corrisponde allo stato logico 1. La connessione di tali linee ai dispositivi del campo (fine corsa, encoders, elettrovalvole, relé di potenza, ecc.) deve avvenire tramite apposite interfacce di potenza che preferibilmente devono essere optoisolate in modo da mantenere isolata la logica del Mini Modulo dagli eventuali disturbi dell'elettronica di potenza.
- I segnali d'ingresso del comparatore analogico devono essere collegati a generatori di segnali analogici a bassa impedenza che rispettino il range di variazione ammesso ovvero da 0 V alla tensione di riferimento selezionata.
- I segnali d'ingresso degli A/D converter devono essere collegati a generatori di segnali analogici a bassa impedenza che rispettino il range di variazione ammesso ovvero da 0 V alla tensione di riferimento selezionata.
- I segnali PWM generati dalle sezioni Timer Counter e OCM, sono a livello TTL e devono essere quindi opportunamente bufferati per essere interfacciati all'azionamento di potenza. Le classiche circuiterie da interporre possono essere dei semplici driver di corrente se è ancora necessario un segnale PWM, oppure un integratore qualora sia necessario un segnale analogico.
- Anche i segnali I²C BUS ed SPI sono a livello TTL, come definito dallo stesso standard; per completezza si ricorda solo che dovendo realizzare una rete con numerosi dispositivi e con una discreta lunghezza si deve studiare attentamente il collegamento oppure configurare lo stadio d'uscita, le molteplici modalità operative ed il bit rate programmabili opportunamente in modo da poter comunicare in ogni condizione operativa. Sul Mini Modulo i segnali di SDA ed SCL hanno un pull-up verso +Vdc POW di 4,7 kΩ.

ALIMENTAZIONE

Il Mini Modulo può essere alimentato mediante una tensione continua nell'intervallo da +3,0 a +5,0 Vdc, indicata con +Vdc POW.

Sulla scheda sono state adottate tutte le scelte circuitali e componentistiche che tendono a ridurre la sensibilità ai disturbi ed i consumi, compresa la possibilità di far lavorare il microcontrollore in quattro diverse modalità a basso assorbimento.

Nella condizione ottimale si arriva ad un consumo minimo (in modalità power down) di 2 mA che ad esempio salvaguarda la durata di batterie, nel caso di applicazioni portatili.

Informazioni più dettagliate sono riportate nel capitolo CARATTERISTICHE ELETTRICHE.

INTERRUPTS

Una caratteristica peculiare della **GMM AM08** è la notevole potenza nella gestione delle interruzioni. Di seguito viene riportata una breve descrizione di quali sono i dispositivi che possono generare interrupts e con quale modalità; per quanto riguarda la gestione di tali interrupts si faccia riferimento ai data sheets del microprocessore oppure all'appendice A di questo manuale.

- Pin 19 di CN1 -> Genera un INT0 sulla CPU.
- Pin 18 di CN1 -> Genera un INT1 sulla CPU.
- Periferiche della CPU-> Generano un interrupt interno. In particolare le possibili sorgenti d'interrupt sono: Timer 0, Timer 1, Timer 2, I²C BUS, USART, A/D converter, comparatore A/D, EEPROM, SPI, SPM Ready, Brown out, Watch Dog, interrupt esterni.

Incorporata nel microcontrollore si trova la logica di gestione degli interrupt che consente di attivare, disattivare, mascherare le 18 sorgenti d'interrupt e che regola l'attivazione contemporanea di più interrupts. In questo modo l'utente ha sempre la possibilità di rispondere in maniera efficace e veloce a qualsiasi evento esterno, stabilendo anche la priorità delle varie sorgenti.

SCHEDE DI SUPPORTO

Il Mini Modulo **GMM AM08** può essere utilizzato come macro componente per alcune schede di supporto sia sviluppate dall'utente che già disponibili nel carteggio **grifo®**. Nei successivi paragrafi vengono illustrate le configurazioni delle schede di supporto più interessanti.

UTILIZZO CON IL MODULO BLOCK GMB HR84

La **GMB HR84** si distingue per essere una scheda che fornisce ai Mini Moduli da 28 pin fino a 8 ingressi optoisolati; 4 uscite a relè, la possibilità di montaggio meccanico su barra ad omega ed il cablaggio tramite comode morsettiere.

La descrizione completa del prodotto è disponibile nel manuale tecnico e nel manuale dell'accoppiata, mentre in questo paragrafo sono riportate le potenzialità offerte.

La **GMB HR84** permette facilmente di:

- alimentare il Mini Modulo tramite l'alimentatore di bordo;
- riportare otto linee dei port di I/O sugli ingressi optoisolati che possono essere indifferentemente connessi come NPN o PNP. Lo stato di tutti gli 8 ingressi viene visualizzato tramite LEDs Verdi. Essendo le linee multiplexate con le periferiche interne è possibile creare rapidamente funzioni evolute come contatori, riconoscimento combinazioni, ecc.;
- riportare quattro linee dei port di I/O su uscite a Relè visualizzate tramite LEDs Rossi.
- avere le linee dell'I²C BUS e dell'alimentazione su un connettore distinto;
- collegare la linea di comunicazione seriale tramite un connettore DB9 da 9 vie;
- bufferare, esternamente al Mini Modulo, i segnali della linea seriale 0 TTL in RS 422, RS 485 o Current Loop;
- collegare i segnali PWM su un connettore AMP MOD II da 8 vie;



FIGURA 11: FOTO DI GMB HR84 E GMM AM08 CON E SENZA CONTENITORE

UTILIZZO CON LA SCHEDA GMM TST 2

Nel carteggio delle schede **grifo®** la **GMM TST 2** si distingue per essere la scheda prototipale progettata esplicitamente per fare da supporto ai Mini Moduli **GMM xxx** da 28 e 40 pins.

Lo schema elettrico della **GMM TST 2** viene mostrato nell'appendice B.

La **GMM TST 2** permette:

- di alimentare il Mini Modulo tramite l'alimentatore di bordo
- di riportare le linee dei port di I/O e dell'A/D converter su un connettore a scatolino compatibile con lo standard **I/O ABACO®**
- di collegare la linea RS 232 tramite un connettore a vaschetta D9 femmina
- di impostare e visualizzare lo stato di 2 linee di I/O, del microcontrollore tramite pulsanti e LEDs di colori differenti, escludibili tramite jumper.
- di generare feedback sonori mediante il buzzer di bordo
- di sviluppare rapidamente e confortevolmente applicazioni di interfaccia utente avvalendosi della tastiera a matrice 4x4 da 16 tasti e del display LCD retroilluminato da 2 righe di 20 caratteri.
- di realizzare facilmente una scheda di supporto che soddisfi le esigenze dell'utente partendo dagli schemi elettrici forniti
- di programmare FLASH ed EEPROM del Mini Modulo usando il Boot Loader e in modalità ISP

La seguente configurazione consente di usare l'accoppiata **GMM TST 2 + GMM AM08** collegato alla linea in RS 232:

*Configurazione **GMM AM08***

DSW1.1	=	ON
DSW1.2	=	ON
DSW1.3	=	OFF
DSW1.4	=	OFF

*Configurazione **GMM TST 2***

J1	=	2-3
J2	=	2-3
J3	=	non connesso
J4	=	non connesso
J5	=	non connesso
J6	=	non connesso
J7	=	non connesso

Cavo collegamento seriale con PC di sviluppo = CCR 9+9E (ovvero cavo prolunga con vaschetta D9 Femmina e D9 Maschio).

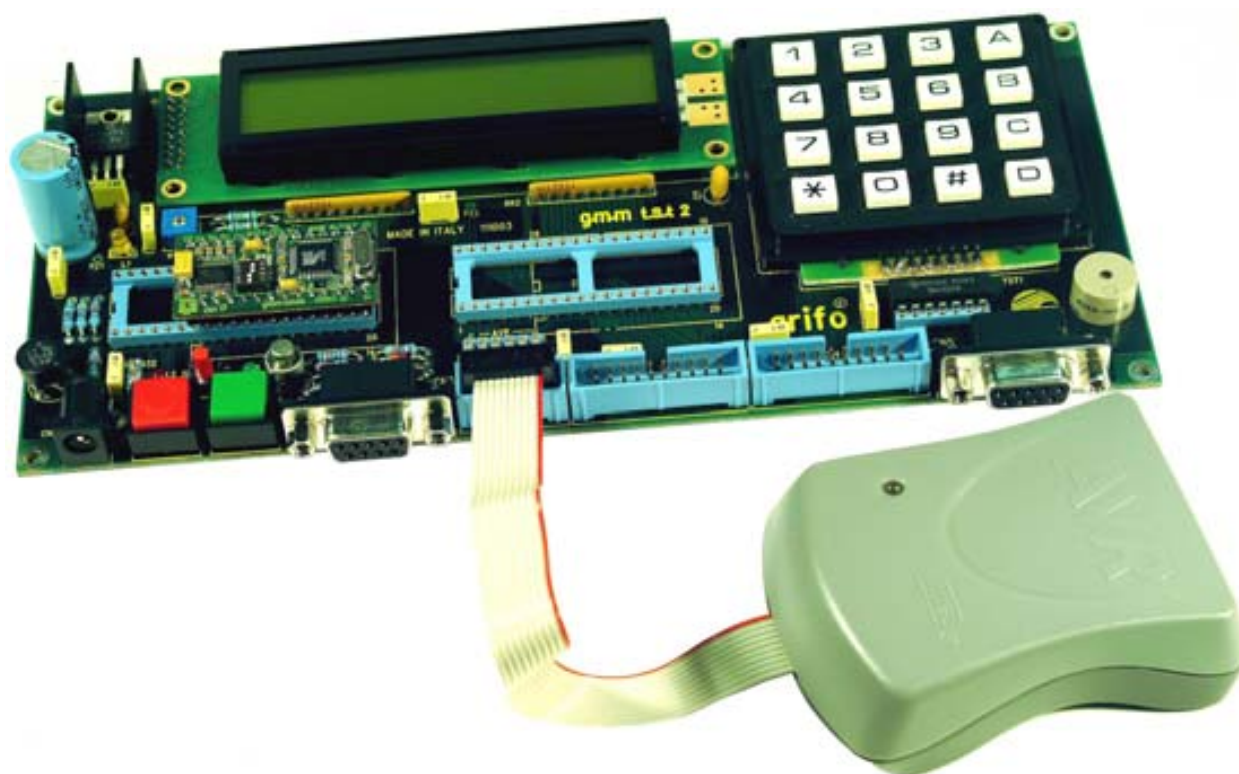


FIGURA 12: SCHEDA DI SPERIMENTAZIONE GMM TST 2 CON GMM AM08 E AVR ISP

COME INIZIARE

Una delle caratteristiche più interessanti è la possibilità di programmare la FLASH del microcontrollore Atmel ATmega8L attraverso appositi tools prodotti dalla **grifo®** e dalla Atmel.

In questa fase si ipotizza di poter disporre di una **GMM TST 2** o di una **GMB HR84** ove montare il Mini Modulo **GMM AM08**.

Per ulteriori informazioni sull'accoppiata **GMB HR84 + GMM AM08**, consultare lo specifico manuale.

A) COLLEGAMENTO SERIALE TRA GMM AM08 ED IL PC

A1) Per prima cosa dovete realizzare fisicamente il collegamento seriale tra il Mini Modulo **GMM AM08** ed il PC. Per questo bisogna costruire un cavo che effettui il collegamento descritto in figura 13.

Il programma già presente nel momento in cui la scheda viene fornita al cliente effettua un lampeggiamento alternato del LED di bordo, senza usare in alcun modo l'interfaccia seriale. Il programma demo invece fa largo uso di tale interfaccia, per la quale presenta anche una sezione dedicata. Pertanto è opportuno provvedere subito al collegamento seriale.

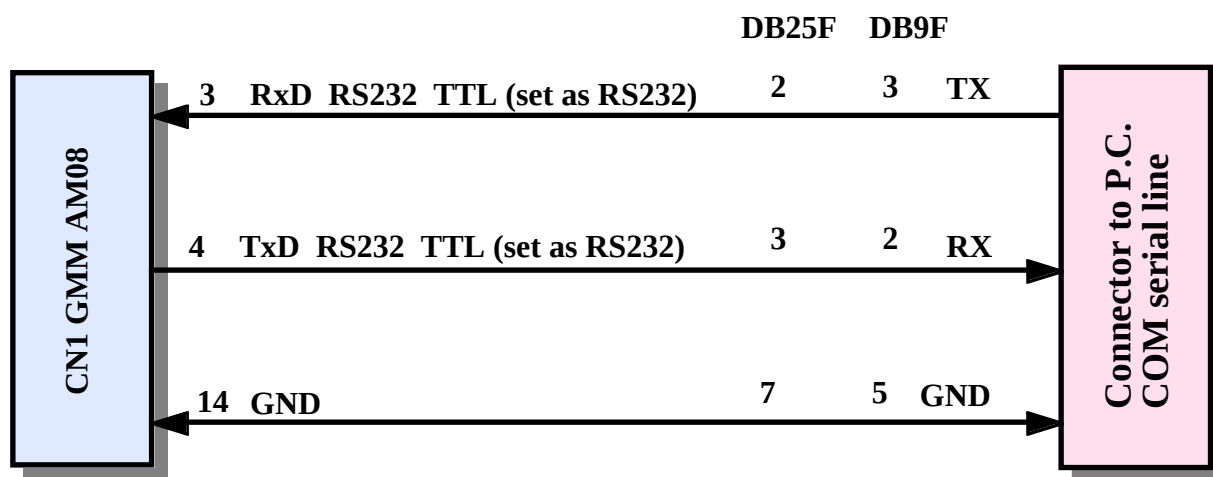


FIGURA 13: COLLEGAMENTO SERIALE RS 232 TRA GMM AM08 E PC

A2) Aprire un emulatore di terminale sul PC (ad esempio Hyperterminal), configurarlo per usare la porta seriale collegata al **GMM AM08** con 19200 baud, 8 bit di dati, 1 bit di stop, nessuna parità.

