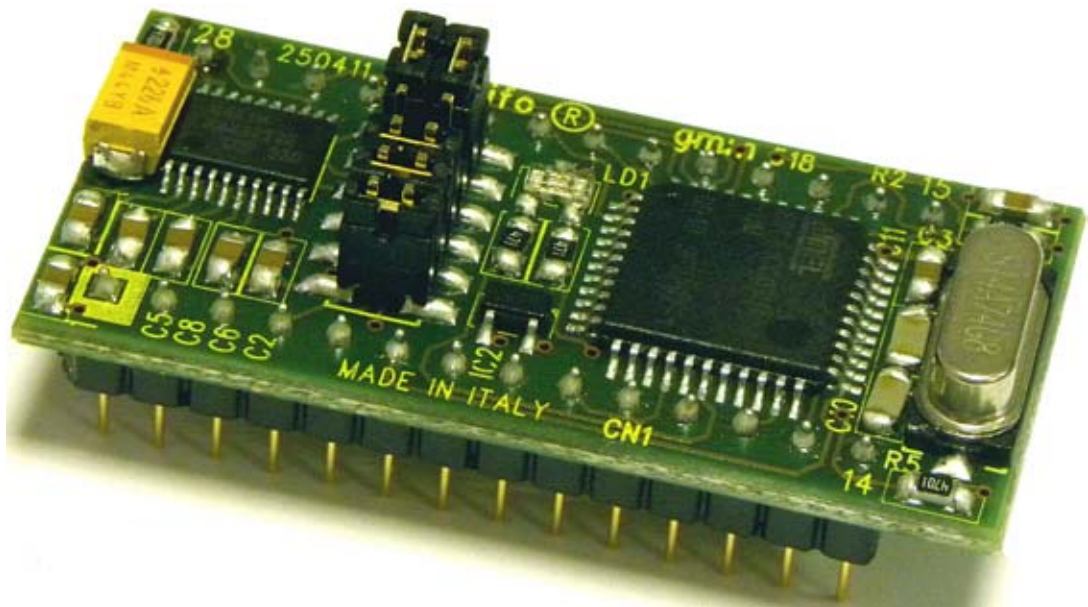


# GMM 518 Zero

grifo® Mini Modulo Atmel AT89C51CC03

## MANUALE UTENTE



**grifo®**

ITALIAN TECHNOLOGY

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

E-mail: [grifo@grifo.it](mailto:grifo@grifo.it)

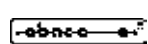
<http://www.grifo.it>

<http://www.grifo.com>

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



GMM 518 Zero Rel. 3.00 Edizione 12 Settembre 2011

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



# GMM 518 Zero

grifo® Mini Modulo Atmel AT89C51CC03

## MANUALE UTENTE

Zoccolo maschio **28** piedini dual in line a passo **100** mils, largo **600** mils; ridottissimo ingombro: **20,6 x 42,6 x 15 mm**; circuito stampato a **4** strati per ottimizzare le immunità e le caratteristiche **EMI**; Unica alimentazione a **+5Vdc 10 mA**; disponibilità di modalità operative a **Basso Consumo**; microcontrollore **Atmel AT89C51CC03** (codice compatibile con **8051**) con quarzo da **14,74 Mhz**; Ciclo macchina programmabile a **12** o **6** periodi di clock; **64K FLASH** per codice, **2K FLASH** per Boot Loader, **256 bytes IRAM** per dati, **2K bytes ERAM** per dati, **2K EEPROM** per dati. **8** canali di **A/D Converter** con **10 bit** di risoluzione, **20 µsec** per ogni conversione; **19** sorgenti di Interrupt con **4** livelli di priorità; **3 Timer Counter** a **16 bits**; **5** canali **PCA** a **16 bit** con funzionalità di **PWM**, comparazione, ecc.; **24** linee di **I/O** digitale collegate al connettore; alcune di queste linee hanno funzionalità multiple. Linea seriale hardware con **Baud Rate** programmabile fino a **115200 Baud**, bufferata in **TTL** o in **RS 232** con protezione contro scariche da **±15 KV**. Circuiteria di **Reset** e controllo alimentazione. Linea **I2C BUS** software, riportata sul connettore; controllore **CAN** compatibile con standard **2.0A** e **2.0B**. Jumpers di configurazione da **5** vie, di cui **1** acquisibili via Firmware; **1 LED** di segnalazione gestiti via Firmware tramite **I/O** digitali.

Gestione della **FLASH** ed **EEPROM** interna in modalità **In System Programming**, ovvero con modulo già montato, sfruttando la linea di comunicazione seriale. **Software** gratuito per **PC** di supporto alla programmazione **ISP**. Vasta disponibilità di software di sviluppo: compilatori **C** (**µC/51**, **SYS51CW**, **DDS Micro C51**); compilatori **BASIC** (**BASCOM 8051**); compilatori **PASCAL** (**SYS51PW**); logica a **Contatti** (**LADDER WORK**), ecc.. Ricca serie di programmi dimostrativi ed esempi di utilizzo forniti sotto forma di sorgenti ampiamente commentati ed eseguibili, 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>

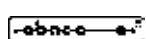
Tel. +39 051 892.052 (r.a.)

<http://www.grifo.com>

FAX: +39 051 893.661



GMM 518 Zero Rel. 3.00 Edizione 12 Settembre 2011

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

## Vincoli sulla documentazione **grifo**<sup>®</sup> Tutti i Diritti Riservati

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**<sup>®</sup>.

### IMPORTANTE

Tutte le informazioni contenute sul presente manuale sono state accuratamente verificate, ciononostante **grifo**<sup>®</sup> 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**<sup>®</sup> 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**<sup>®</sup>.

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

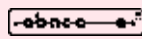


Attenzione: Pericolo di alta tensione



Attenzione: Dispositivo sensibile alle cariche elettrostatiche

### MARCHI REGISTRATI

, **GPC**<sup>®</sup>, **grifo**<sup>®</sup> : sono marchi registrati della **grifo**<sup>®</sup>.

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

# INDICE GENERALE

INTRODUZIONE .....	1
VERSIONE SCHEDA .....	3
NOTA SUL NOME MODULO .....	3
INFORMAZIONI GENERALI .....	4
COMUNICAZIONE SERIALE .....	5
LINEE DI I/O DIGITALE TTL .....	6
LINEA I2C BUS .....	6
LINEE ANALOGICHE DI A/D CONVERTER .....	6
MEMORIE .....	8
WATCH DOG .....	8
CLOCK .....	8
CONFIGURAZIONE SCHEDA .....	9
CONTROLLORE CAN .....	10
SETTORI DI UTILIZZO .....	10
CARATTERISTICHE TECNICHE .....	12
CARATTERISTICHE GENERALI .....	12
CARATTERISTICHE FISICHE .....	12
CARATTERISTICHE ELETTRICHE .....	13
INSTALLAZIONE .....	14
SEGNALAZIONI VISIVE .....	14
ALIMENTAZIONE .....	14
CONNESSIONI .....	15
CN1 - CONNETTORE CON SEGNALI DEL MINI MODULO .....	16
RESET, CONTROLLO ALIMENTAZIONE, WATCH DOG .....	17
JUMPERS .....	18
INTERRUPTS .....	19
INTERFACCIAMENTO CON IL CAMPO .....	20
SELEZIONE COMUNICAZIONE SERIALE .....	20
SELEZIONE MODO OPERATIVO .....	22
INTERFACCIA CAN .....	22
SCHEDE DI SUPPORTO .....	24
UTILIZZO CON MODULO GMB HR84 .....	24
UTILIZZO CON SCHEDA GMM TST E GMM TST3 .....	26

COME INIZIARE .....	28
A) PREDISPOSIZIONE COLLEGAMENTI .....	28
B) PROVA PROGRAMMA DEMO SALVATO NEL MINI MODULO .....	29
C) RIPROGRAMMAZIONE DELLA FLASH CON PROGRAMMA DEMO .....	29
D) CREAZIONE DEL CODICE ESEGUIBILE DEL PROGRAMMA DEMO .....	33
D3A) RICOMPILAZIONE CON BASCOM 8051 .....	33
D3B) RICOMPILAZIONE CON $\mu$ C/51 .....	35
D3C) RICOMPILAZIONE CON LADDER WORK.....	38
E) PREPARAZIONE DEFINITIVA DELL'APPLICAZIONE.....	40
AMBIENTI DI SVILUPPO .....	42
DESCRIZIONE SOFTWARE PERIFERICHE DI BORDO .....	43
INGRESSI DI CONFIGURAZIONE .....	43
LED DI SEGNALAZIONE .....	43
SEGNALI MULTIFUNZIONE .....	44
ACCESSO ALLE MEMORIE .....	44
BIBLIOGRAFIA .....	45
APPENDICE A: DESCRIZIONE COMPONENTI DI BORDO .....	A-1
AT89C51CC03 .....	A-1
APPENDICE B: SCHEMA ELETTRICO GMM TST3.....	B-1
APPENDICE C: CONFIG. BASE, OPZIONI, ACCESSORI .....	C-1
APPENDICE D: INDICE ANALITICO .....	D-1

# INDICE DELLE FIGURE

<b>FIGURA 1: POSIZIONE DELLA VERSIONE SCHEDA .....</b>	<b>3</b>
<b>FIGURA 2: SCHEMA A BLOCCHI .....</b>	<b>7</b>
<b>FIGURA 3: VISTA COMPLESSIVA .....</b>	<b>9</b>
<b>FIGURA 4: PIANTA COMPONENTI (LATO SUPERIORE) .....</b>	<b>11</b>
<b>FIGURA 5: PIANTA COMPONENTI (LATO INFERIORE) .....</b>	<b>11</b>
<b>FIGURA 6: VISTA DALL'ALTO .....</b>	<b>13</b>
<b>FIGURA 7: TABELLA DELLE SEGNALAZIONI VISIVE .....</b>	<b>14</b>
<b>FIGURA 8: DISPOSIZIONE LED, CONNETTORI, JUMPERS .....</b>	<b>15</b>
<b>FIGURA 9: CN1 - ZOCCOLO CON SEGNALI DEL MINI MODULO .....</b>	<b>16</b>
<b>FIGURA 10: TABELLA UTILIZZO JUMPERS .....</b>	<b>18</b>
<b>FIGURA 11: DISPOSIZIONE DEI JUMPERS .....</b>	<b>19</b>
<b>FIGURA 12: ESEMPIO COLLEGAMENTO SERIALE TTL .....</b>	<b>21</b>
<b>FIGURA 13: ESEMPIO COLLEGAMENTO SERIALE RS 232 .....</b>	<b>21</b>
<b>FIGURA 14: GMB HR84 .....</b>	<b>24</b>
<b>FIGURA 15A: GMB HR84 APERTA .....</b>	<b>25</b>
<b>FIGURA 15B: ACCOPPIATA GMB HR84 + GMM 518 ZERO APERTA .....</b>	<b>25</b>
<b>FIGURA 16: ACCOPPIATA GMM TST + GMM 518 ZERO .....</b>	<b>26</b>
<b>FIGURA 17: COLLEGAMENTO SERIALE RS 232 TRA GMM ACB ZERO E PC .....</b>	<b>28</b>
<b>FIGURA 18: CD ESEMPI .....</b>	<b>30</b>
<b>FIGURA 19: FINESTRA SETTAGGIO FLIP (1 DI 3) .....</b>	<b>31</b>
<b>FIGURA 20: FINESTRA SETTAGGIO FLIP (2 DI 3) .....</b>	<b>31</b>
<b>FIGURA 21: FINESTRA SETTAGGIO FLIP (3 DI 3) .....</b>	<b>32</b>
<b>FIGURA 22: CARICAMENTO SORGENTE CON BASCOM 8051 .....</b>	<b>34</b>
<b>FIGURA 23: CONFIGURAZIONE COMPILATORE BASCOM 8051 .....</b>	<b>34</b>
<b>FIGURA 24: COMPILAZIONE CON BASCOM 8051 .....</b>	<b>35</b>
<b>FIGURA 25: CARICAMENTO AMBIENTE CON <math>\mu</math>C/51 E JFE .....</b>	<b>36</b>
<b>FIGURA 26: COMPILAZIONE CON <math>\mu</math>C/51 E JFE .....</b>	<b>36</b>
<b>FIGURA 27: CARICAMENTO SORGENTE CON <math>\mu</math>C/51 ED UEDIT .....</b>	<b>37</b>
<b>FIGURA 28: CARICAMENTO PROGETTO CON <math>\mu</math>C/51 ED UMShell .....</b>	<b>37</b>
<b>FIGURA 29: COMPILAZIONE CON <math>\mu</math>C/51 ED UMShell .....</b>	<b>38</b>
<b>FIGURA 30: CARICAMENTO SCHEMATICO CON LADDER WORK .....</b>	<b>38</b>
<b>FIGURA 31: CONFIGURAZIONE COMPILATORE CON LADDER WORK .....</b>	<b>39</b>
<b>FIGURA 32: COMPILAZIONE CON LADDER WORK .....</b>	<b>39</b>
<b>FIGURA 33: RISULTATO COMPILAZIONE CON LADDER WORK .....</b>	<b>39</b>
<b>FIGURA 34: ESEMPI DI CONNESSIONE .....</b>	<b>41</b>
<b>FIGURA C1: TABELLA CONFIGURAZIONE BASE .....</b>	<b>C-1</b>





## 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 **98-37/CE**.



I pin della scheda non sono dotati di protezione contro le cariche elettrostatiche. Visto che esiste un collegamento diretto tra numerosi pin della scheda ed i rispettivi pin dei componenti di bordo e che quest'ultimi sono sensibili ai fenomeni ESD, il personale che maneggia la scheda è invitato a prendere tutte le precauzioni necessarie per evitare i possibili danni che potrebbero derivare dalle cariche elettrostatiche.

Scopo di questo manuale é 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 518 Zero** con versione stampato **250411** e successive. La validità delle informazioni riportate è quindi subordinata al numero di versione della scheda in uso e l'utente deve quindi sempre verificarne la giusta corrispondenza. Sulla scheda il numero di versione è riportato in più punti sia a livello di serigrafia che di stampato e la seguente figura illustra la posizione più facilmente accessibile.

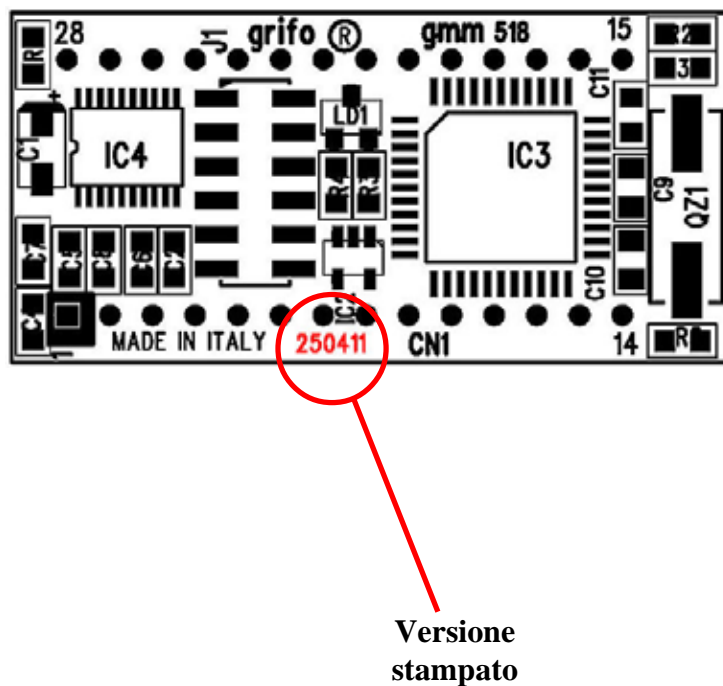


FIGURA 1: POSIZIONE DELLA VERSIONE SCHEDA

## NOTA SUL NOME DEL MINI MODULO

Si prega di notare che accanto al numero di revisione dello stampato, appena descritto, il nome della scheda che è **GMM 518 Zero**. La **GMM 518 Zero** viene infatti realizzata utilizzando lo stampato della **GMM 518** su cui viene montato il microcontrollore **AT89C51CC03**.

Quindi per distinguere una **GMM 518** da una **GMM 518 Zero** bisogna controllare il componente montato su **IC3** e poi usare la seguente corrispondenza:

<i>Microcontrollore Montato</i>	->	<i>Nome del Mini Modulo</i>
Atmel AT89C51CC03	->	<b>GMM 518 Zero</b>
Atmel AT89C51AC2	->	<b>GMM 518</b>

## INFORMAZIONI GENERALI

Il modulo **GMM 518 Zero** é un **Mini Modulo** basato sul microcontrollore **Atmel 89C51CC03**, ovvero un potente e completo sistema dotato di **CPU**; memorie integrate ed una ricca serie di periferiche rivolte alle tipiche richieste dell'automazione.

Il modulo ha già montati, nella sua ridottissima area, i componenti che servono a valorizzare le principali caratteristiche del microcontrollore ed a renderne utilizzabili tutte le modalità operative; inoltre dispone di ulteriori componenti che facilitano ed ampliano i possibili campi di utilizzo.

**GMM 518 Zero** può essere usato per numerose applicazioni sia nel settore **Industriale, Domestico, Automobilistico e Didattico**, come illustrato nei seguenti paragrafi.

La facilità di impiego di questo **Mini Modulo** é determinata anche dalla ricca serie di ambienti di sviluppo software basati su linguaggi sia a basso che alto livello che consentono di poter lavorare al meglio utilizzando unicamente un normale **PC**. Tra questi tools si ricordano i vari Compilatori **C, BASIC, PASCAL** ed anche un ambiente di programmazione grafica a contatti. Grande attenzione é stata riservata alla messa a punto dell'applicativo, rendendo disponibili dei prodotti che salvano direttamente il programma utente nella **FLASH** di bordo, con l'ausilio di un normale PC, e che consentono di provare lo stesso programma direttamente sulla scheda.

La **GMM 518 Zero** é dotata di un connettore normalizzato, che le consente di essere montata immediatamente su schede di supporto come la **GMM TST3**, la **GMB HR84** e la **GAB H844** oppure di essere utilizzata su una scheda dell'utente, come macro componente. In entrambi i casi si riduce drasticamente il 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 518 Zero** sono:

- Zoccolo maschio **28** piedini dual in line a passo 100 mils, largo 600 mils.
- **Ridottissimo ingombro**: 20,6 x 42,6 x 15 mm.
- Circuito stampato a **4 strati** per ottimizzare le immunità e le caratteristiche **EMI**.
- Unica tensione di alimentazione a **+5 Vdc, 10 mA** (l'assorbimento varia in base ai collegamenti ed allo stato del modulo).
- Disponibilità di modalità operative a basso consumo come **Idle Mode e Power Down Mode**.
- Microcontrollore **Atmel AT89C51CC03** (codice compatibile **8051**) con quarzo da **14,74 Mhz**.
- **Ciclo Macchina** programmabile a **12 o 6** periodi di clock.
- **64K FLASH** per codice, **2K FLASH** per boot loader, **256 bytes IRAM** per dati, **2K bytes ERAM** per dati, **2K EEPROM** per dati.
- **8** canali di **A/D** converter con **10 bit** di risoluzione, **20 µsec** per ogni conversione.
- **Tensione di Riferimento** della sezione **A/D** riportata sul connettore.
- **19** sorgenti di **Interrupt** con **4** livelli di priorità.
- **3 Timer Counter** a **16** bits.
- **5** canali **PCA** a **16** bist con funzionalità autonome di **PWM**, comparazione, cattura su evento, generatore di frequenze, ecc.
- Sistema di **Watchdog** che conferisce al sistema controllato una **Sicurezza** estrema in tutte le condizioni operative.
- **24** linee di **I/O** digitale collegate al connettore. Alcune di queste hanno funzionalità multiple.
- Circuiteria di **Reset** e controllo alimentazione.
- Linea seriale hardware con **Baud Rate** programmabile fino a **115.200 Baud**, bufferata

in **TTL** o in **RS 232**.

- Driver per linea seriale **RS 232** con protezione contro scariche fino a  $\pm 15$  kV.
- Linea **I2C BUS** software, riportata sul connettore.
- Controllore **CAN** compatibile con standard **2.0A** e **2.0B**, che tramite un adeguato driver esterno, consente il collegamento a tutte le linee basate su questo standard.
- **Jumpers** di configurazione ad **5** vie di cui **1** acquisibili via software.
- **1 LED** di segnalazione gestiti via Firmware tramite **I/O** digitali.
- Possibilità di gestione della **FLASH** ed **EEPROM** di bordo in modalità **In System Programming**, ovvero con modulo già montato, sfruttando la linea di comunicazione seriale.
- **Software Gratuito** per **PC**, di supporto alla programmazione **ISP**, che permette di scaricare il codice generato nella **FLASH** di bordo.
- Vasta disponibilità di software di sviluppo che consentono di poter utilizzare il modulo tramite un semplice **PC**. Tra i pacchetti disponibili si possono ricordare: vari **Assemblatori** (ASM51, A51, SXA51); Compilatori **C** ( $\mu$ C/51, SYS51CW); Compilatori **BASIC** (BASCOM 8051); Compilatori **PASCAL** (SYS51PW); linguaggi con Logica a Contatti (LadderWORK); ecc.
- Ricca serie di programmi dimostrativi ed esempi di utilizzo forniti sotto forma di sorgenti ampiamente commentati ed eseguibili, per i vari ambienti di sviluppo.

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

## COMUNICAZIONE SERIALE

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

La linea seriale è collegata al connettore **CN1** a livello **TTL** o **RS 232**, grazie alla configurazione di alcuni Jumper di bordo. Quando la scheda deve essere collegata in una rete, collegata a notevole 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 disponibile 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 sulla scelta del protocollo elettrico leggere il paragrafo SELEZIONE COMUNICAZIONE SERIALE e contattare direttamente la **grifo®** se necessario; mentre per dettagli sulla programmazione dei registri con cui impostare il protocollo fisico e gestire la comunicazione fare riferimento alla documentazione del microcontrollore all'APPENDICE A di questo manuale ed ai numerosi esempi forniti.

## LINEE DI I/O DIGITALE TTL

Il **Mini Modulo GMM 518 Zero** mette a disposizione **24** linee di I/O digitale **TTL** del microcontrollore **Atmel AT89C51CC03** raggruppate in quattro port da **8** bit (P0, P1, P2, P3) ed un port da 2 bit (P4). Per convenzione tutti i segnali dei Port sono designati con i nomi P0.0÷7, P1.0÷7, P2.0÷7, P3.0÷7 e P4.0÷1.

Tali linee sono collegate direttamente al connettore a 40 vie con una disposizione standard comune a tutti i **Mini Moduli grifo®** ed hanno quindi la possibilità di essere direttamente collegate a numerose schede d'interfaccia.

Via software è definibile ed acquisibile la funzionalità, la direzionalità e lo stato di queste linee, con possibilità di associarle anche alle periferiche della scheda (**Timer, Counter, Interrupt, A/D, PCA, CAN**, ecc.), tramite una semplice programmazione di alcuni registri interni del microcontrollore. Per maggiori informazioni fare riferimento ai paragrafi **CONNESSIONI** e **DESCRIZIONE SOFTWARE PERIFERICHE DI BORDO**.