A3) Alimentare **GMM TST 2** o **GMB HR84**. Il LED di bordo del Mini Modulo deve cominciare a lampeggiare alternativamente circa due volte al secondo.



TIPO DI SCHEDA	GET	ASM	Ladder	Abaco® Link BUS	BASIC CBZ80	BASIC BASCOM 8051	BASIC BASCOM AVR	PIC BASIC	BASIC VARI	MCS® Basic 52	C	PASCAL	TIPO DI CPU / BLOCK
VARI	-	-	-	-	-	-	-	-	-	-	-	-	-
CAN GM0	-	-	-	-	-	-	-	-	-	-	-	-	Atmel T89C51cc03 - 8051 Code
CAN GM1	-	-	-	-	-	-	-	-	-	-	-	-	Atmel T89C51cc01 - 8051 Code
CAN GM2	-	-	-	-	-	-	-	-	-	-	-	-	Atmel T89C51cc02 - 8051 Code
GMM 5115	-	-	-	-	-	-	-	-	-	-	-	-	Atmel T89C5115 - 8051 Code
GMM 876	-	-	-	-	-	-	-	-	-	-	-	-	Microchip PIC16F876A - PIC 14 Code
GMM 932	-	-	-	-	-	-	-	-	-	-	-	-	PHILIPS P89LPC932 - 8051 Code
GMM AC2	-	-	-	-	-	-	-	-	-	-	-	-	Atmel T89C51AC2 - 8051 Code
GMM AM08	-	-	-	-	-	-	-	-	-	-	-	-	Atmel ATmega08 - AVR Code
GMM AM32	-	-	-	-	-	-	-	-	-	-	-	-	Atmel ATmega32 - AVR Code
GMB HR84	-	-	-	-	-	-	-	-	-	-	-	-	Mini Block 8 input opto 4 output relè
GMB HR168	-	-	-	-	-	-	-	-	-	-	-	-	Mini Block 16 input opto 8 output relè




FIGURA 14: TABELLA ESEMPI

B) RIPROGRAMMAZIONE DELLA FLASH

La programmazione del Mini Modulo avviene utilizzando un Boot Loader preprogrammato nella Flash del Mini Modulo stesso.

Tale programma permette di creare una comunicazione tra il Mini Modulo ed il PC, e di usarla per inviare un file in formato .hex nella memoria codice e/o nella EEPROM del microcontrollore.

Il Boot Loader non permette di programmare i bit di configurazione ed i fusibili di sicurezza del microcontrollore, cosa che comunque è possibile fare con la programmazione ISP (si veda l'appendice B per ulteriori informazioni).

L'occupazione del Boot Loader è pari a 2KBytes, ovvero viene riservato il blocco alla fine della memoria Flash pari ad 1KWord, che quindi non è più disponibile per il programma utente.

Il Boot Loader comunica tramite la porta seriale del PC installando una apposita utility (scaricabile gratuitamente dal nostro sito www.grifo.it o dal nostro CD) chiamata AVRBootloaderGrifo, che permette di selezionare la porta seriale da usare e di selezionare i files con cui programmare le memorie del Mini Modulo.

Vi sono metodi di programmazione che richiedono l'uso di un programmatore ISP. Questi sono descritti nell'appendice B.

La casella combinata "Com Port" permette di selezionare la porta seriale da usare per la comunicazione con il Mini Modulo.

Mettendo lo spunto alla casella "Application Code" il file il cui nome è scritto nella casella di testo accanto verrà memorizzato nella Flash del Mini Modulo. Per scegliere il nome del file premere il pulsante "Browse" adiacente.

Analogamente, la casella "EEPROM Code" controlla la programmazione del contenuto della EEPROM.

Esiste inoltre la possibilità di utilizzare direttamente il AVRBootloaderGrifo da finestra DOS, Prompt dei comandi, linea di comando o come tool esterno in un IDE.

Le opzioni accettate dalla linea di comando sono:

/com1 , /com2 , ... , /com9	Numero della porta seriale usata per la comunicazione. Sono supportate le porte dalla COM1 alla COM9
/f <ProgrammaFLASH>	Specifica il nome completo di path del file .HEX da scrivere nella memoria FLASH del microcontrollore
/e <ProgrammaEEPROM>	Specifica il nome completo di path del file .HEX da scrivere nella memoria EEPROM del microcontrollore

Ad esempio:

AVRBootloaderGrifo.exe /com2 /f C:\Progetti\ControlloMotori\Main.hex

apre una connessione sulla porta seriale COM2 verso il Boot Loader ed invia il file Main.hex che si trova nella cartella C:\Progetti\ControlloMotori\.

Per integrare AVRBootloaderGrifo.exe in un IDE, ad esempio quello del BASCOM AVR (descritto nel punto C), si deve.

- 1) Aprire la finestra del menu Options | Programmer
 - 2) Scegliere nella casella di riepilogo Programmer "External Programmer"
 - 3) Scegliere la linguetta Other
 - 4) Nella casella Program inserire il path di AVRBootloaderGrifo.exe premendo il pulsante Browse
 - 5) Nella casella di testo Parameters inserire i parametri per la linea di comando di AVRBootloaderGrifo. Tenete presente che inserendo la stringa {file} nella suddetta casella di testo, il BASCOM AVR la sostituisce automaticamente con il nome del file .hex appena generato.
- Si veda la documentazione del BASCOM AVR per ulteriori informazioni.

- B1) Localizzare e salvare in una posizione comoda sul disco rigido del PC il file si chiama "d_am08.HEX" seguendo il percorso: Italiano | Programmi di Esempio | Programmi Mini Moduli e Mini Block | GMM AM08 (vedere la figura 14). Eventualmente, rimuovere l'attributo di sola lettura (read-only) del file salvato.
- B2) Collegare il Mini Modulo alla porta seriale del PC realizzando il collegamento elettrico descritto nel punto A. Chiudere l'emulatore terminale.
- B3) Indicare nella casella combinata "Com Port" la porta seriale collegata al punto precedente.
- B4) Mettere lo spunto alla casella "Application Code" e premere il pulsante "Browse" adiacente, quindi selezionare il file precedentemente salvato al punto B1.
- B5) Togliere lo spunto alla casella "EEPROM code", se presente.
- B6) Premere il pulsante "Synch to Bootloader..." o la combinazione di tasti Alt+S sul PC, quindi resettare il Mini Modulo o togliere e rimettere l'alimentazione. Il file viene scaricato nel Mini Modulo. Se questo non dovesse accadere, e il programma indicasse il messaggio "No response from target bootloader", provare a ripetere l'operazione diminuendo il tempo che intercorre tra la pressione del tasto su PC ed il reset del Mini Modulo. Se il problema dovesse persistere, ricontrollare il cavo ed i collegamenti.
- B7) Alla fine dell'operazione il programma ne riporta l'esito. In caso di problemi ricontrollare il cavo ed i collegamenti.
- B8) Avviare l'emulatore terminale configurato come in A2 e verificare che il programma applicativo appena memorizzato venga eseguito dalla FLASH interna.



FIGURA 15: BOOT LOADER UTILITY

C) CREAZIONE DEL CODICE ESEGUIBILE DEL PROGRAMMA DEMO

- C1) Installare sul disco rigido del PC l'ambiente di sviluppo scelto per realizzare programma applicativo. Sono disponibili diversi ambienti in modo da soddisfare le richieste di ogni utente, ma qui si ricordano quelli più diffusi come il BASCOM AVR, ICC AVR, ecc., si faccia riferimento al manuale del software per ulteriori informazioni.
- C2) Sul CD **grifo®** oltre al file con il codice eseguibile del demo, descritto al punto B2, sono presenti anche il/i file sorgenti dello stesso. Questi hanno un'estensione che identifica l'ambiente di sviluppo usato (d_am08.bas per il BASCOM AVR, d_am08.c per ICC AVR) e sono opportunamente organizzati nelle tabelle degli esempi presenti sul CD, insieme agli eventuali file di definizione o di progetto (ad esempio: d_am08.prj per ICC AVR). Una volta localizzati questi file devono essere salvati in una posizione comoda sul disco rigido del PC di sviluppo.
- C3) Ricompilare il sorgente usando l'ambiente di sviluppo scelto, in modo da ottenere il file d_am08.hex identico a quello presente sul CD **grifo®** e già usato nei punti a partire da B1. Questa operazione si differenzia notevolmente a seconda dell'ambiente di sviluppo utilizzato, pertanto qui di seguito vengono esposti i passi dettagliati:

C3 Bascom AVR) Ricompilazione con BASCOM AVR.

C3 Bascom AVR a) Una volta entrati nell'IDE del BASCOM, caricare il programma sorgente con il menu File | Open:

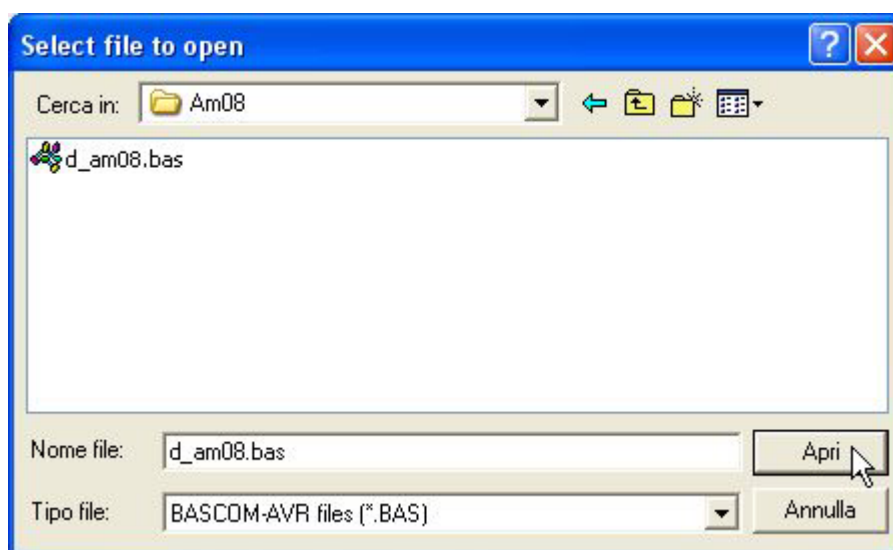


FIGURA 16: CARICAMENTO SORGENTE CON BASCOM AVR

C3 Bascom AVR b) Dal menu Options | Compiler | Chip impostare i valori 64 per HW Stack, 32 per Soft Stack, 64 per Framesize e premere OK. Tali valori sono da considerarsi dei minimi e, se necessario, vanno aumentati:

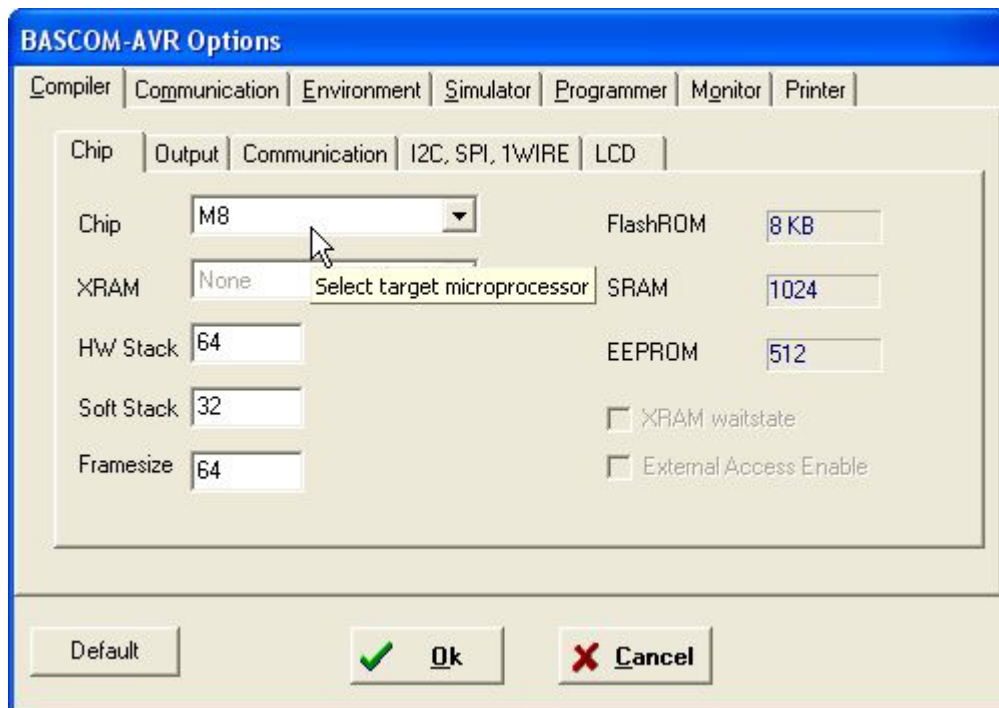


FIGURA 17: CONFIGURAZIONE COMPILATORE BASCOM AVR

C3 Bascom AVR c) Compilare il sorgente premendo il pulsante con il disegno del circuito integrato:

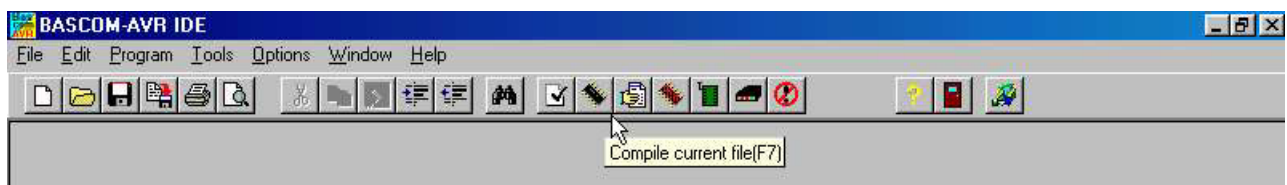


FIGURA 18: COMPILAZIONE CON BASCOM AVR

C3 ICC AVR) Ricompilazione con ICC AVR.

C3 ICC AVR a) Una volta aperto l'editor standard, caricare il file di progetto usando il menu Project | Open...:

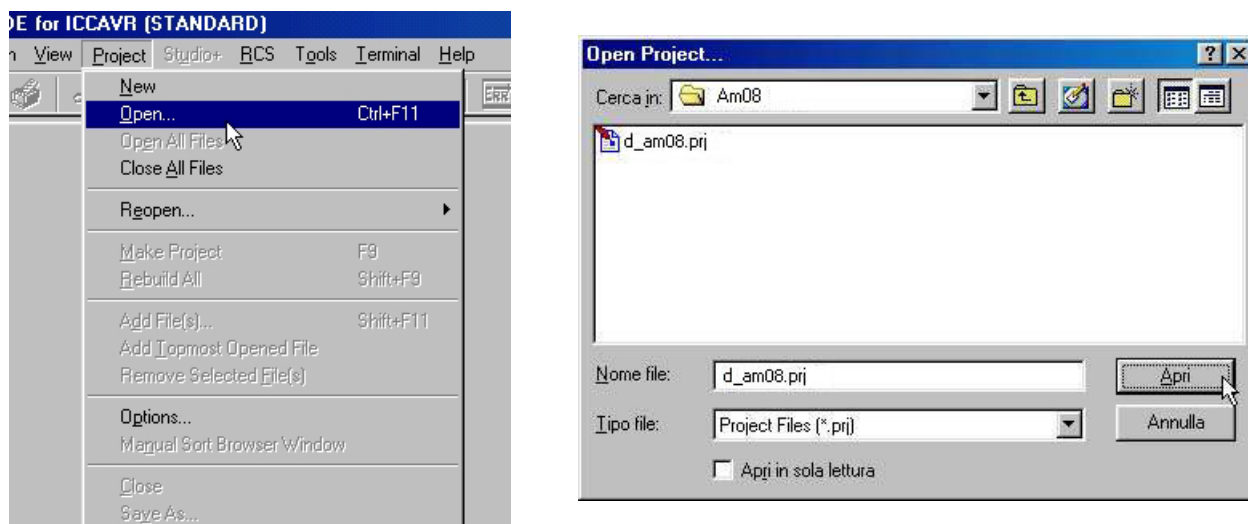


FIGURA 19: CARICAMENTO FILE DI PROGETTO CON ICC AVR

C3 ICC AVR b) Compilare usando il menu Project | Make Project:

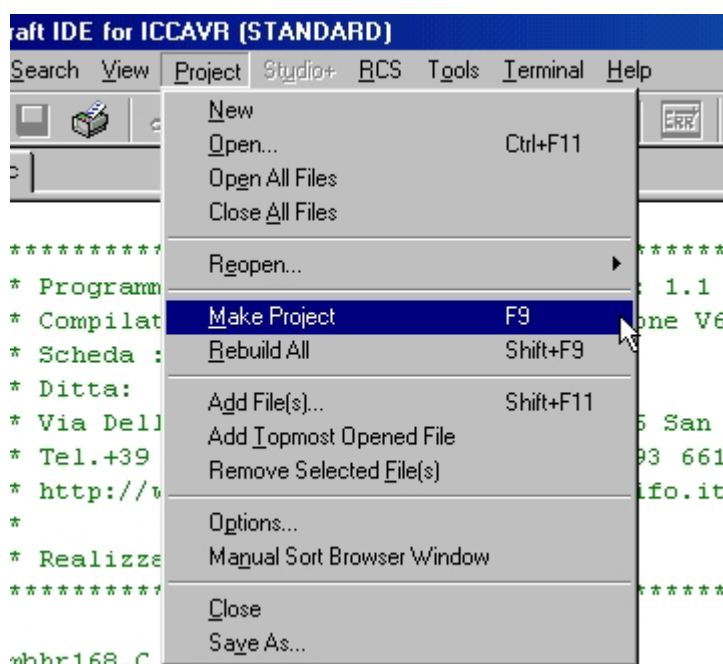


FIGURA 20: COMPILAZIONE CON ICC AVR

- C4) Effettuare il salvataggio del file ottenuto nella FLASH del Mini Modulo, ripetendo i punti della sezione B.

D) PREPARAZIONE DEFINITIVA DELL'APPLICATIVO

- D1) Chiudere il programma di comunicazione con il Boot Loader.

Se durante l'esecuzione dei passi sopra elencati si presenta un problema od un'anomalia si consiglia all'utente di rileggere e ripetere i passi con attenzione e qualora il malfunzionamento persista, di contattare direttamente la **grifo®**.

In caso di esecuzione corretta di tutte le fasi sopra descritte l'utente ha realizzato e salvato il suo primo programma applicativo coincidente con il demo del Mini Modulo **GMM AM08**.

A questo punto è possibile modificare il sorgente del/dei programmi demo in modo da soddisfare le richieste dell'applicazione da realizzare e provarla con i passi sopra elencati (parti B e C) in modo ciclico, fino a quando il programma applicativo realizzato è perfettamente funzionante.

Raggiunto questo obiettivo, si può eliminare il PC di sviluppo.

Ricordarsi di riconfigurare la seriale 0 del Mini Modulo **GMM AM08** in RS 232, se richiesto.



DESCRIZIONE SOFTWARE

Questa scheda ha la possibilità di usufruire di una ricca serie di strutture software che consentono di utilizzarle al meglio le caratteristiche. In generale il Mini Modulo può sfruttare tutte le risorse software per il microprocessore montato e tutti i pacchetti ideati per la famiglia AVR, sia ad alto che a basso livello. Tutti i pacchetti di sviluppo software forniti dalla **grifo®** sono sempre accompagnati da esempi che illustrano come gestire le sezioni della scheda e da una completa documentazione.

BASCOM AVR

Cross compilatore a basso costo per files sorgenti scritti in BASIC, disponibile in ambiente WINDOWS con un comodo IDE che mette a disposizione un editor, il compilatore ed un simulatore molto potente per il debugger del sorgente. Comprende molti modelli di memoria, svariati tipi di dati, istruzioni dedicate alle risorse hardware e librerie aggiuntive per simulare od interfacciare dispositivi intelligenti esterni (tastiere PC, display alfanumerici e grafici, lettori di badge, ecc.). L'IDE é in grado di eseguire anche programmi esterni per la programmazione ISP.

Viene fornito su CD con la relativa manualistica tecnica e gli esempi d'uso.

ICC AVR

Cross compilatore per files sorgenti scritti in ANSI C, disponibile in ambiente WINDOWS con un comodo IDE che mette a disposizione un editor, il compilatore un ottimizzatore ed un linker. Comprende molti modelli di memoria, svariati tipi di dati, le classiche librerie ANSI con i relativi sorgenti, gestione completa del floating point, un application builder, ecc. Il compilatore si integra con il pacchetto AVR Studio con cui si può effettuare il debugger a livello sorgente dell'applicativo utente in C e con programmi esterni per la programmazione ISP.

Viene fornito su CD con la relativa manualistica tecnica e gli esempi d'uso.

DDS MICRO C AVR

E' un comodo pacchetto software, a basso costo, che tramite un completo IDE permette di utilizzare un editor, un compilatore "C" (integer), un assembler, un linker ed un ottimizzatore. Sono inclusi i sorgenti delle librerie, una serie di utility ed una ricca documentazione su dischetto da 3"1/2 nel formato MS-DOS.

AVR Studio

E' un ambiente di sviluppo per la famiglia AVR di microprocessori che consente di controllare l'esecuzione dei programmi applicativi tramite appositi in circuit emulator esterni oppure in simulazione su PC. AVR Studio consente il debug a livello sorgente assembly e/o C di codici oggetto generati da pacchetti esterni (assemblatori, compilatori, ecc.) e si presenta con una serie di finestre che riportano le principali informazioni necessarie come i registri, la memoria, le periferiche, le variabili in modo da fornire all'utente il pieno controllo di ogni elemento. Include inoltre un "costruttore di applicativo" che facilita la generazione del codice che inizializza le varie periferiche hardware (UART, SPI, Port, ADC, ecc) partendo da un'interfaccia grafica.

Notevole inoltre è la possibilità di pilotare l'interfaccia JTAG "JTAG ICE" prodotta dalla ATMEL. Una interfaccia JTAG permette di entrare nel cuore del microcontrollore per esaminarne lo stato durante l'esecuzione stessa direttamente nella struttura dell'applicativo.

L'utente ha la possibilità di inserire dei breakpoint sia hardware che software, e quando l'esecuzione è ferma può esaminare lo stato dei registri interni e della memoria.

L'interfaccia JTAG permette anche di programmare le memorie del microcontrollore.

Usando l'interfaccia JTAG la difficoltà di molti problemi di debugging viene drasticamente ridotta, aumentando così la facilità di eliminazione, e diminuendo di conseguenza il tempo necessario per giungere all'applicativo finale totalmente debuggato.

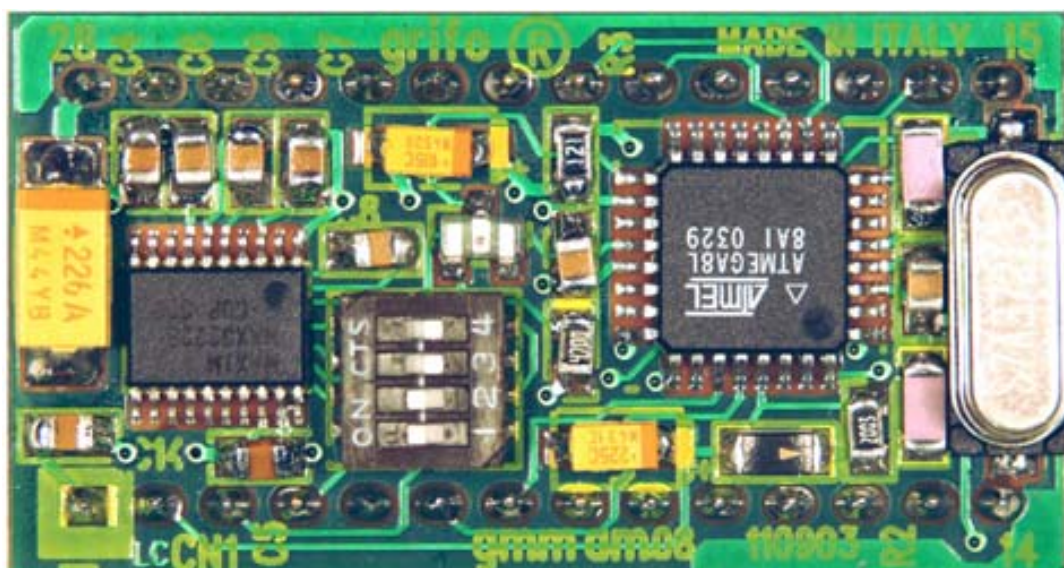


FIGURA 21: VISTA DALL'ALTO DELLA GMM AM08

DESCRIZIONE SOFTWARE DELLE PERIFERICHE DI BORDO

Di seguito viene riportata una descrizione dettagliata della gestione software delle periferiche di bordo. Qualora la documentazione riportata fosse insufficiente fare riferimento direttamente alla documentazione tecnica della casa costruttrice del componente. In questo paragrafo inoltre non vengono descritte le sezioni che fanno parte del microprocessore; per quanto riguarda la programmazione di quest'ultime si faccia riferimento all'appendice A di questo manuale. Nei paragrafi successivi si usano le indicazioni **0÷7** e **.0÷7** per fare riferimento ai bits della combinazione utilizzata nelle operazioni di I/O.