## LINEA I2C BUS

Sul connettore della **GMM 518 Zero** sono disponibili i segnali di un'interfaccia **I2C BUS**, emulata via **Firmware**, e gestita da due linee di I/O del microcontrollore.

Grazie a questa interfaccia sincrona possono essere collegati dispositivi dotati dello stesso standard di comunicazione in modo da espandere localmente le potenzialità del 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, controllori di display, memorie, sensori di temperatura, ecc.

A tale proposito può essere utile esaminare la **K51-AVR** di cui è disponibile sia il manuale tecnico con il relativo schema elettrico che una completa raccolta di esempi in vari linguaggi. In aggiunta si ricordano anche alcuni modelli dei pannelli operatore **QTP** che possono essere comandati lasciando libera l'alternativa linea di comunicazione seriale asincrona.

## LINEE ANALOGICHE DI A/D CONVERTER

Il **Mini Modulo GMM 518 Zero** mette a disposizione **8** linee di ingresso analogiche collegate alla sezione **A/D** converter del microcontrollore. A queste linee possono essere collegati sensori esterni per numerose grandezze fisiche come temperatura, pressione, velocità, peso, ecc. o qualsiasi sistema che fornisca un segnale in tensione compatibile.

Le caratteristiche principali di questa sezione sono: risoluzione di **10** bit; il range d'ingresso massimo **0÷3 V**; tempo di conversione tipico su singolo canale di **20 µsec**; semplicissima gestione software; generazione interrupt di fine conversione

Le conversioni **A/D** sono eseguite con la tecnica delle approssimazioni successive e si effettuano tramite un'opportuna manipolazione di appositi registri interni.

Al fine di semplificare la gestione del convertitore **A/D**, alcuni pacchetti software forniscono delle procedure di utility che gestiscono la sezione in tutte le sue parti.

Per ulteriori informazioni si vedano i documenti nell'**APPENDICE A** ed il paragrafo **CONNESSIONI**.

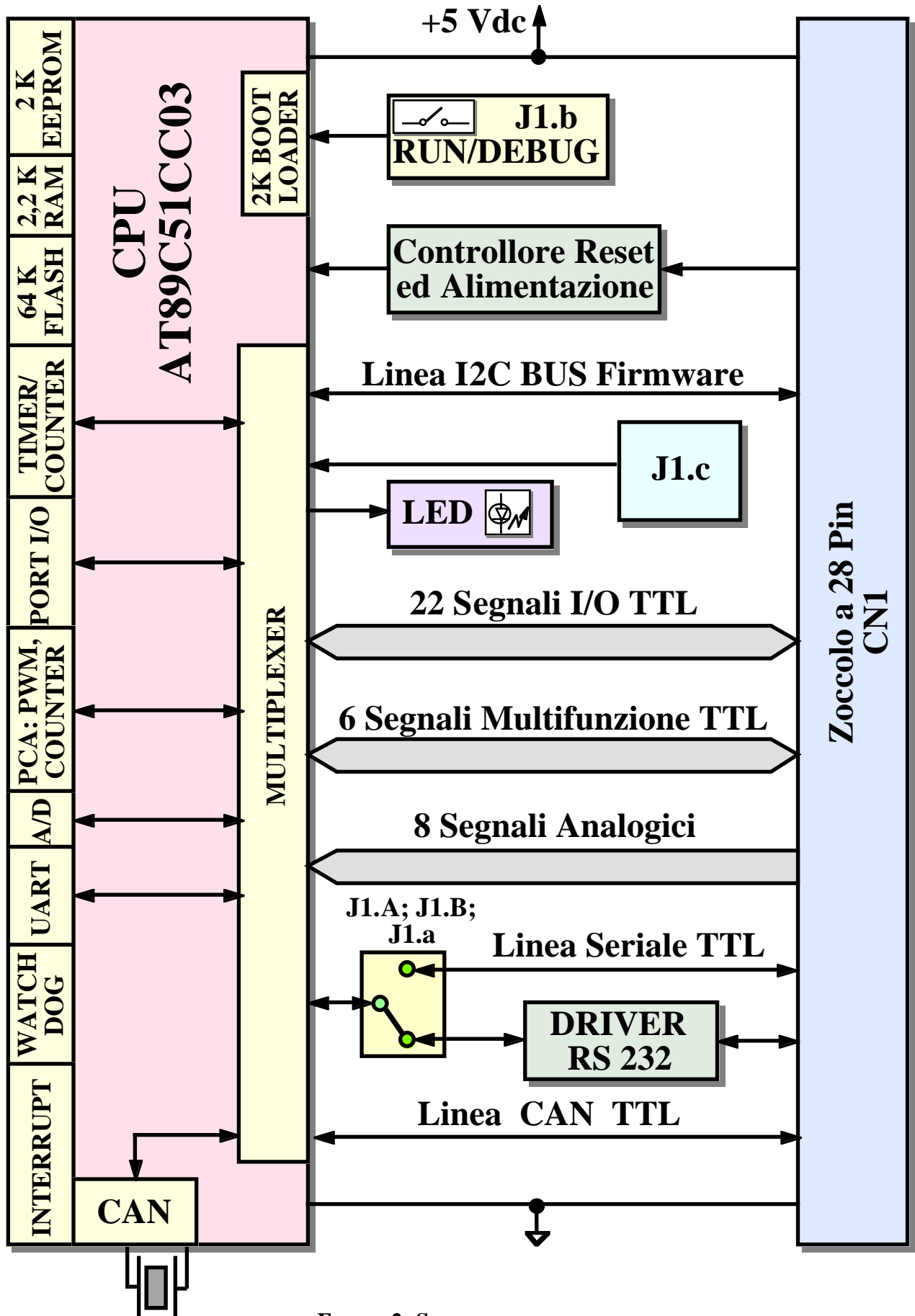


FIGURA 2: SCHEMA A BLOCCHI

## MEMORIE

La scheda è dotata di un massimo di **70,25K** di memoria variamente suddivisi in:

- **64K** Bytes FLASH EPROM;
- **2K** Bytes FLASH EPROM per il Boot Loader;
- **256** Bytes di IRAM interna;
- **2K** Bytes di ERAM esterna;
- **2K** Bytes di EEPROM.

Tutte le memorie, ad eccezione della **FLASH** per il **Boot Loader**, sono a completa disposizione dell'utente sia per il codice che per i dati del suo programma applicativo.

Grazie alla **EEPROM** del microcontrollore 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 le quantità di memorie risultino insufficienti (ad esempio per sistemi di data login o applicazioni molto articolate) si possono sempre collegare dei dispositivi di memoria esterni nelle tecnologie **SRAM**, **EEPROM**, **EPROM** e **FLASH** tramite l'interfaccia **I2C BUS** descritta prima.

## WATCH DOG

Il microcontrollore **AT89C51CC03** 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. Lo scopo principale di questa sezione è quella di conferire al sistema controllato dal **Mini Modulo** una sicurezza intrinseca che ne garantisce il lavoro in qualsiasi condizione operativa e che evita pericolosi malfunzionamenti non previsti durante lo sviluppo.

Il tempo di intervento della sezione è programmabile in un range piuttosto ampio, da un minimo di circa **6** millisecondi ad **1,7** secondi.

Per ulteriori informazioni si veda la documentazione del microcontrollore o l'APPENDICE A di questo manuale.

## CLOCK

Sul modulo **GMM 518 Zero** è presente una circuiteria basata su un quarzo, che si occupa della generazione del segnale di clock per il microcontrollore. Questa circuiteria genera una frequenza di **14,7456 MHz** che viene quindi usata direttamente o indirettamente da tutte le periferiche del modulo che richiedono temporizzazioni, ovvero la linea seriale asincrona, i **Timer**, il **Watch Dog**, i canali **PCA**, il controllore **CAN**, ecc.

Dal punto di vista delle prestazioni si ricorda che la **GMM 518 Zero** ha la possibilità di impostare la durata di un ciclo macchina a **12** o **6** cicli di clock (**X2 mode**) e che se paragonata ad una **CPU I51** classica, esegue il codice 2 volte più velocemente.



## CONFIGURAZIONE SCHEDA

Allo scopo di rendere configurabile la scheda ed in particolare il programma applicativo sviluppato, è stato previsto un Jumper ad 5 vie. L'acquisizione via software dello stato di 1 di questi Jumper, fornisce all'utente la possibilità di gestire diverse condizioni tramite un unico programma, senza dover rinunciare ad altre linee d'ingresso (le applicazioni caratteristiche sono: selezione della lingua di rappresentazione, definizione di parametri del programma, selezione delle modalità operative, ecc).

In aggiunta la scheda ha un **LED**, gestibili via software, che possono essere usati per segnalare visivamente la configurazione attuale della scheda, come descritto negli appositi paragrafi. Tutte le risorse di configurazione descritte sono completamente gestite via software, tramite la programmazione di appositi registri interni del microcontrollore. Per ulteriori informazioni si vedano i paragrafi DIP SWITCH, SEGNALAZIONI VISIVE e LEDS DI SEGNALAZIONE.

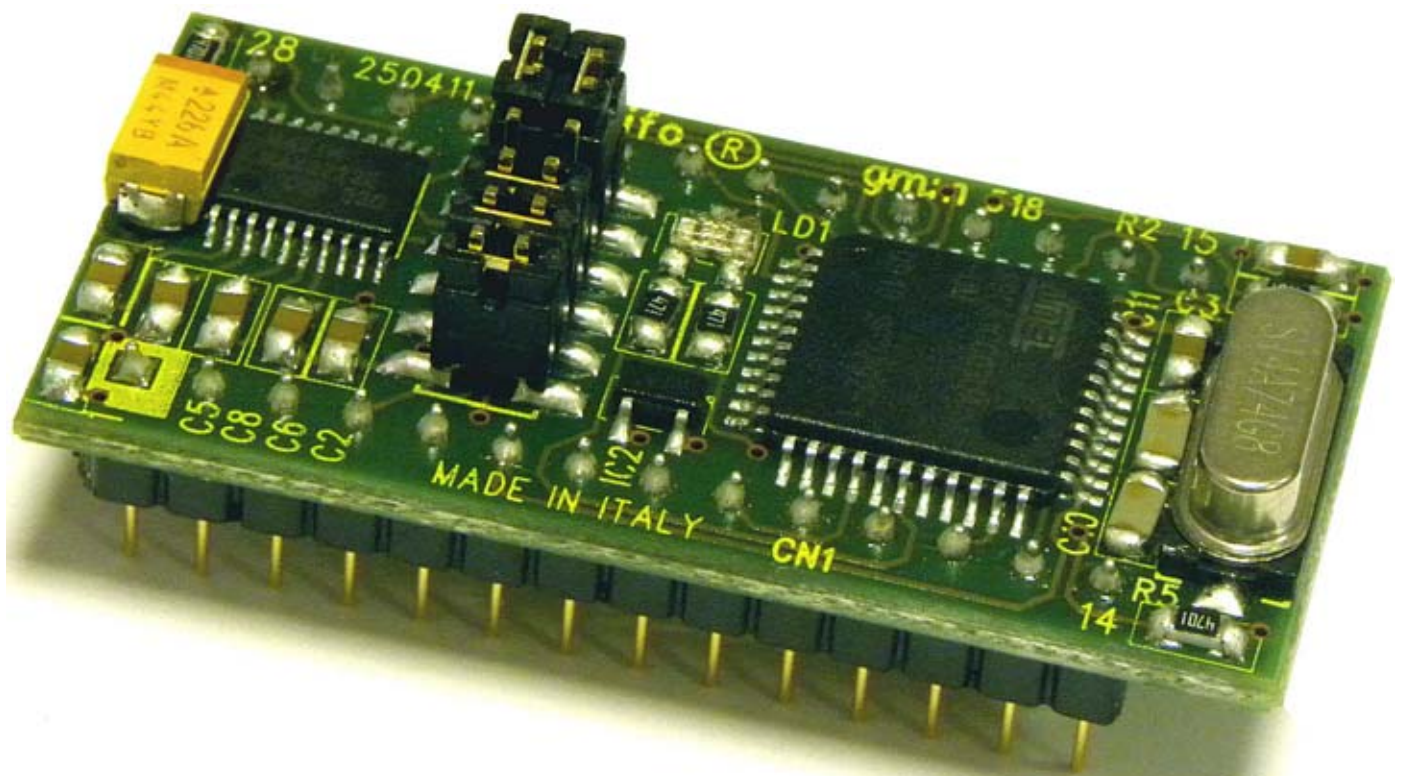


FIGURA 3: VISTA COMPLESSIVA

## CONTROLLORE CAN

Il controllore **CAN** della **GMM 518 Zero** si preoccupa della completa gestione dell'omonimo protocollo in tutte le sue modalità ed aspetti. In dettaglio si possono brevemente elencare le caratteristiche di questa sezione:

- supporto protocollo BasicCAN (2.0 A);
- supporto protocollo PeliCAN (2.0 B);
- gestione identificatori da 11 e 29 bits;
- 15 gestori di messaggi indipendenti;
- identificatori, maschere e comandi separati per ogni gestore;
- lunghezza buffer di ricezione programmabile;
- temporizzatore programmabile per sincronismi e trigger;
- contatori di errori leggibili;
- genera diversi interrupt a seconda dello stato raggiunto;
- completamente gestibile tramite registri interni del microcontrollore;
- ecc.

Per ulteriori informazioni si veda il paragrafo INTERFACCIA CAN e l'APPENDICE A di questo manuale.

## SETTORI DI UTILIZZO

Le possibili applicazioni dei moduli **GMM 518 Zero** 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, ecc.
- L'uso in **Sistemi a Logica Distribuita** tipo robot, automazioni su macchine di produzione in linea, automazioni di fabbriche di grosse dimensioni.
- La **Teleacquisizione** ed il **Telecontrollo** su medio brevi distanze.
- L'**Automazione Domestica** (accensione e spegnimento luci, controllo riscaldamento e condizionamento, supervisione elettrodomestici e servizi elettrici, sistemi di sorveglianza e controllo accesso).
- Il **Settore Automobilistico** (accensione e spegnimento luci, controllo temperature, supervisione servizi per conducente, sistemi antifurto, diagnostica di funzionamento).
- Anche le **Applicazioni CAN** sono possibili, a patto che il Mini Modulo sia integrato con un driver di linea esterno. In dettaglio si ricordano nuovamente le applicazioni nel settore automobilistico; il collegamento su reti **CAN** con protocolli proprietari o con protocolli standard come **CANopen**; **DeviceNet**; **SDS**, **CAN Kingdom**; **J1939**, ecc..
- Da non dimenticare il settore **didattico** dove la **GMM 518 Zero** offre la possibilità di apprendere il funzionamento di un microcontrollore della famiglia **8051** e di sviluppare le sue applicazioni canoniche ad un costo veramente basso. A questo scopo è ideale abbinarla alle schede di supporto **GMM TST3**, che risolvono i problemi dell'alimentazione, del collegamento seriale al **PC** di sviluppo, del collegamento delle linee del modulo e che allo stesso tempo dispongono di una tastiera a matrice ed un display **LCD** che consentono di studiare e provare soluzioni a basso costo per l'interfacciamento operatore.
- Tutti i casi in cui si vogliono ridurre i tempi e costi di sviluppo, infatti il modulo può essere utilizzato direttamente su una scheda di supporto dell'utente, come un **Macro Componente**.

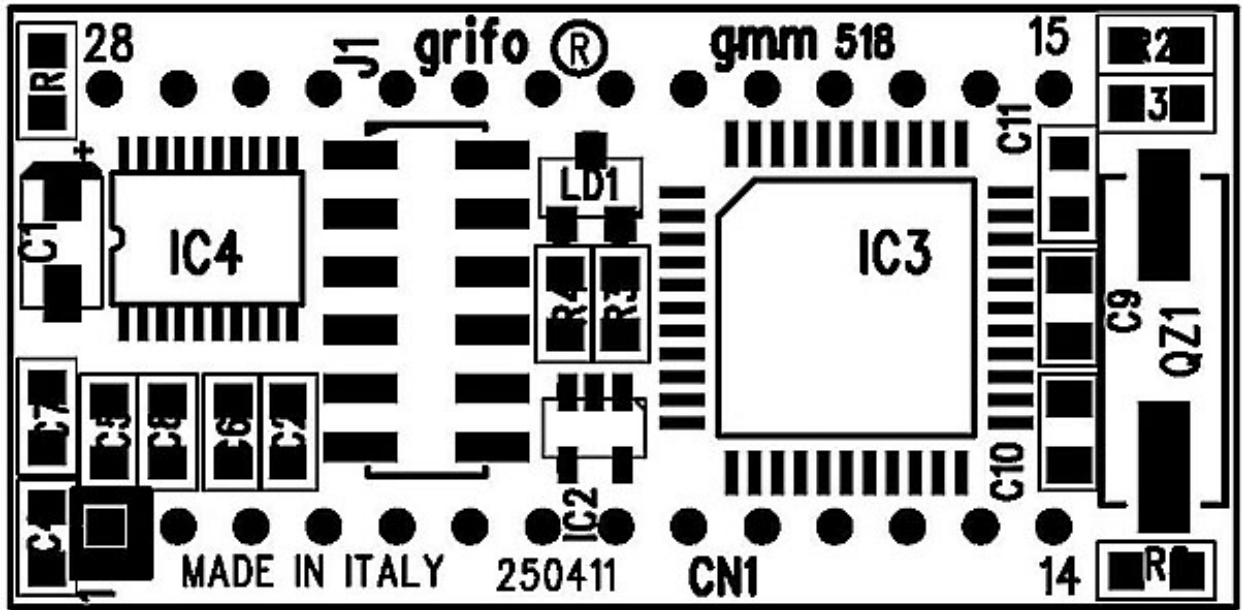


FIGURA 4: PIANTA COMPONENTI (LATO SUPERIORE)

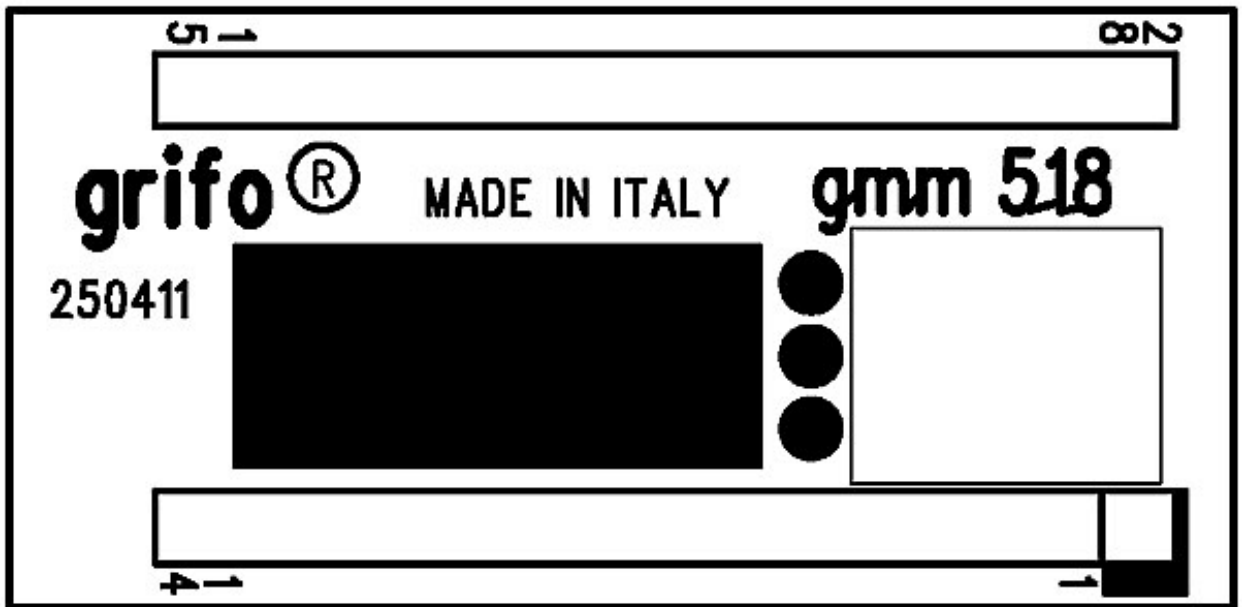


FIGURA 5: PIANTA COMPONENTI (LATO INFERIORE)

## CARATTERISTICHE TECNICHE

### CARATTERISTICHE GENERALI

**Risorse della scheda:**

- 24 linee di I/O digitale TTL
- 8 ingressi analogici su A/D converter
- 5 sezioni PCA per comparazione, cattura, PWM, frequenze
- 1 sezione Watch Dog
- 3 Timer, Counter multifunzione
- 19 sorgenti e 4 livelli di interrupt
- 1 circuiteria di reset e controllo alimentazione
- 1 linea seriale RS 232 o TTL
- 1 controllore CAN
- 5 Jumper di configurazione
- 1 LEDs di segnalazione

**Memorie:**

64K Bytes FLASH	programma utente
2K Bytes FLASH	boot loader
2K Bytes EEPROM	dati utente (ultimi 8 Bytes riservati)
2K Bytes ERAM	dati utente
256 Bytes IRAM	dati utente

**Microcontrollore:** Atmel AT89C51CC03

**Frequenza clock:** 14,7465 MHz

**Risoluzione A/D:** 10 bit

**Tempo conversione A/D:** 20  $\mu$ sec

**Tempo reset:** tipico 200 msec

**Tempo intervento Watch Dog:** programmabile da circa 6 msec ad 1,7 sec

### CARATTERISTICHE FISICHE

**Dimensioni (L x A x P):** 20,61 x 42,6 x 15 mm

**Peso:** 9 g

**Connettori:** CN1: zoccolo maschio da 28 piedini, passo 100 mils, largo 600 mils

**Range di temperatura:** da 0 a 50 gradi Centigradi

**Umidità relativa:** 20% fino a 90% (senza condensa)

## CARATTERISTICHE ELETTRICHE

<b>Tensione di alimentazione:</b>	+5 Vdc $\pm$ 5%
<b>Consumo di corrente:</b>	10 mA * (normale) 14 mA * (massimo)
<b>Range ingressi analogici:</b>	0÷Vref
<b>Tensione riferimento A/D Vref:</b>	0÷3 V
<b>Impedenza ingressi analogici:</b>	elevata
<b>Soglia controllore alimentazione:</b>	tipica 4,65 Vdc
<b>Protezione RS 232:</b>	$\pm$ 15 kV

\* I dati riportati sono riferiti ad un lavoro a temperatura ambiente di **20** gradi centigradi (per ulteriori informazioni fare riferimento al paragrafo ALIMENTAZIONE).