LED DI ATTIVITA'

La **GMM AM08** consente la gestione software di un LED di attività o stato, DL1, tramite una linea di I/O del microprocessore con la seguente corrispondenza:

PB5 = 0	->	DL1 attivo
PB5 = 1	->	DL1 disattivo

Si ricorda che il segnale PB5 è disponibile su CN1 al pin 12.

Il segnale PB5 viene mantenuto alto dal microcontrollore dopo un reset o un power on, di conseguenza in seguito ad una di queste fasi il LED è disattivo.

PERIFERICHE DELLA CPU

La descrizione dei registri e del relativo significato di tutte le periferiche interne del microprocessore (COMPARATORE, A/D CONVERTERS, TMR CNT, USART, I²C BUS, SPI, ecc) é disponibile nell'apposito manuale tecnico e manuale d'uso della casa costruttrice.

Per ulteriori informazioni si vedano la BIBLIOGRAFIA e l'appendice A di questo manuale.

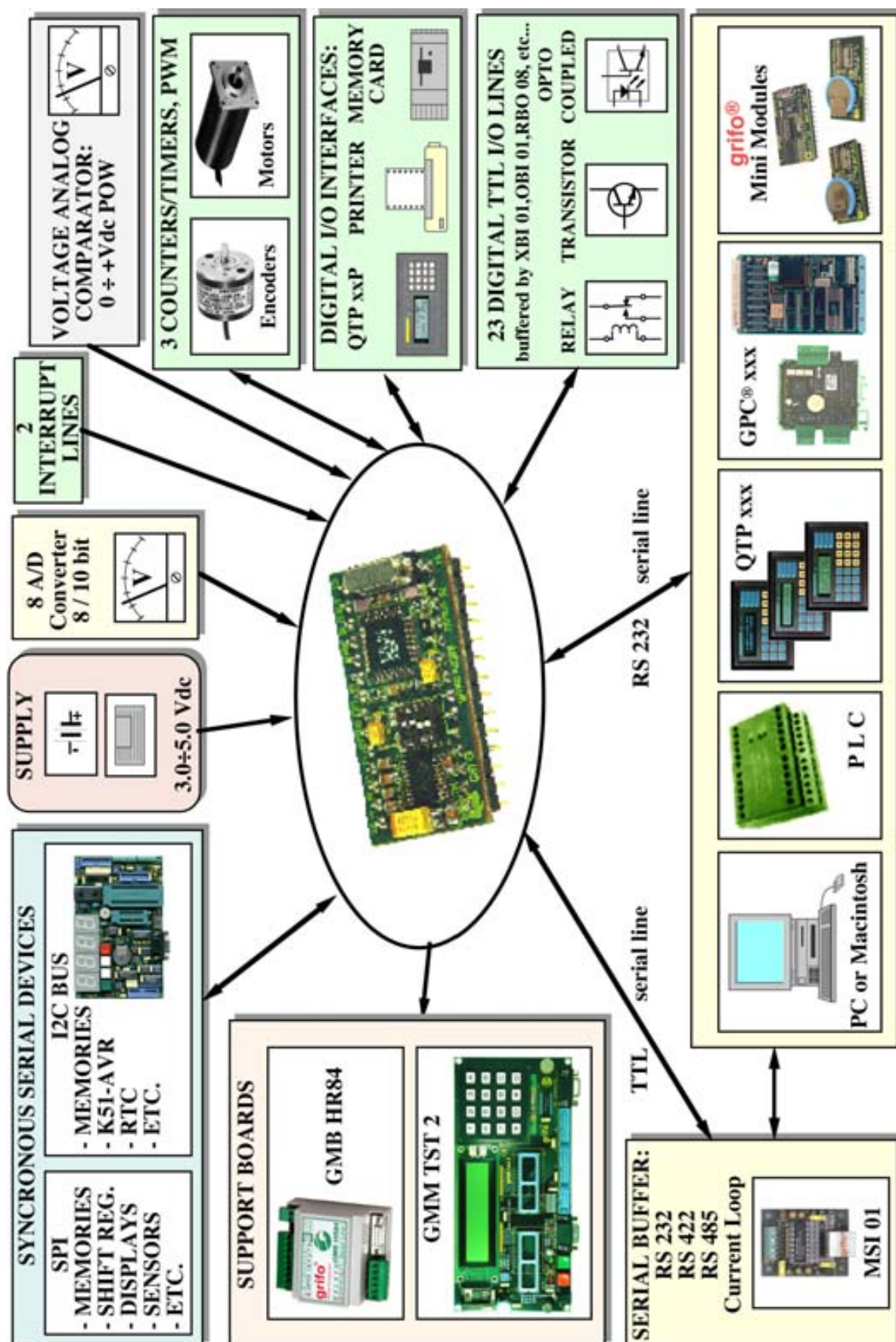


FIGURA 22: SCHEMA DELLE POSSIBILI CONNESSIONI

BIBLIOGRAFIA

E' riportato di seguito, un elenco di manuali e note tecniche, a cui l'utente può fare riferimento per avere maggiori chiarimenti, sui vari componenti montati a bordo del Mini Modulo **GMM AM08**.

Manuale MAXIM: *New Releases Data Book - Volume IV*

Manuale MAXIM: *New Releases Data Book - Volume V*

Manuale NATIONAL SEMICONDUCTOR: *Linear Databook - Volume 1*

Per reperire questi manuali fare riferimento alle case produttrici ed ai relativi distributori locali. In alternativa si possono ricercare le medesime informazioni o gli eventuali aggiornamenti ai siti internet delle case elencate.

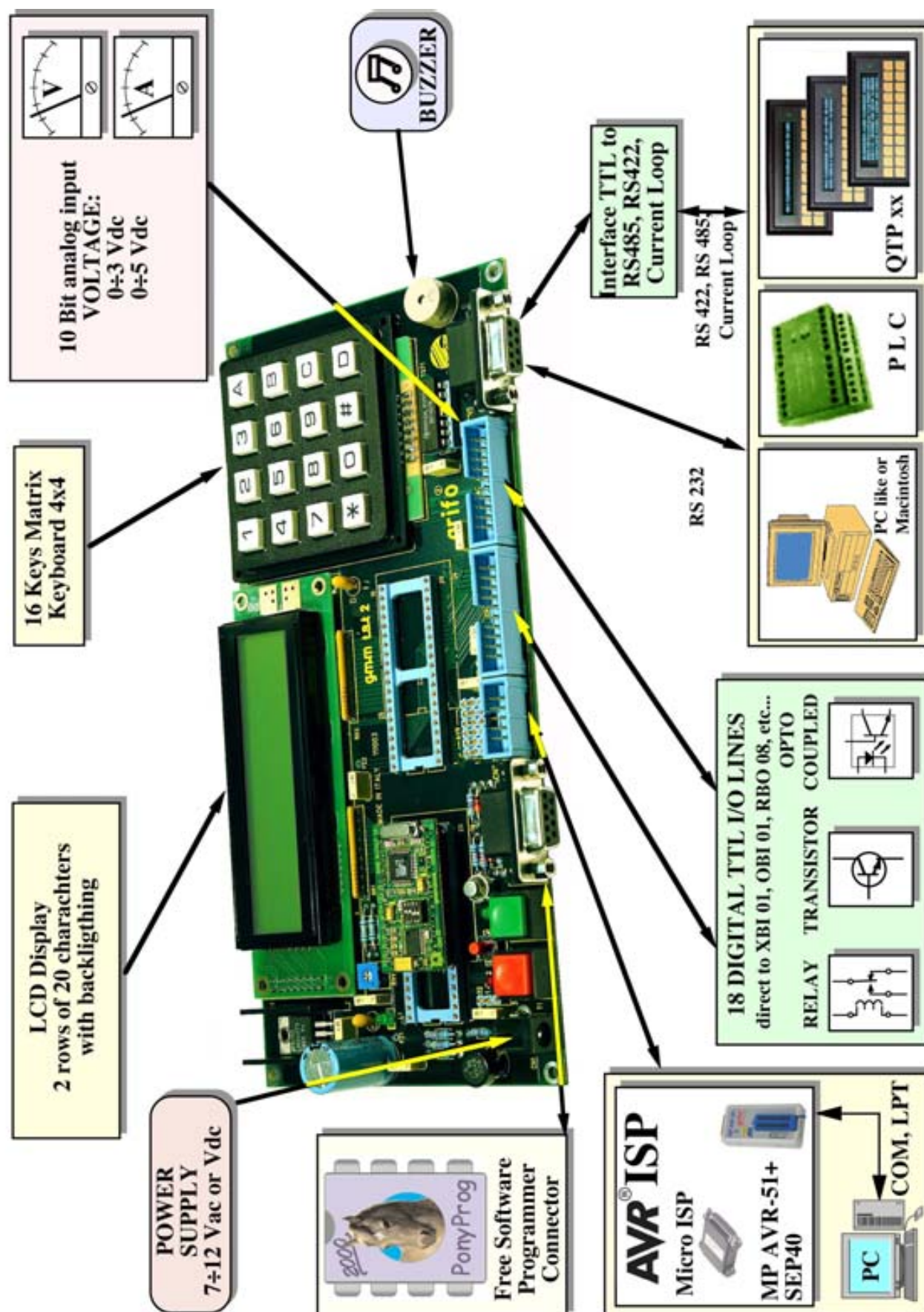


FIGURA 23: SCHEMA DELLE CONNESSIONI A GMM TST 2 CON GMM AM08

APPENDICE A: DESCRIZIONE COMPONENTI DI BORDO

La **grifo®** fornisce un servizio di documentazione tecnica totalmente gratuito attraverso il proprio sito internet in cui possono essere scaricati i data sheets completi dei componenti usati a bordo scheda. Si rimanda quindi l'utente a tali documenti, di cui viene riportato il percorso sia tramite i link che tramite l'URL completo, assieme alle prime pagine degli stessi documenti.

ATmega8L

Link: Home | Servizio Documentazione Tecnica | ATMEL | Data-Sheet ATmega8L

URL: <http://www.grifo.it/PRESS/DOC/Atmel/ATMEGA8L.pdf>

Features

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 130 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16 MHz
 - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
 - 8K Bytes of In-System Self-Programmable Flash
 - Endurance: 10,000 Write/Erase Cycles
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - 512 Bytes EEPROM
 - Endurance: 100,000 Write/Erase Cycles
 - 1K Byte Internal SRAM
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Three PWM Channels
 - 8-channel ADC in TQFP and MLF package
 - Six Channels 10-bit Accuracy
 - Two Channels 8-bit Accuracy
 - 6-channel ADC in PDIP package
 - Four Channels 10-bit Accuracy
 - Two Channels 8-bit Accuracy
 - Byte-oriented Two-wire Serial Interface
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-lead PDIP, 32-lead TQFP, and 32-pad MLF
- Operating Voltages
 - 2.7 - 5.5V (ATmega8L)
 - 4.5 - 5.5V (ATmega8)
- Speed Grades
 - 0 - 8 MHz (ATmega8L)
 - 0 - 16 MHz (ATmega8)
- Power Consumption at 4 Mhz, 3V, 25°C
 - Active: 3.6 mA
 - Idle Mode: 1.0 mA
 - Power-down Mode: 0.5 µA



8-bit AVR®
with 8K Bytes
In-System
Programmable
Flash

ATmega8
ATmega8L

Preliminary

Rev. 2486J-AVR-02/03

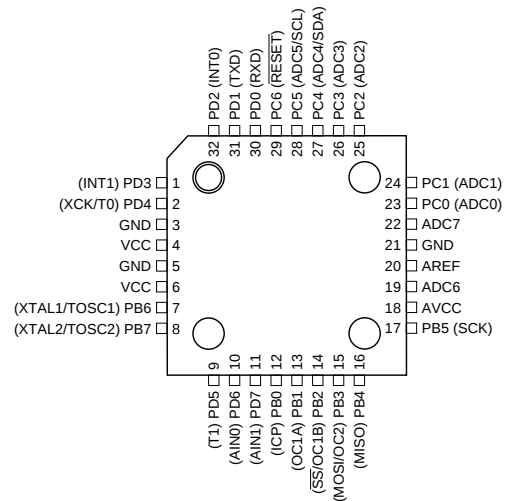


Pin Configurations

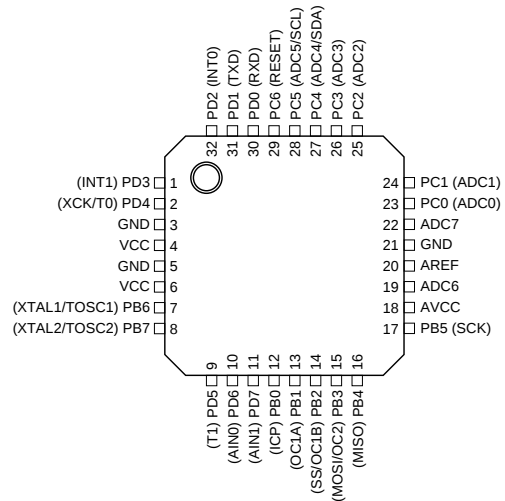
PDIP

(RESET) PC6	1	28	PC5 (ADC5/SCL)
(RXD) PD0	2	27	PC4 (ADC4/SDA)
(TXD) PD1	3	26	PC3 (ADC3)
(INT0) PD2	4	25	PC2 (ADC2)
(INT1) PD3	5	24	PC1 (ADC1)
(XCK/T0) PD4	6	23	PC0 (ADC0)
VCC	7	22	GND
GND	8	21	AREF
(XTAL1/TOSC1) PB6	9	20	AVCC
(XTAL2/TOSC2) PB7	10	19	PB5 (SCK)
(T1) PD5	11	18	PB4 (MISO)
(AIN0) PD6	12	17	PB3 (MOSI/OC2)
(AIN1) PD7	13	16	PB2 (SS/OC1B)
(ICP) PB0	14	15	PB1 (OC1A)

TQFP Top View



MLF Top View



APPENDICE B: PROGRAMMAZIONE ISP CON GMM TST 2

In questa appendice vengono date le istruzioni specifiche per poter programmare il Mini Modulo **GMM AM08** a bordo di una **GMM TST 2** usando un programmatore ISP esterno.

Si tratta di un metodo di programmazione della FLASH e della EEPROM del Mini Modulo alternativo a quello presentato nella sezione "B" del capitolo "Come Iniziare" del manuale.

Le altre voci del suddetto capitolo rimangono comunque valide.

La programmazione ISP permette di modificare anche i bit di configurazione ed i fusibili di sicurezza, oltre a permettere di programmare completamente sia la FLASH che la EEPROM.

L'utilizzo del Boot Loader non permette di programmare completamente la FLASH, in quanto la sezione in cui risiede il Boot Loader stesso è protetta.

Inoltre i bit di configurazione, che tra l'altro abilitano il Boot Loader stesso, ed i fusibili di sicurezza sono al di fuori della portata del Boot Loader e devono necessariamente essere programmati in modalità ISP.

Se si ritiene più conveniente o comunque sufficiente usare un Boot Loader su linea seriale RS 232, si faccia riferimento alla suddetta sezione.

Per informazioni su come collegare una porta seriale del PC alla **GMM TST 2** (ed anche al Mini Modulo da solo) per provare il programma, si faccia riferimento alla sezione "A" del capitolo "Come Iniziare" del manuale.

Per informazioni su come sviluppare e debuggare il firmware, si faccia riferimento alla sezione "C" del capitolo "Come Iniziare" del manuale.

I programmatori utilizzabili sono:

- **grifo® MP-AVR 51+** ; **grifo® UEP 49**
- Equinox EPSILON5
- ATMEL AVR ISP
- PonyProg

Le informazioni qui riportate vengono inoltre integrate con degli schemi di connessione diretta (presenti in pagina B-3 e B-5) che permettono all'utente di costruire il proprio hardware di supporto ISP o di integrare il supporto ISP nel proprio progetto.

A) OPERAZIONI COMUNI PER TUTTI I PROGRAMMATORI

A1) La programmazione del Mini Modulo a bordo della **GMM TST 2** avviene sfruttando un apposito connettore della scheda ed usando eventualmente un adattatore di interfacciamento tra il programmatore ed il connettore stesso.

Per i due programmatori **grifo®** che si possono interfacciare con **GMM TST 2**, ovvero **MP-AVR 51+** e **UEP 49**, lo schema di collegamento delle interfacce viene mostrato rispettivamente nelle figure B-1 e B-2. Sul lato sinistro delle figure si trova l'elenco delle vie del connettore CN7 di **GMM TST 2** che vanno collegati ai pin del connettore del programmatore indicati con la freccia.

Per i programmatori AVR ISP ed Equinox EPSILON5 non c'è bisogno di alcuna interfaccia specifica.

- A2) Inserire il Mini Modulo **GMM AM08** in uno degli zoccoli a 40 vie della **GMM TST 2** allineandolo verso il basso, come indicato dalla serigrafia e mostrato nella figura 12 del manuale.
- A3) Localizzare e salvare in una posizione comoda sul disco rigido del PC il file si chiama "d_am08.HEX" seguendo il percorso: Italiano | Programmi di Esempio | Programmi Mini Moduli e Mini Block | GMM AM08 (fare riferimento alla figura 14 del manuale). Nei punti successivi viene descritta la configurazione dettagliata per ogni singolo programmatore.

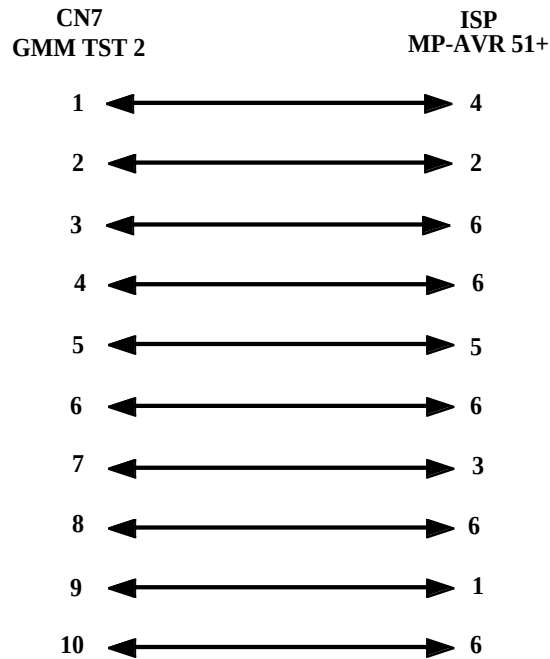


FIGURA B-1: SCHEMA DELL'INTERFACCIA TRA GMM TST 2 E MP-AVR 51+

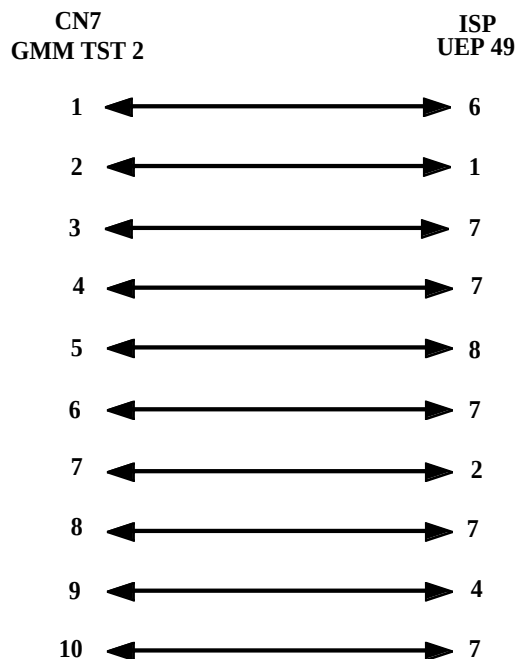


FIGURA B-2: SCHEMA DELL'INTERFACCIA TRA GMM TST 2 E UEP 49

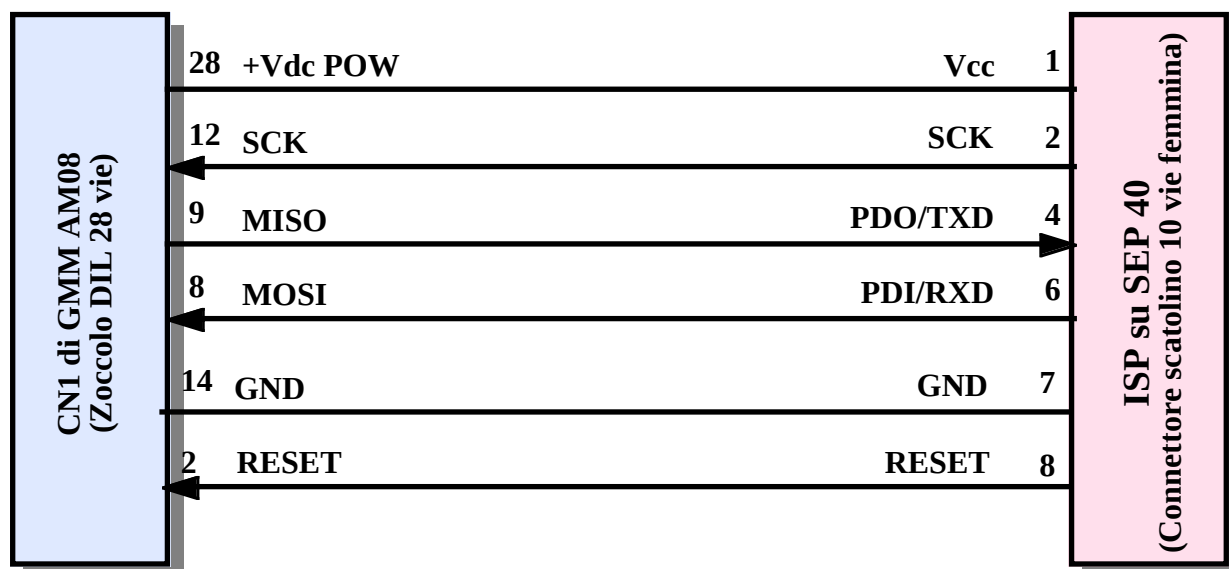


FIGURA B-3: SCHEMA DI CONNESSIONE DIRETTA TRA GMM AM08 E SEP 40

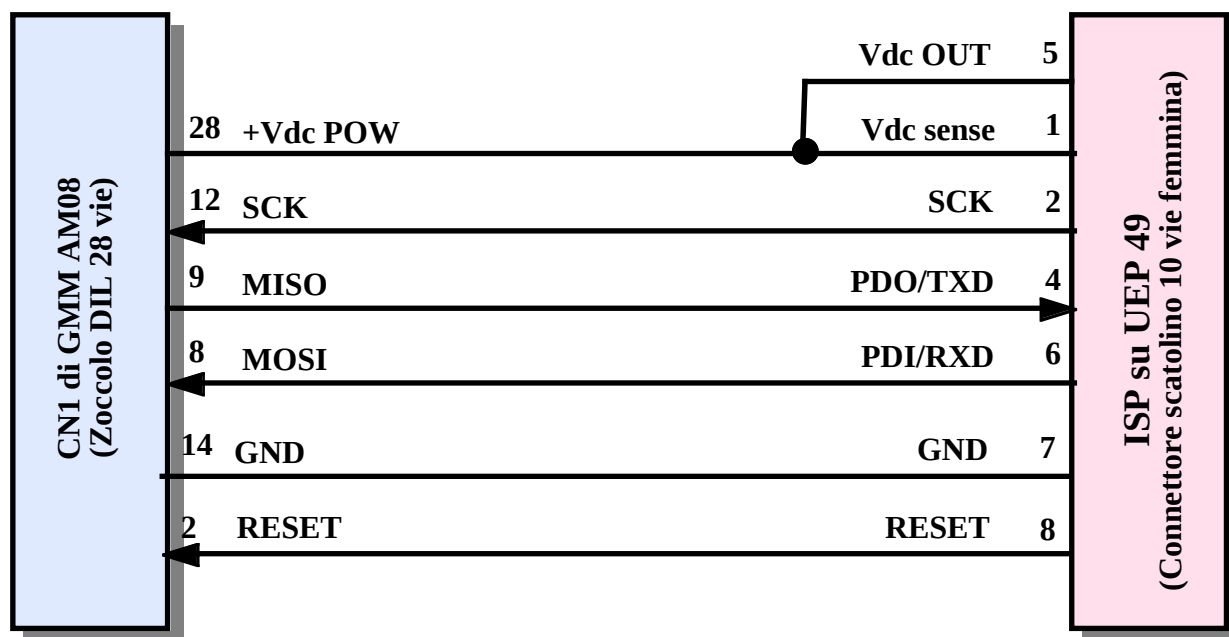


FIGURA B-4: SCHEMA DI CONNESSIONE DIRETTA TRA GMM AM08 E UEP 49

B) PROGRAMMAZIONE CON GRIFO® MP-AVR 51+ ; GRIFO® UEP 49

B1) Per installare, collegare al PC ed utilizzare il programmatore, si prega di fare riferimento al manuale fornito con il programmatore stesso. Nella figura di seguito viene mostrata un esempio di schermata iniziale del programma di controllo dei programmatori **grifo®**. Questo programma si chiama PG4UW e verrà indicato con tale nome nel seguito.