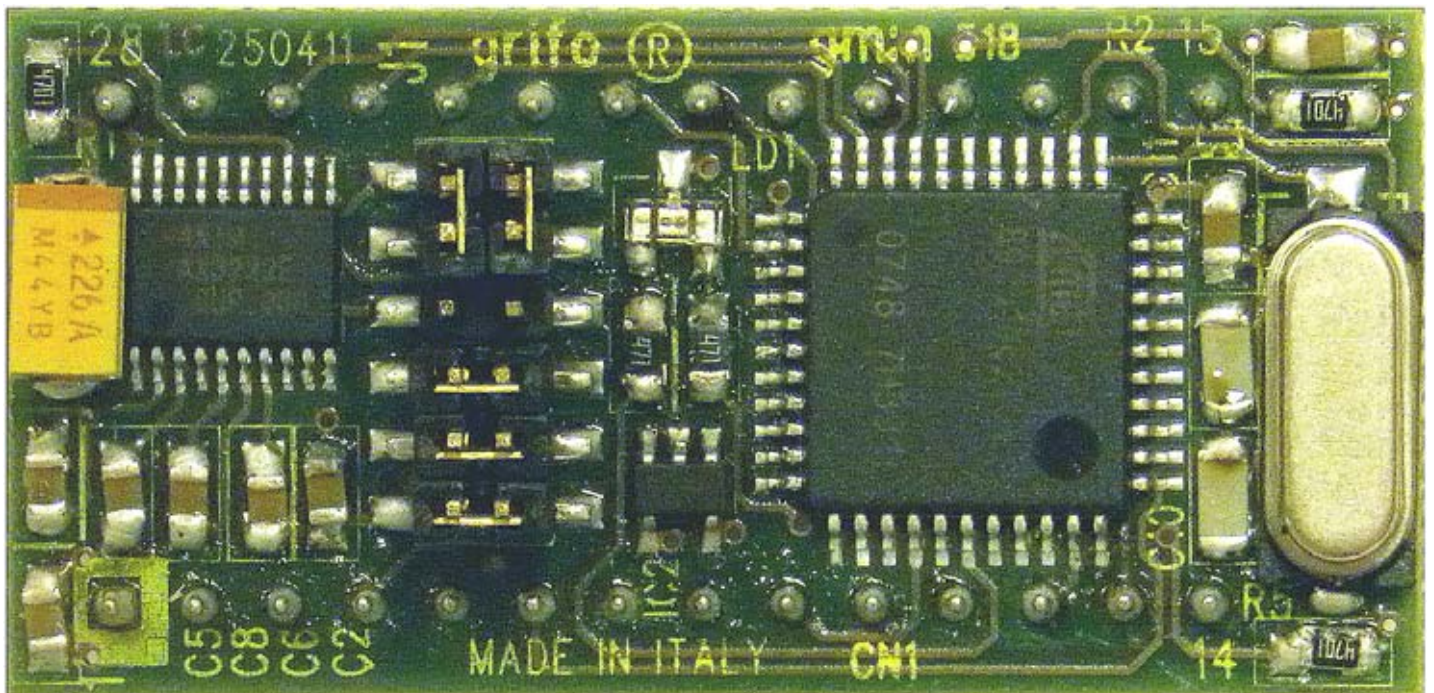


FIGURA 6: VISTA DALL'ALTO

## 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 **LED**, Jumpers, ecc. presenti sulla **GMM 518 Zero**.

### SEGNALAZIONI VISIVE

Il Mini Modulo **GMM 518 Zero** è dotato delle segnalazioni visive descritte nella seguente tabella:

LED	COLORE	SIGNIFICATO
LD1	Rosso	Se è Acceso, indica che la linea P2.7 (pin 16 della CPU) è a livello Logico Basso ( <b>L</b> ). Se è Spento, indica che la linea P2.7 (pin 16 della CPU) è a livello Logico Alto ( <b>H</b> ).

**FIGURA 7: TABELLA DELLE SEGNALAZIONI VISIVE**

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

### ALIMENTAZIONE

Il **Mini Modulo** deve essere alimentato mediante una tensione di **+5 Vdc ± 5%** da fornire sugli appositi piedini **14 e 28** di **CN1**.

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 diverse modalità a basso assorbimento che ad esempio salvaguardano la durata di batterie, nel caso di applicazioni portatili. In dettaglio si possono utilizzare le modalità operative di power down ed idle che consentono di definire la frequenza di lavoro del microcontrollore e sono selezionate programmando l'apposito registro interno **PCON**. Il programma applicativo sviluppato dall'utente può quindi ridurre il consumo sull'alimentazione ed eventualmente ripristinare il funzionamento normale in corrispondenza di un evento presatibilito come ad esempio un interrupt, variazione di un ingresso digitale e/o analogico, intervallo di tempo trascorso, ecc.

Per ulteriori informazioni si faccia riferimento al paragrafo **CARATTERISTICHE ELETTRICHE**.

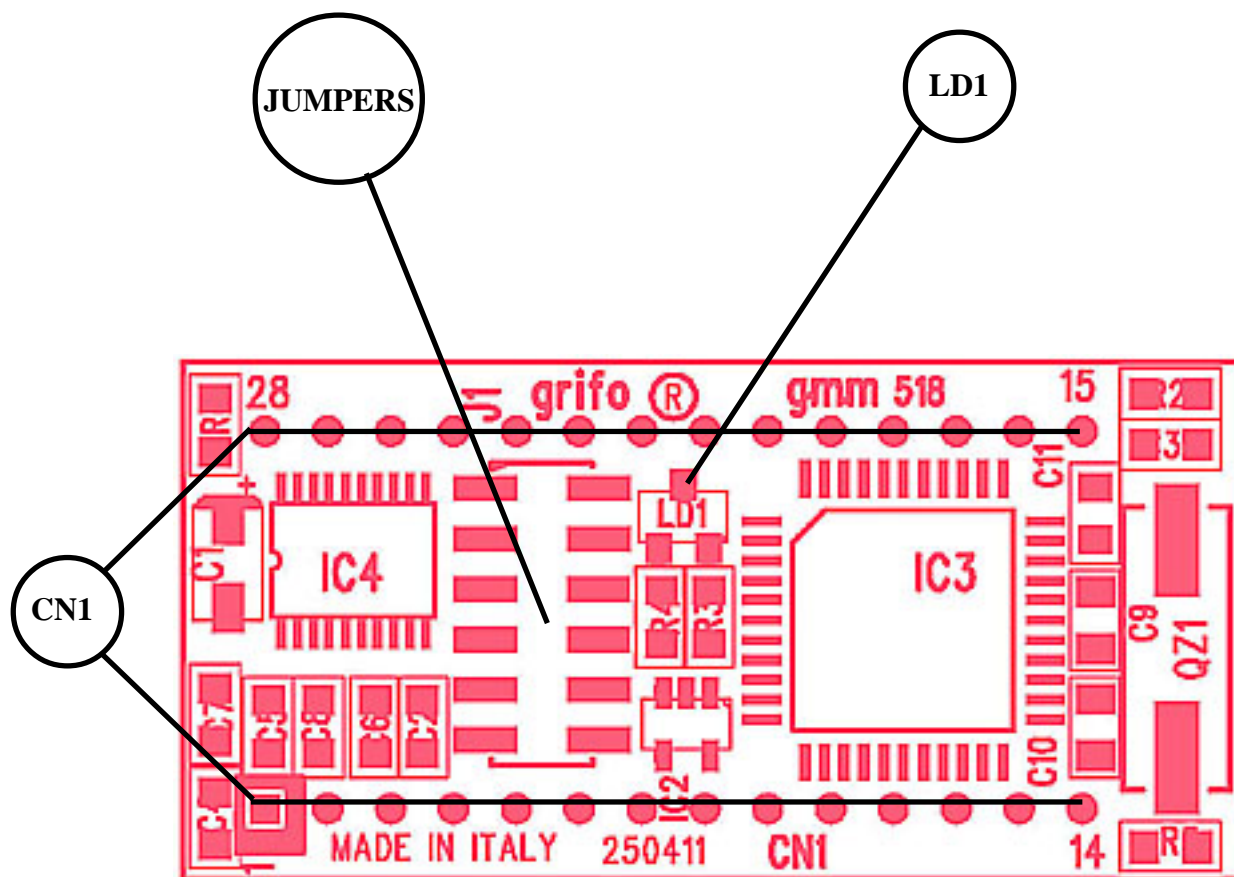


FIGURA 8: DISPOSIZIONE LEDs, CONNETTORI, DIP SWITCH

## CONNESSIONI

Il **Mini Modulo GMM 518 Zero** 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 riportata la disposizione 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.

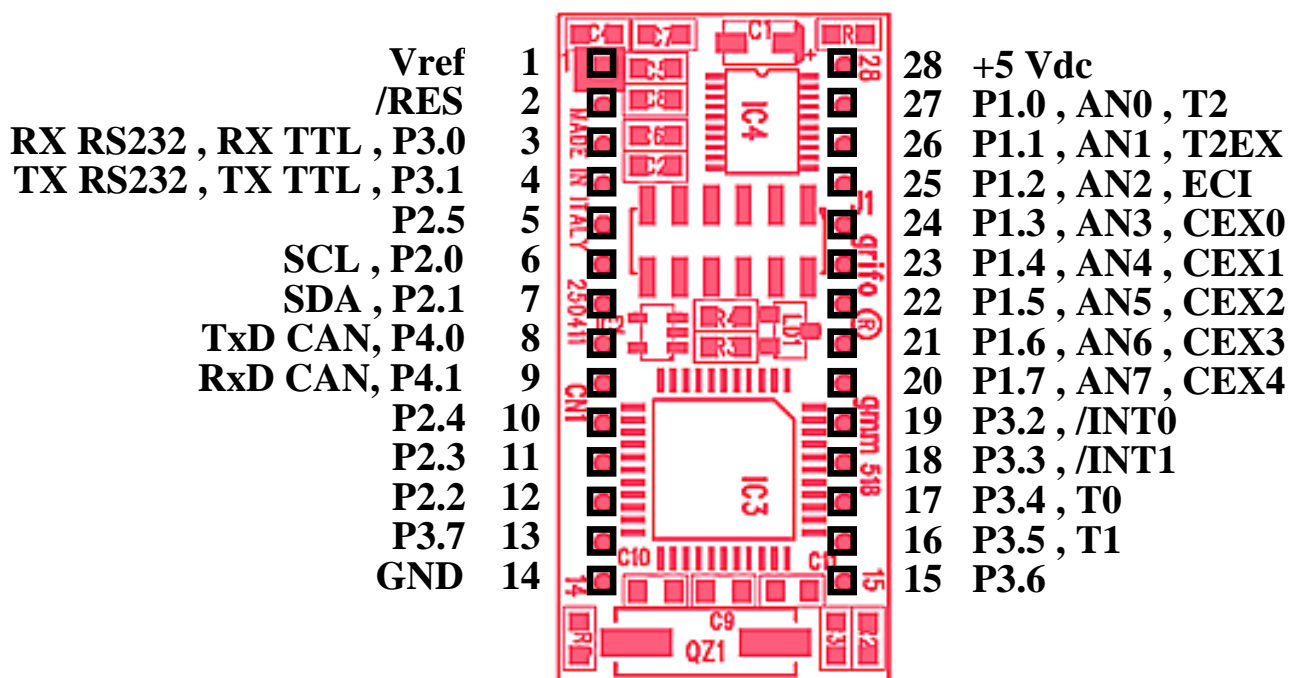
Tutti i connettori delle schede **grifo®** rispettano una disposizione segnali standard in modo da poter facilmente sostituire una scheda anche con un modello diverso, senza tempi e costi aggiuntivi.

## CN1 - CONNETTORE CON SEGNALI DEL MINI MODULO

Il connettore **CN1** è 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, i segnali di espansione, ecc.

Numerosi piedini di questo connettore hanno una duplice o triplice funzione infatti, via software, alcune sezioni interne del microcontrollore 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 CON IL CAMPO** e seguono il pin out standardizzato dei **Mini Moduli grifo®**.

Al fine di evitare problemi di conteggio e numerazione la figura **9** descrive i segnali direttamente sulla vista dall'alto della **GMM 518 Zero**, inoltre la serigrafia riporta la numerazione sui **4** angoli della scheda sia sul lato superiore che inferiore.



**FIGURA 9: CN1 - ZOCCHOLO CON SEGNALI DEL MINI MODULO**

Legenda:

<b>RX RS232</b>	= I - Receive Data: linea di ricezione seriale bufferata in RS 232
<b>TX RS232</b>	= O - Transmit Data: linea di trasmissione seriale bufferata in RS 232
<b>RX TTL</b>	= I - Receive Data: linea di ricezione seriale TTL
<b>TX TTL</b>	= O - Transmit Data: linea di trasmissione seriale TTL
<b>Px.n</b>	= I/O - Segnale n del Port x di I/O digitale del microcontrollore
<b>RxD CAN</b>	= I - Receive Data CAN: linea di ricezione seriale TTL
<b>TxD CAN</b>	= O - Transmit Data CAN: linea di trasmissione seriale TTL
<b>/INTn</b>	= I - Linea d'interrupt n del microcontrollore
<b>Tn</b>	= I - Segnale collegato alla sezione Timern del microcontrollore
<b>T2EX</b>	= I - Segnale di trigger esterno per Timer 2



<b>SCL</b>	= I/O - Linea clock dell'interfaccia I2C BUS software
<b>SDA</b>	= I/O - Linea dati dell'interfaccia I2C BUS software
<b>RxD CAN</b>	= I - Linea di ricezione controllore CAN
<b>TxD CAN</b>	= O - Linea di trasmissione controllore CAN.
<b>Vref</b>	= I - Tensione di riferimento dell'A/D converter
<b>ANn</b>	= I - Ingressi analogici collegati a sezione A/D converter
<b>CEXn</b>	= I/O - Segnale collegato alla sezione PCAn (uscite PWM, ingressi cattura e comparazione, ecc.)
<b>ECI</b>	= I - Ingresso di clock esterno delle sezioni PCA
<b>/RES</b>	= I - Segnale di reset
<b>+5 Vdc</b>	= I - Linea di alimentazione +5 Vdc
<b>GND</b>	= - Linea di massa

## RESET, CONTROLLO ALIMENTAZIONE, WATCH DOG

Sulla **GMM 518 Zero**, sono presenti tre diverse sorgenti di reset, che possono essere così riassunte:

- 1) Circuiteria di controllo alimentazione che attiva il reset quando la tensione di alimentazione scende al di sotto della soglia di **4,65 Vdc**.
- 2) Segnale collegato al pin **8** di **CN1**, attivo basso, che può essere collegato anche ad un semplice pulsante normalmente aperto e che, una volta premuto, collega il segnale **/RES** alla massa **GND**. La sua funzione principale é quella di uscire da condizioni di loop infinito, soprattutto durante la fase di debug oppure per rieseguire il programma applicativo senza dover interrompere l'alimentazione della scheda.
- 3) Circuiteria di **Watch Dog**, interna al microcontrollore, molto efficiente e di facile gestione software. In dettaglio le caratteristiche di questa circuiteria sono le seguenti:
  - funzionamento astabile;
  - tempo d'intervento programmabile via software da circa **6 msec** fino a circa **1.700 msec**;
  - attivazione via software tramite doppia scrittura consecutiva su registri di gestione;
  - retrigger via software.

Nel funzionamento astabile una volta scaduto il tempo d'intervento, la circuiteria si attiva, rimane attiva per il tempo di reset e poi si disattiva nuovamente. Si ricorda che la funzione principale della circuiteria di **Watch Dog** é quella di conferire una sicurezza intrinseca per la corretta esecuzione del programma da parte della scheda. Infatti se il programma non é più eseguito regolarmente, non effettua il retrigger periodico della circuiteria e questa quindi interviene resettando la scheda, come descritto. Per maggiori informazioni sulla circuiteria di **Watch Dog** e sull'operazione di retrigger, fare riferimento alla documentazione del microcontrollore oppure all'APPENDICE A di questo manuale.

Per le prime due sorgenti, la circuiteria di reset rimane attiva per **200 msec** e poi si disattiva, mentre per la terza rimane attiva solo per pochi  $\mu$ sec. A questo punto tutte le sezioni della scheda vengono resettate per garantire una condizione di azzeramento generale la scheda riprende l'esecuzione del programma salvato in **FLASH** all'indirizzo **0000H**.

La circuiteria di reset così realizzata assicura il corretto funzionamento della scheda e dell'eventuale elettronica collegata, in ogni condizione operativa e soprattutto nella sempre difficile fase di accensione e spegnimento.

## JUMPERS

JUMPER	POSIZIONE	UTILIZZO	DEF.
J1.A	1-2	<p><b>Connessione in TTL</b></p> <p>- Collega il segnale di ricezione della linea seriale asincrona, sul pin 3 di CN1, direttamente al microcontrollore. Usato in abbinamento ai jumpers J1.B e J1.a.</p>	*
	2-3	<p><b>Connessione in RS232</b></p> <p>- Collega il segnale di Ricezione (<b>RX</b>) della linea seriale asincrona, sul pin 3 di CN1, al driver RS 232. Usato in abbinamento ai jumpers J1.B e J1.a.</p>	
J1.B	1-2	<p><b>Connessione in TTL</b></p> <p>- Collega il segnale di trasmissione della linea seriale asincrona, sul pin 4 di CN1, direttamente al microcontrollore. Usato in abbinamento ai jumpers J1.A e J1.a.</p>	*
	2-3	<p><b>Connessione in RS232</b></p> <p>- Collega il segnale di Trasmissione (<b>TX</b>) della linea seriale asincrona, sul pin 4 di CN1, al driver RS 232. Usato in abbinamento ai jumpers J1.A e J1.a.</p>	
J1.a	Non Connesso	<p><b>Connessione in TTL</b></p> <p>- Non collega segnale di ricezione della linea seriale asincrona del microcontrollore al driver RS 232. Usato in abbinamento ai jumpers J1.A e J1.B.</p>	*
	Connesso	<p><b>Connessione in RS232</b></p> <p>- Collega segnale di ricezione della linea seriale asincrona del microcontrollore al driver RS 232. Usato in abbinamento ai jumpers J1.A e J1.B.</p>	
J1.b	Non Connesso	<p><b>Non Collega</b> - Non collega la linea /PSEN del microcontrollore. Se il <b>Mini Modulo</b> viene acceso o resettato in questa condizione si attiva la modalità <b>RUN</b> e parte il programma memorizzato nella <b>FLASH</b> del microcontrollore.</p>	*
	Connesso	<p><b>Collega</b> - Collega la linea /PSEN del microcontrollore alla massa (zero volt). Se il <b>Mini Modulo</b> viene acceso o resettato in questa condizione si attiva la modalità <b>DEBUG</b> e parte il <b>Boot Loader</b>.</p>	
J1.c	Non Connesso	<p><b>Non Collega</b> - Lascia libero il pin 24 della <b>CPU</b> che va quindi al +5Vdc. Questo condizione pone a livello logico <b>1 (H)</b> il segnale P0.0 del microcontrollore, agendo come ingresso di configurazione acquisibile via software.</p>	*
	Connesso	<p><b>Collega</b> - Collega il pin 24 della <b>CPU</b> a massa. Questo condizione pone a livello logico <b>0 (L)</b> il segnale P0.0 del microcontrollore, agendo come ingresso di configurazione acquisibile via software.</p>	

FIGURA 10: TABELLA DI UTILIZZO DEI JUMPERS

A bordo del **Mini Modulo GMM 518 Zero** é presente un Jumpers da **5 vie**, denominato **J1.x**, con cui è possibile effettuare alcune selezioni che riguardano il modo di funzionamento dello stesso. Nelle seguenti figure ne è riportato l'elenco e la loro funzione nelle varie modalità di connessione.

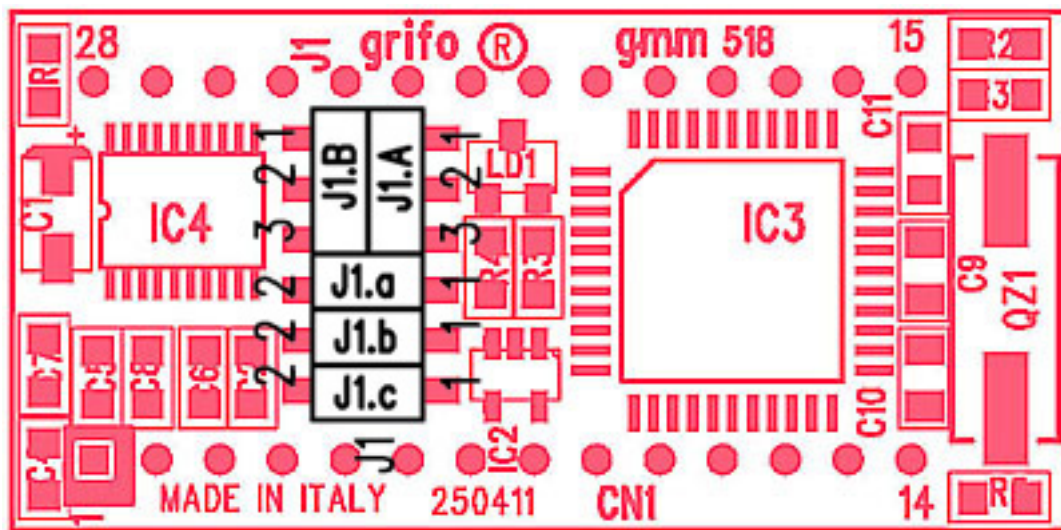


FIGURA 11: DISPOSIZIONE DEI JUMPERS

Nelle precedenti tabelle l'\* (asterisco) indica la configurazione di default, ovvero quella impostata in fase di collaudo, con cui la scheda viene fornita. L'utente può verificare la configurazione base ricevuta, anche nell'APPENDICE C del manuale.

Per individuare facilmente la posizione del Jumpers si faccia riferimento alla figura 11.

## INTERRUPTS

Una caratteristica peculiare della **GMM 518 Zero** 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 alla documentazione del microprocessore oppure all'APPENDICE A di questo manuale.

- Pin 19 di CN1 -> Genera un /INT0 = P3.2 sul microcontrollore.
- Pin 18 di CN1 -> Genera un /INT1 = P3.3 sul microcontrollore.
- Periferiche di bordo -> Generano un interrupt interno. In particolare le possibili sorgenti d'interrupt interno sono le sezioni: Timer 0, Timer 1, Timer 2, PCA, UART seriale, A/D converter, controllore CAN, temporizzatore CAN, ecc.

Complessivamente il modulo dispone di **19** sorgenti d'interrupt e di **4** livelli di priorità ed incorporata nel microcontrollore si trova la logica di gestione degli interrupt che consente di attivare, disattivare, mascherare le sorgenti 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.

## INTERFACCIAMENTO CON IL CAMPO

Al fine di evitare eventuali problemi di collegamento della scheda con tutta l'elettronica del campo a cui la **GMM 518 Zero** si deve interfacciare, si devono seguire le informazioni riportate in tutto il manuale e le relative figure che illustrano le modalità interne di connessione.