B2) Collegare il programmatore alla **GMM TST 2** usando l'interfaccia descritta al punto A1.

B3) Inserire il Mini Modulo in uno zoccolo di **GMM TST 2** come descritto nel punto A2, poi alimentare la **GMM TST 2** e configurare quest'ultima come se si dovesse usare un programmatore **AVR ISP**(vedere manuale **GMM TST 2**). .

B4) Nel PG4UW, selezionare come componente da programmare ATmega8L in modalità ISP, come indicato in figura B-5.

B5) Caricare il file precedentemente salvato al punto A3 usando il pulsante "Carica".

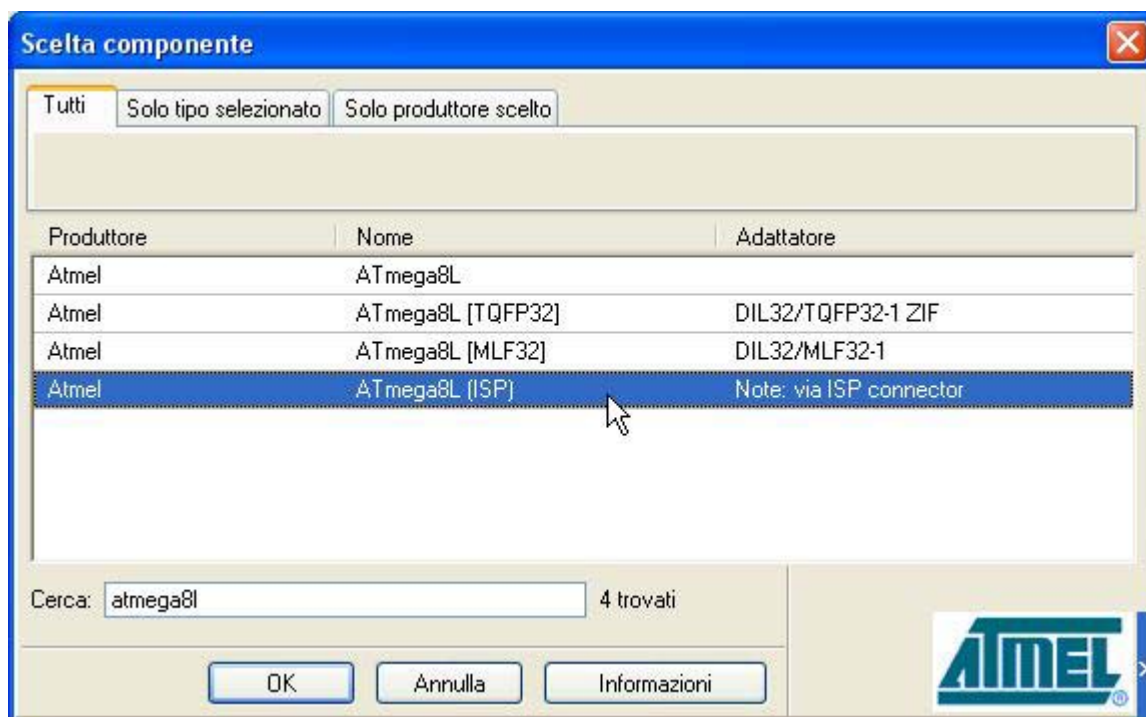


FIGURA B-5: SELEZIONE DEL COMPONENTE CON PG4UW

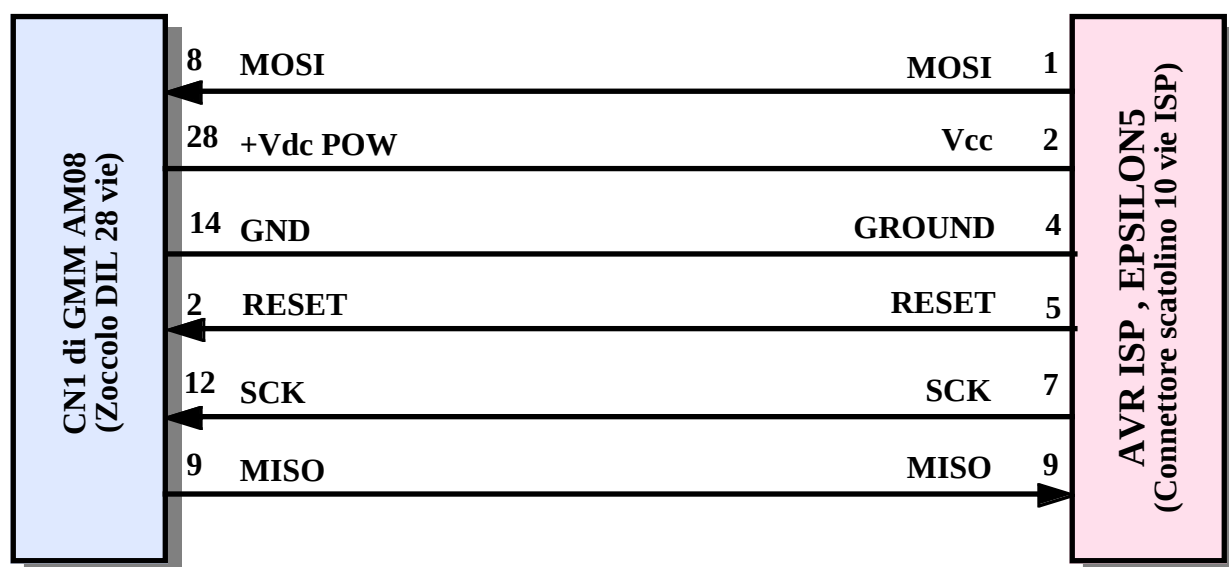


FIGURA B-6: SCHEMA DI CONNESSIONE DIRETTA TRA GMM AM08 ED EPSILON5 o AVR ISP

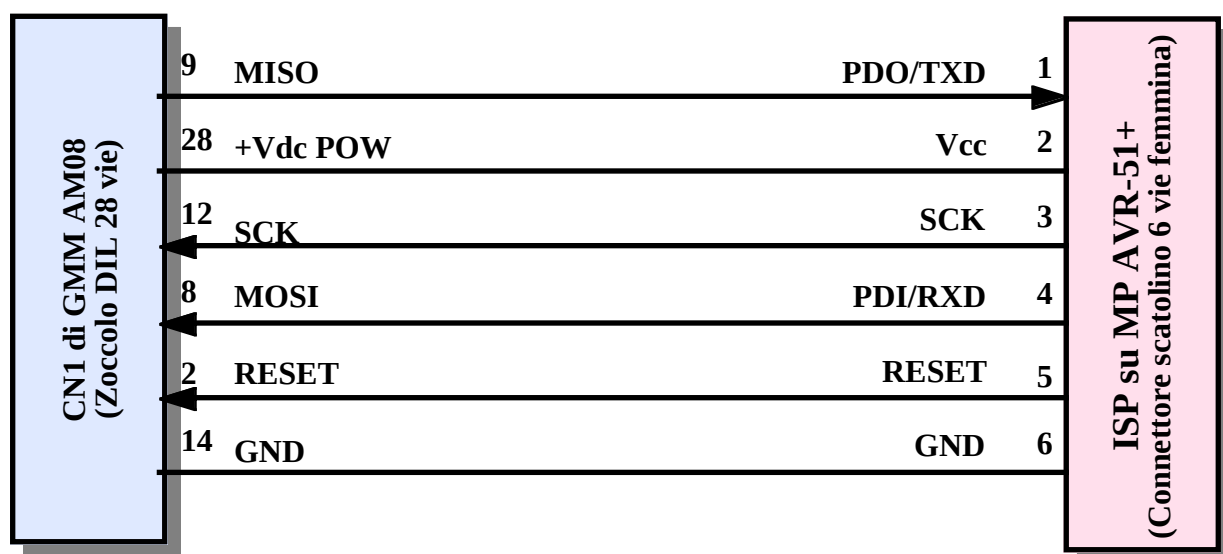


FIGURA B-7: SCHEMA DI CONNESSIONE DIRETTA TRA GMM AM08 E MP AVR-51+

B6) Configurare il componente usando il sottomenu Opzioni componente | View/Edit options and security. Il componente deve essere configurato come indicato in figura B-8, ovvero per fare funzionare il demo in d_am08.HEX il quarzo ed il tempo di avvio devono essere configurati come nella figura. Le altre impostazioni sono irrilevanti. Ovviamente, l'utente può effettuare la configurazione che preferisce.

B7) Configurare il programmatore usando il sottomenu Opzioni componente | Opzioni operative. Il programmatore deve essere configurato come indicato in figura B-9.

B8) Avviare la programmazione, premendo il pulsante "Prog.".

B9) Durante la programmazione la barra progressiva avanza fino al completamento. In caso di errori, questi vengono segnalati immediatamente.

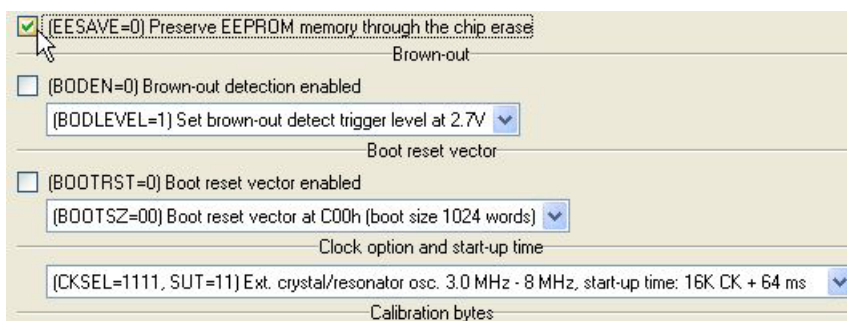


FIGURA B-8: CONFIGURAZIONE DEL COMPONENTE CON PG4UW

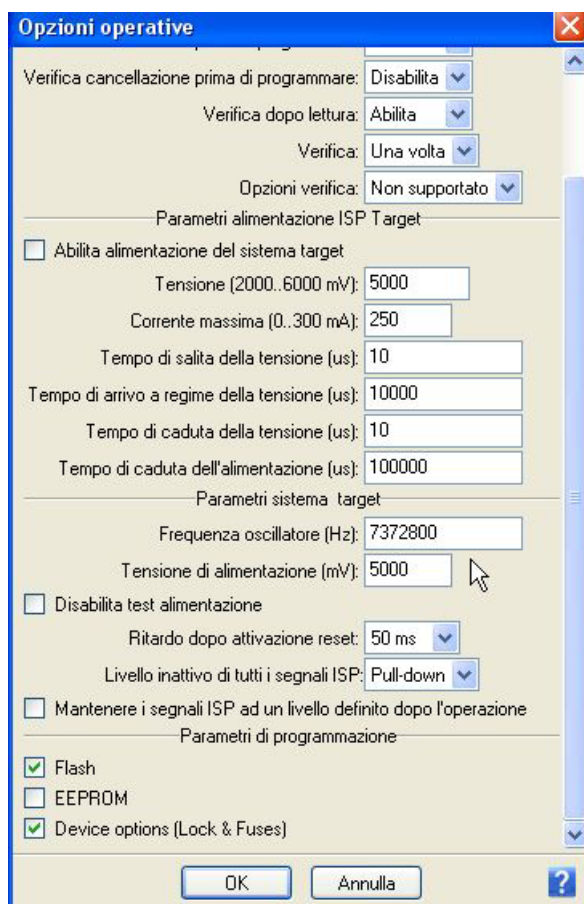


FIGURA B-9: CONFIGURAZIONE DEL PROGRAMMATORE PG4UW

C) PROGRAMMAZIONE CON EQUINOX EPSILON5

C1) La programmazione del Mini Modulo a bordo della **GMM TST 2** avviene sfruttando un apposito connettore della scheda. Il programmatore Equinox EPSILON5 deve innanzitutto essere configurato fisicamente per collegarsi alla **GMM TST 2**. Per effettuare tale configurazione è necessario:

- Collegare il cavo flat da 10 vie al connettore "J7-ATMEL10" dell'EPSILON5
- Chiudere il jumper J9 dell'EPSILON5

per ulteriori informazioni si prega di fare riferimento al manuale dell'EPSILON5.

C2) Dopo avere configurato e richiuso l'EPSILON5, è necessario installare il suo programma di controllo, chiamato EQTools. Per ulteriori informazioni su come installare il programma e collegare l'EPSILON5 al PC, si faccia riferimento al manuale utente del suddetto.

C3) Collegare il programmatore al connettore CN7 della **GMM TST 2** e configurare quest'ultima come se si dovesse usare un programmatore **AVR ISP**(vedere manuale **GMM TST 2**).

C4) Inserire il Mini Modulo nello zoccolo Z1 o Z2 di **GMM TST 2** usando l'interfaccia e le istruzioni descritte ai punti A, poi alimentare la **GMM TST 2** come descritto nel manuale della **GMM TST 2** stessa.