- Per i segnali che riguardano la comunicazione seriale con il protocollo **RS 232** fare riferimento alle specifiche standard di questo protocollo.
- 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 **5V** 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 che preferibilmente devono essere optoisolate in modo da mantenere la logica del **Mini Modulo** separata dagli eventuali disturbi dell'elettronica di potenza.
- I segnali d'ingresso alla sezione **A/D** devono essere collegati a segnali analogici a bassa impedenza che rispettino il range di variazione ammesso ovvero da **0** a **3 Vdc**. Queste indicazioni valgono anche per la tensione di riferimento  $V_{ref}$  che inoltre deve essere perfettamente filtrata e stabilizzata.
- I segnali **PWM** generati dalle sezioni **Timer**, **Counter** e **PCA**, 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** software 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 e configurare opportunamente lo stadio d'uscita, le molteplici modalità operative ed il bit rate in modo da poter comunicare in ogni condizione operativa. In ogni collegamento **I2C BUS** ci devono essere due resistenze di pull up alle estremità della stessa, come definito dalle specifiche standard di questa interfaccia.
- I segnali di comunicazione **CAN** sono a livello **TTL** e non possono essere collegati direttamente ai segnali delle linee **CAN**, bensì si deve interporre un adeguato driver di linea, come indicato nel paragrafo INTERFACCIA CAN.

## SELEZIONE COMUNICAZIONE SERIALE

La linea di comunicazione seriale asincrona della scheda **GMM 518 Zero** può essere bufferata in **RS 232** o **TTL**. Nel caso venga bufferata in **RS 232**, i segnali della linea sono protetti da scariche fino a  $\pm 15$  KV.

Dal punto di vista software sulla linea può essere definito il protocollo fisico di comunicazione tramite la programmazione di alcuni registri interni del microprocessore.

La selezione del protocollo elettrico avviene via hardware e richiede un'opportuna configurazione del Jumper 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 TTL

J1.A = 1 - 2  
 J1.B = 1 - 2  
 J1.a = Non Connesso

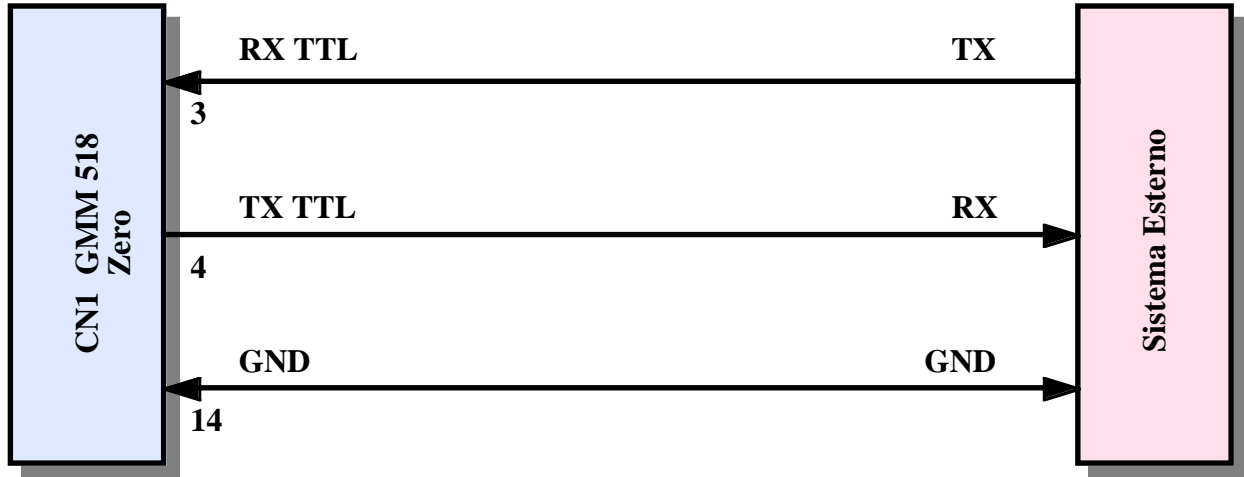


FIGURA 12: ESEMPIO COLLEGAMENTO SERIALE TTL

- LINEA SERIALE SETTATA IN RS 232 (configurazione default)

J1.A = 2 - 3  
 J1.B = 2 - 2  
 J1.a = Connesso

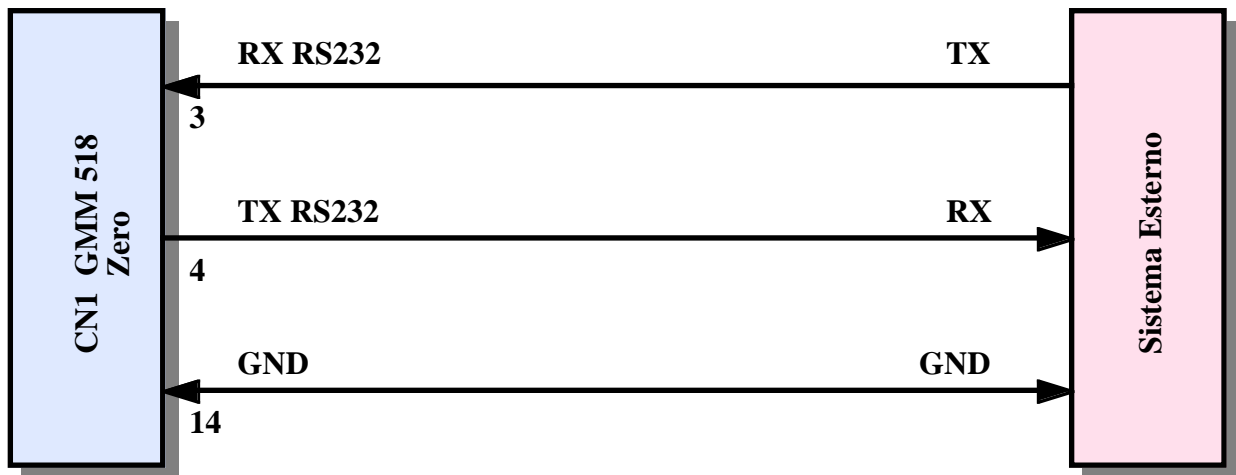


FIGURA 13: ESEMPIO COLLEGAMENTO SERIALE RS 232

## SELEZIONE MODO OPERATIVO

Come descritto nella figura **10** e nei successivi paragrafi il Jumper **J1.b** seleziona il modo operativo del **Mini Modulo GMM 518 Zero**. In particolare sono disponibili **2** modi operativi corrispondenti alle seguenti configurazioni:

<b>J1.b</b>		<b>Modo operativo</b>
Non Collegato	->	Modalità <b>RUN</b>
Collegato	->	Modalità <b>DEBUG</b>

In modalità **RUN** a seguito di un'accensione o reset parte sempre il programma applicativo salvato in **FLASH** indipendentemente dalle condizioni esterne, mentre in modalità **DEBUG** parte il **Boot Loader** del microcontrollore.

Programmi per **PC** come il **FLIP** comunicano con il **Boot Loader** attraverso la porta seriale e permettono di leggere, cancellare e riscrivere le memorie di bordo del microcontrollore in modalità **ISP**.

La programmazione **ISP** (*In System Programming*) riduce i costi ed i tempi di sviluppo dell'applicazione, infatti elimina la necessità di usare **EPROM** esterne, programmatori, cancellatori, ecc. Per ulteriori informazioni sulla programmazione **ISP** si prega di consultare il capitolo **COME INIZIARE** oppure la specifica documentazione tecnica rilasciata dalla **ATMEL**.

## INTERFACCIA CAN

Dal punto di vista elettrico il **CAN BUS**, secondo lo standard **ISO-11898**, coincide con una linea differenziale con impedenza di **60 Ω**. Tale linea é notevolmente diversa dalle due linee di comunicazione **CAN** presenti su **CN1** che infatti sono a livello **TTL**. Per questa ragione il collegamento della **GMM 518 Zero** ad una qualsiasi interfaccia **CAN** deve essere effettuato interponendo un apposito driver di linea **CAN**.

Per informazioni dettagliate su tale driver si consiglia di leggere la documentazione tecnica del componente scelto, che spesso riporta anche schemi applicativi (si veda ad esempio il componente **PHILIPHS P82C250**).

Realizzando un corretto collegamento del driver sia sui segnali **TTL** provenienti dal controllore del **Mini Modulo** che su quelli differenziali del **BUS CAN** si può arrivare alla velocità di **1 Mbit/sec** rendendo possibile la comunicazione con qualsiasi sistema dotato della stessa interfaccia. Ad esempio si suggerisce di prevedere lungo la linea **CAN** due resistenze di terminazione (**120 Ω**), alle estremità della stessa, che assicurano la giusta impedenza; qualora i sistemi collegati sulla rete **CAN** risultino a differenze di potenziale elevate si può ovviare ad eventuali problemi di comunicazione e/o funzionamento, collegando anche le masse dei sistemi.

La **grifo**<sup>®</sup> dispone di una ricca serie di schede dotate di interfaccia **CAN** pronte all'uso, anche nel formato **Mini Modulo**: l'utente prima di aggiungere il driver alla **GMM 518 Zero** può esaminare l'utilizzo di questi prodotti.

## SCHEDE DI SUPPORTO

Il **Mini Modulo GMM 518 Zero** 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 gli abbinamenti con le schede di supporto più interessanti.

In corrispondenza del primo ordine viene fornita una ricca serie di programmi dimostrativi che consentono di utilizzare immediatamente sia le risorse del **Mini Modulo**, che quelle della scheda di supporto.

### UTILIZZO CON MODULO GMB HR84

La **GMB HR84** si distingue per essere una scheda che fornisce ai **Mini Moduli** da **28** pin di essere collegati ai segnali del campo tramite delle apposite circuiterie di potenza optoisolate, in modo da essere facilmente e velocemente installati nei quadri elettrici del sistema da realizzare

La descrizione completa del prodotto è disponibile nel relativo foglio e manuale tecnico, mentre in questo paragrafo sono riportate le potenzialità offerte e la configurazione richiesta da questa accoppiata.

La **GMB HR84** permette facilmente di:

- alimentare il **Mini Modulo** tramite lo switching di bordo che accentra un ingresso ad ampio range sia in continua che alternata;
- riportare **8** linee dei port di **I/O** sugli ingressi optoisolati che possono essere indifferentemente di tipo **NPN** o **PNP**. Lo stato di tutti gli **8** ingressi viene visualizzato tramite **LED** verdi. Essendo le linee multiplexate con le periferiche interne è possibile creare rapidamente funzioni evolute come contatori, riconoscimento combinazioni, gestore di eventi; ecc.;
- fornire una alimentazione galvanicamente isolata per gli ingressi **NPN** o **PNP**, in modo da poter collegare all'esterno dei semplici ed economici contatti puliti;
- riportare **4** linee dei port di **I/O** su uscite a Relè visualizzate tramite **LED** rossi;
- avere l'interfaccia **I2C BUS**, completa di alimentazione, su un connettore dedicato;
- collegare la linea di comunicazione seriale tramite un connettore **AMP MODU II** da **8** vie;
- bufferare, esternamente al **Mini Modulo**, i segnali della linea seriale **TTL** in **RS 422**, **RS 485** o **Current Loop** passivo;
- collegare i segnali **PWM** su un connettore **AMP MODU II** da **8** vie;
- fornire la tensione di riferimento per la sezione **A/D** e collegare un ingresso analogico;
- cablare facilmente tutti i segnali tramite connettori a morsettiera con rapida estrazione ed altri connettori standard;
- effettuare un montaggio meccanico su barra ad **Omega**.