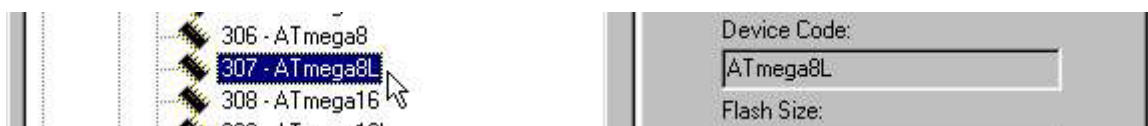
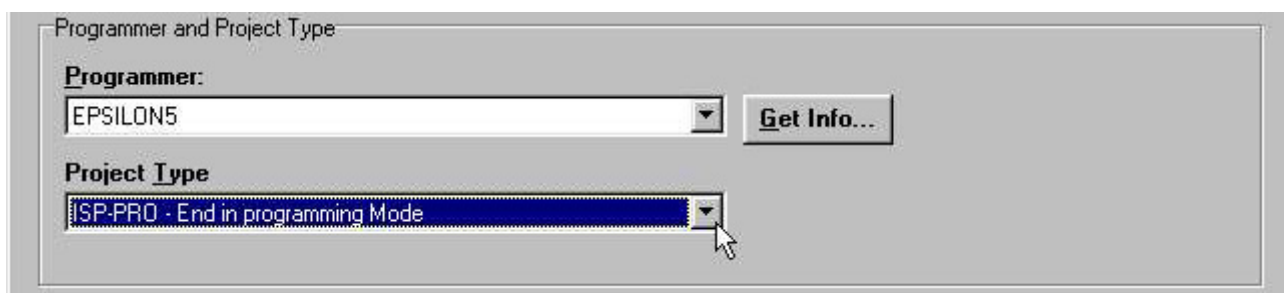
C5) Per usare l'Equinox EPSILON5 collegato al PC(anziché stand-alone) bisogna aprire un file di progetto (estensione EDS). Questo si può fare creandone uno nuovo dalla schermata iniziale che presenta EQTools all'avvio, usando gli appositi menu e pulsanti o caricando un progetto esistente. Per ulteriori informazioni sulla gestione sui file di progetto, si prega di fare riferimento al manuale di EQTools.

C6) Se si crea un nuovo progetto, bisogna assicurarsi che siano eseguite le impostazioni mostrate nelle figure in posizione B-6. Se si apre un progetto esistente, bisogna assicurarsi che siano presenti. Le impostazioni servono a garantire che:

- Il progetto venga eseguito correttamente da un EPSILON5
- Il componente target sia un ATmega8L
- Il file che viene programmato sul target sia d_am08.HEX
- La configurazione del componente target sia corretta

per ulteriori informazioni sulla configurazione di un progetto, si faccia riferimento alla documentazione di EQTools.

- C7) Una volta completata la configurazione del progetto, se lo si sta creando dal nulla, bisogna indicare di usarlo in modalità "Test EDS". Per attivare l'operazione di scrittura nella memoria Flash del Mini Modulo, bisogna tornare al menu della Flash, mettere lo spunto alla casella "Edit Menu" e premere il pulsante "Write".
- C8) La finestra "Write Block to Flash" fornisce un riassunto di alcune delle impostazioni correnti, per verificarle. Se risultano corrette, premendo OK inizia il procedimento fisico di scrittura nella memoria.
- C9) Lo stato della programmazione viene indicato dalla barra progressiva, al termine una finestra indica l'esito finale dell'operazione.
- C10) Per scrivere i bit di configurazione del componente, bisogna tornare al menu apposito ("Fuses") e premere il pulsante "Write" nel riquadro "Target Fuses".



Fuses	PC Fuse State	Target Fuse State	Description
CKSEL0	1 - UNPROGRAMMED	?	Clock Select 0
CKSEL1	1 - UNPROGRAMMED	?	Clock Select 1
CKSEL2	1 - UNPROGRAMMED	?	Clock Select 2
CKSEL3	1 - UNPROGRAMMED	?	Clock Select 3
SUT0	1 - UNPROGRAMMED	?	Select Start Up Time 0
SUT1	1 - UNPROGRAMMED	?	Select Start Up Time 1
BODEN	1 - UNPROGRAMMED	?	Brown out Detector - ENABLE
BODLEVEL	1 - UNPROGRAMMED	?	Brownout Level - select trip volta...
BOOTRST	1 - UNPROGRAMMED	?	RESET VECTOR
BOOTSZ0	0 - PROGRAMMED	?	Boot Size 0
BOOTSZ1	0 - PROGRAMMED	?	Boot Size 1
EESAVE	0 - PROGRAMMED	?	Preserve EEPROM during ERAS...
CKOPT	1 - UNPROGRAMMED	?	Clock Oscillator Options

FIGURA B-10: CONFIGURAZIONE PROGETTO CON EQTOOLS

D) *PROGRAMMAZIONE CON ATMEL AVR ISP*

- D1) Il programma di controllo dell'**AVR ISP** è **AVR STUDIO**, versione 4 o superiori. L'ultima versione è scaricabile dal sito Atmel www.atmel.com. Scaricatelo e installatelo seguendo le istruzioni a video.
- D2) Configurare **AVR ISP** per usare il flat da 10 vie e collegarlo a CN7 di **GMM TST 2**, collegare **AVR ISP** alla porta seriale del PC (vedere istruzioni di **AVR ISP**), configurare la **GMM TST 2** per programmare con **AVR ISP**, seguendo le istruzioni dei punti A, ed alimentarla (vedere manuale **GMM TST 2**).
- D3) Lanciare **AVR STUDIO**. Il programma di controllo di **AVR ISP** si attiva premendo il pulsante con il chip AVR come icona.
- D4) Selezionare come CPU ATmega8L, come indicato in figura B-11.
- D5) Caricare il file precedentemente salvato al punto A3 usando il pulsante "Carica".
- D6) Configurare la CPU come indicato nelle immagini di figura B-12.
- D7) Configurare il programmatore per effettuare la verifica dell'ID, cancellare il dispositivo e riprogrammare con verifica la FLASH e i bit di configurazione, come in figura B-13.
- D8) Eseguire la programmazione premendo il tasto "Start" come indicato nella figura B-13.

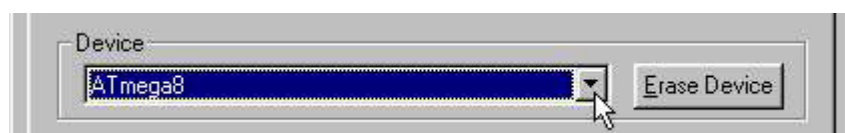


FIGURA B-11: SELEZIONE CPU CON AVR STUDIO

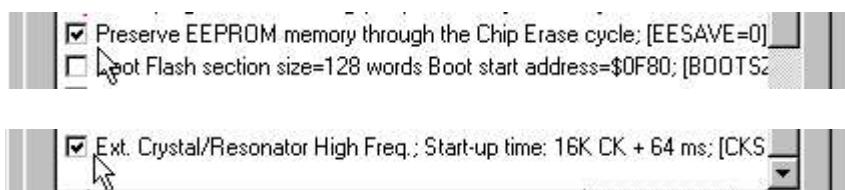


FIGURA B-12: CONFIGURAZIONE CPU CON AVR STUDIO

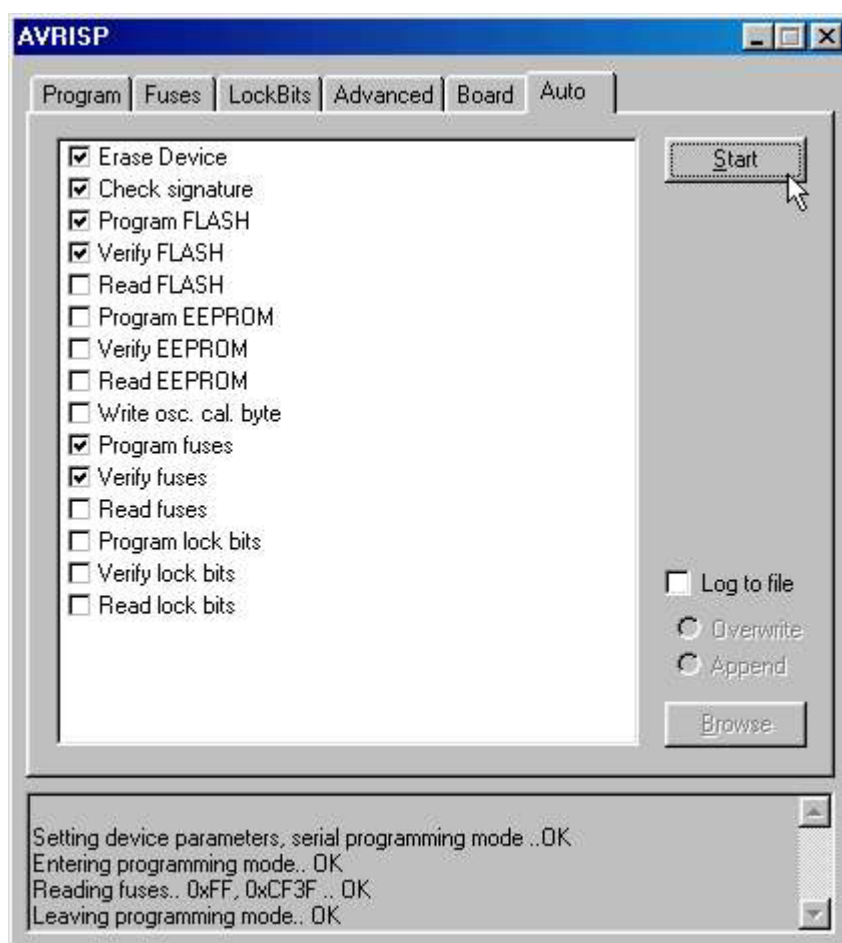


FIGURA B-13: CONFIGURAZIONE AVR ISP

E) PROGRAMMAZIONE CON PONY PROG

- E1) Il programma **PonyProg** permette di programmare la **GMM AM08** a bordo di una **GMM TST 2** semplicemente collegando la porta seriale del PC al connettore CN6. La versione 2.06c (che programma il microcontrollore Atmel ATmega8L) si trova anche sul CD **grifo®**, l'ultima versione è scaricabile dal sito www.lancos.com. Scaricatelo e installatelo seguendo le istruzioni a video.
- E2) Collegare CN6 della **GMM TST 2** alla porta seriale del PC, configurarla per la programmazione con **Pony Prog** e alimentarla (vedere manuale **GMM TST 2**).
- E3) Lanciare **Pony Prog** ed effettuare la calibrazione mediante il menu Setup | Calibration.
- E4) Selezionare la libreria di comunicazione SI Prog API con il menu Setup | Interface setup.
- E5) Selezionare dalle apposite caselle "AVR micro" e "ATmega8" (vedere figura B-15).
- E6) Aprire il file "d_am08.hex" precedentemente salvato.
- E7) Configurare la CPU per preservare il contenuto della EEPROM in fase di cancellazione e utilizzare un quarzo esterno ad alta frequenza, come indicato in figura B-16.
- E8) Configurare il programmatore per effettuare la verifica dell'ID, cancellare il dispositivo e riprogrammare con verifica la FLASH e i bit di configurazione.
- E9) Effettuare la programmazione premendo il pulsante indicato in figura B-17 e verificare che al termine delle varie fasi (che possono durare anche decine di secondi) compaia il messaggio "Program Succesful".



FIGURA B-14: LOGO DEL PONYPROG



FIGURA B-15: SELEZIONE MICROCONTROLLORE CON PONYPROG



FIGURA B-16: CONFIGURAZIONE MICROCONTROLLORE CON PONYPROG



FIGURA B-17: AVVIO DELLA PROGRAMMAZIONE CON PONYPROG

APPENDICE C: SCHEMA ELETTRICO GMM TST 2

In questa appendice sono disponibili gli schemi elettrici della scheda di supporto **GMM TST 2** che illustra alcune modalità di connessione dei segnali dei Mini Moduli. Informazioni più dettagliate su questa scheda sono disponibili nel relativo manuale tecnico e l'utente le può usare liberamente ad esempio per realizzare una propria scheda che usa la **GMM AM08** come macro componente.

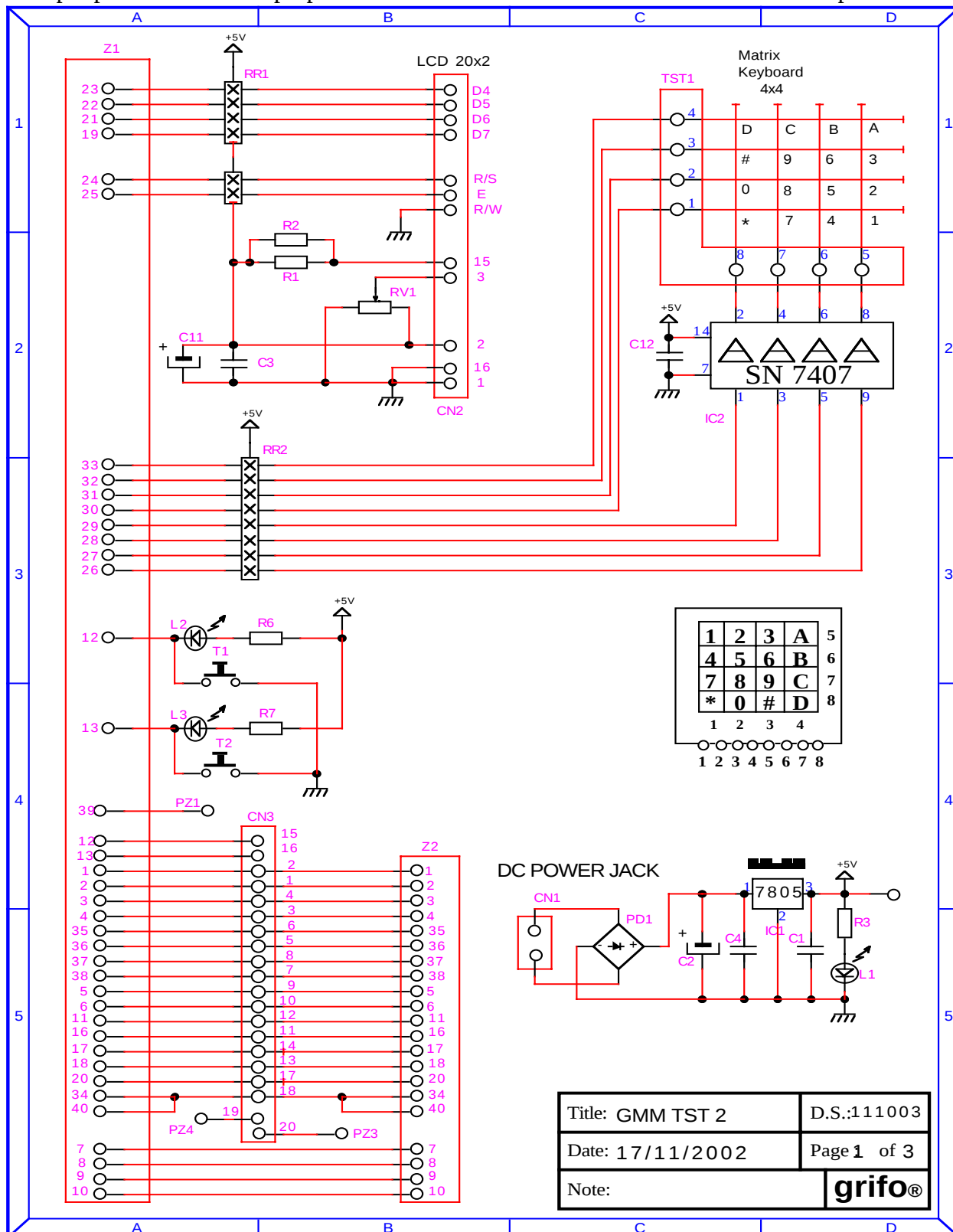


FIGURA C-1: SCHEMA ELETTRICO GMM TST 2 (1 di 3)

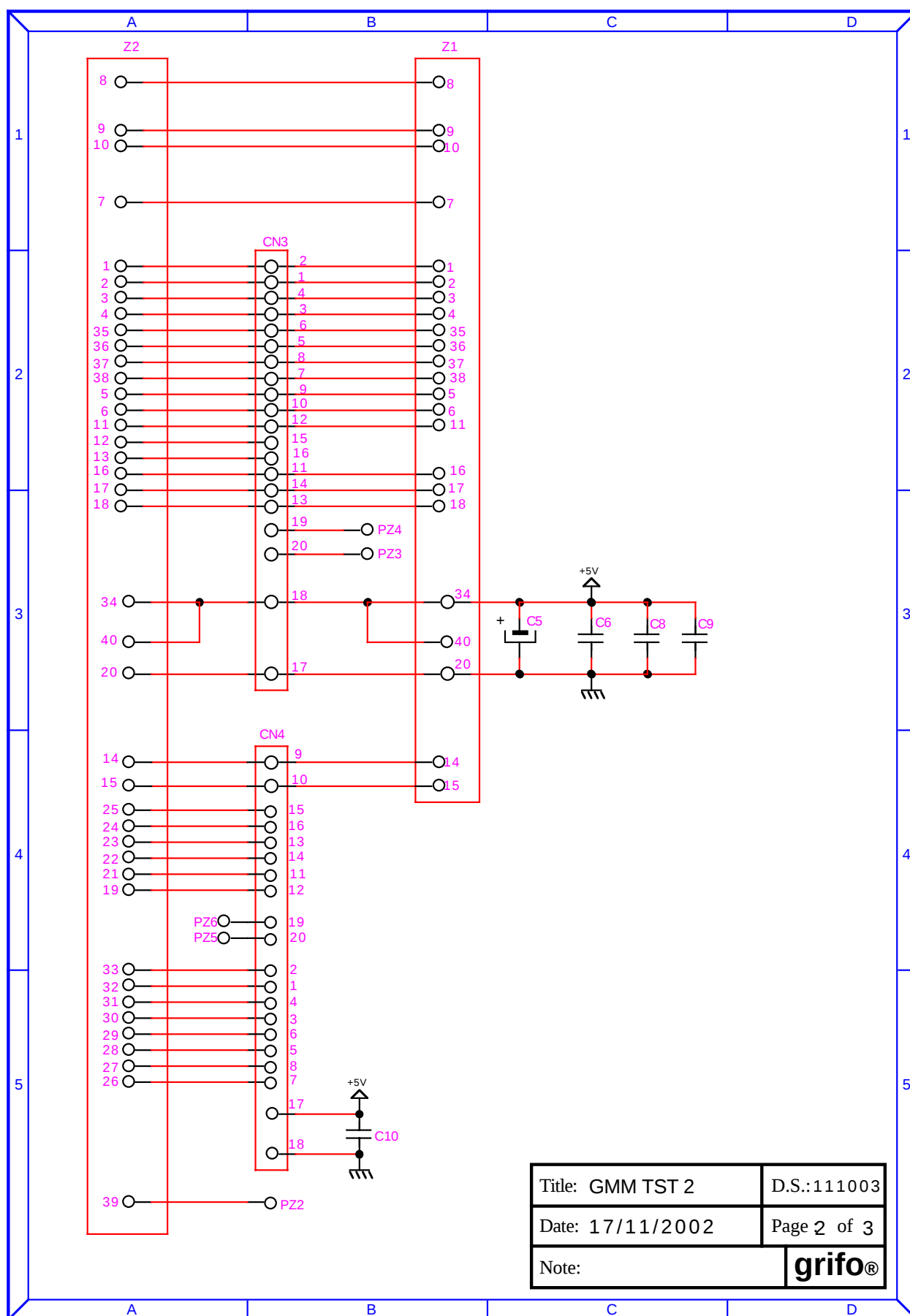


FIGURA C-2: SCHEMA ELETTRICO GMM TST 2 (2 DI 3)

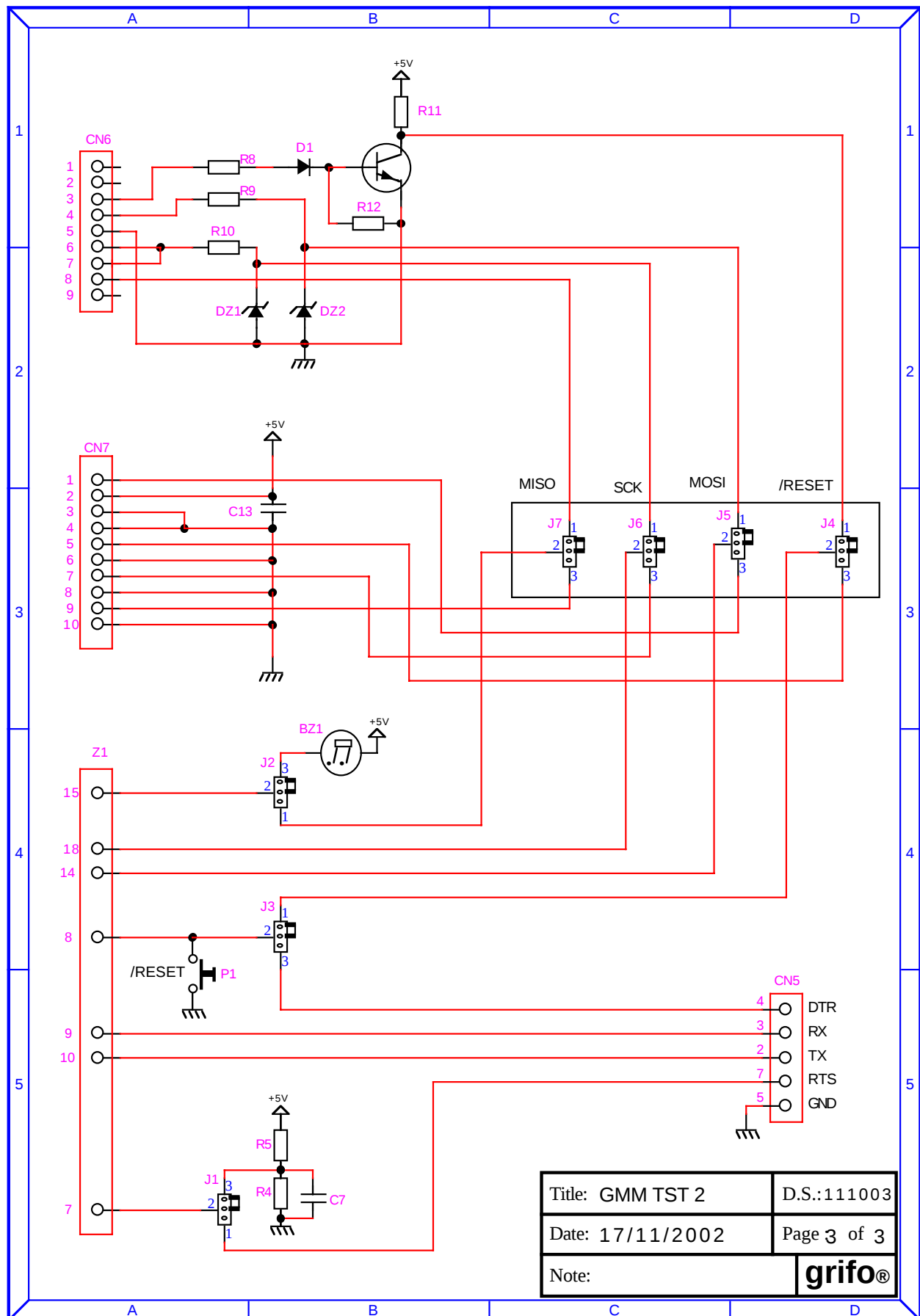


FIGURA C-3: SCHEMA ELETTRICO GMM TST 2 (3 DI 3)

APPENDICE D: INDICE ANALITICO

SIMBOLI**+VDC POW 13, 18****A****A/D CONVERTER 6, 10, 13, 18****ALIMENTAZIONE 11, 13****AVR ISP B-7****AVRBOOTLOADERGRIFO 26****B****BASCOM AVR 28****BIBLIOGRAFIA 36****BROWN OUT 11****C****CLOCK 9****COMPARATORE ANALOGICO 13, 18****CONNETTORI 10****CONSUMO DI CORRENTE 11****CPU 10, 34****CURRENT LOOP 8****D****DIMENSIONI 10****DIP SWITCH 6, 10, 14****E****EQUINOX B-5****EEPROM 9, 10****F****FLASH 9, 10****FREQUENZA DI CLOCK 10****FREQUENZA MASSIMA CONTATORI 10****G****GMB HR84 20****GMM TST 2 22, B-1, C-1**

I

I/O DIGITALE 6, 10, 13

I²C BUS 8, 13, 18

ICC AVR 30

IMPEDENZA GENERATORI SEGNALI ANALOGICI 11

INT0 13, 19

INT1 13, 19

INTERRUPT 10, 13, 19

L

LED 10, 12, 34

M

MEMORIA 9

MP-AVR 51+ B-1, B-3

MSI 01 8

O

OCM 18

P

PESO 10

PONY PROG B-9

POWER ON 10

PROTEZIONE RS 232 11

PULL-UP SU I²C BUS 11

PWM 8, 18

Q

QTP 8

R

RANGE DI TEMPERATURA 10

RESET 10, 13

RISOLUZIONE A/D 10

RS 232 8, 10, 13, 14, 16, 18

RS 422 8

RS 485 8

S

SPI 6, 13, 18

SRAM 9, 10

T

TIMER/COUNTER 8, 10, 13, 18

TTL 6, 8, 10, 13, 14, 16, 18

U

UEP 49 B-1, B-3

UMIDITÀ RELATIVA 10

V

VERSIONE SCHEDA 3

W

WATCH DOG 6, 10