**FIGURA 14: GMB HR84 - MODULO PERIFERICO DA CAMPO ADATTO ALLA GMM 518 ZERO**

La seguente configurazione consente di usare l'accoppiata **GMB HR84 + GMM 518 Zero** in modalità **RUN**, con linea seriale in **RS 232** ed ingressi optoisolati:

*Configurazione GMM 518 Zero*

J1.A = 2 - 3  
 J1.B = 2 - 3  
 J1.a = Connesso  
 J1.b = Non Connesso  
 J1.c = Indifferente

*Configurazione GMB HR84*

J1 , J9 = Non connessi  
 J2, J3, J4 = 2-3  
 J5 , = 2-3  
 J6 = 1-2  
 J7 = 1-2  
 J8 = Non connesso  
 J10 = 2-3  
 J11 = Non connesso  
 J13, J14 = 1-2

Cavo collegamento seriale con PC di sviluppo = **AMP8 Cable** con vaschetta **D9** Femmina.



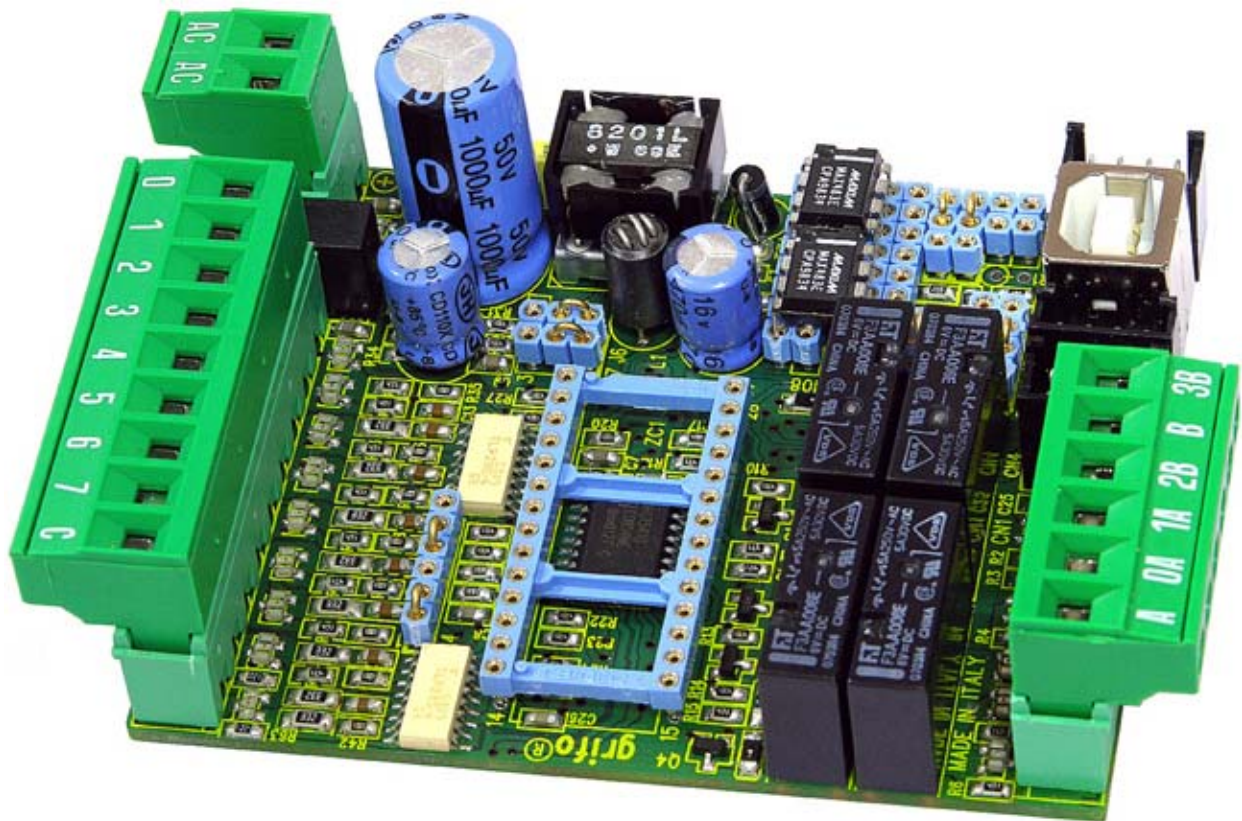


FIGURA 15A: VISTA DELLA GMB HR84 APERTA E SENZA ALCUN MINI MODULO.

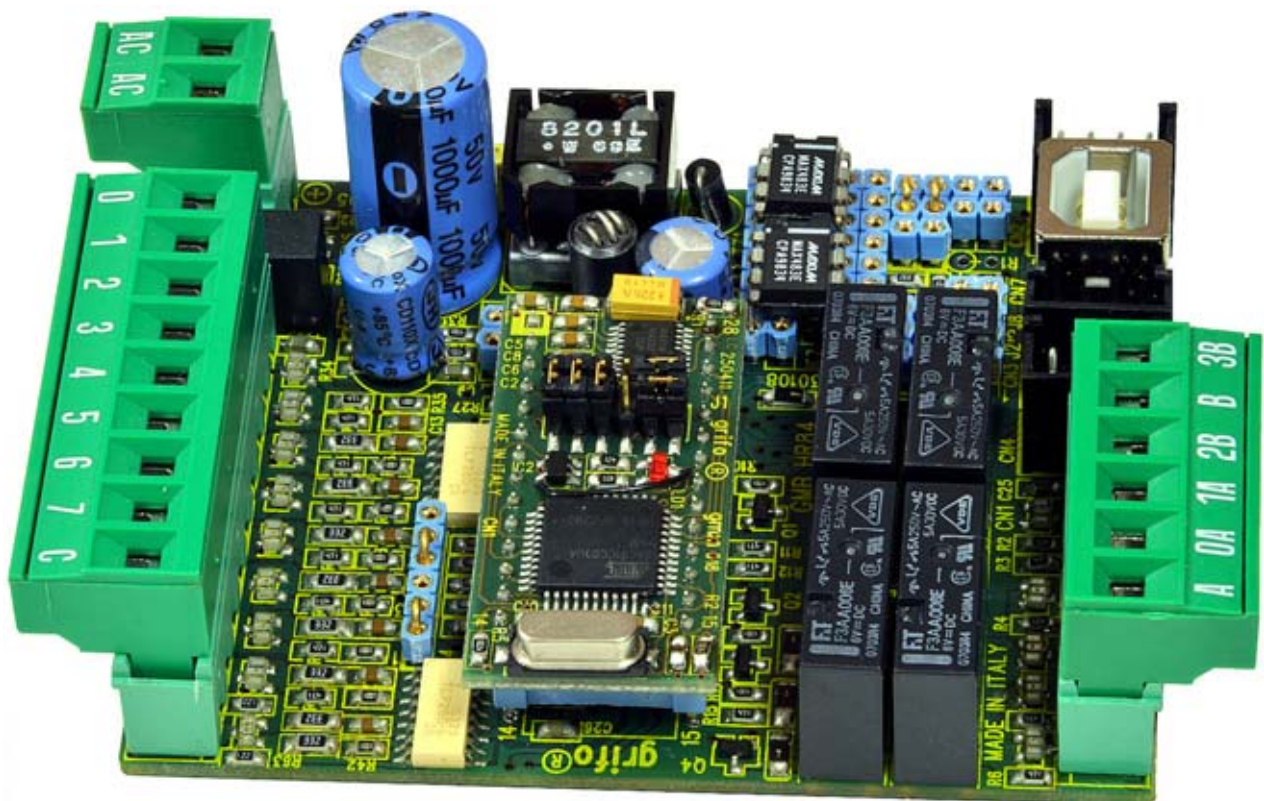


FIGURA 15B: VISTA DELLA GMB HR84 APERTA CON MONYATO UNA GMM 518 ZERO

## UTILIZZO CON SCHEDA GMM TST3

Nel carteggio **grifo®** le **GMM TST3** si distinguono per essere le schede valutative progettate esplicitamente per fare da supporto ai **Mini Moduli GMM xxx** da 28 e 40 pins.

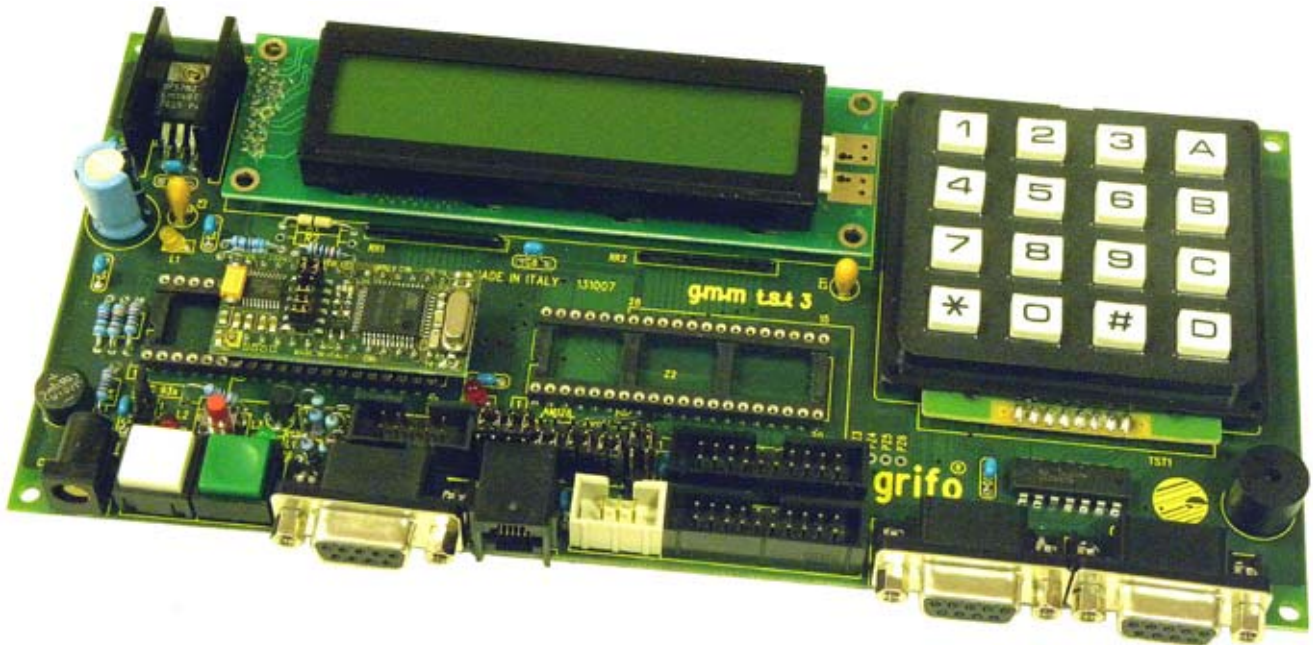


FIGURA 16: ACCOPPIATA GMM TST3 + GMM 518 ZERO

La **GMM TST3** é un'evoluzione della **GMM TST**, ma per il **Mini Modulo GMM 518** possono essere usate tutte indifferentemente. Nel presente manuale si descrive la **GMM TST3** ma l'utente può facilmente usare gli altri modelli seguendo le indicazioni del relativo manuale tecnico. Nell'appendice **B** viene riportato anche lo schema elettrico completo di questa scheda di supporto. La **GMM TST3** permette di:

- alimentare il **Mini Modulo** tramite l'alimentatore lineare di bordo che accetta un ingresso ad ampio range sia in continua che alternata;
- riportare le linee dei port di **I/O** e dell'**A/D** converter su connettori a scatolino compatibili con lo standard **I/O ABACO®** su cui sono disponibili schede di pilotaggio **I/O** Digitali di diversi tipi (Relé, Transistor, Ingressi Optoisolati, **LED** pulsanti, ecc.);
- collegare la linea seriale **RS 232** tramite un connettore a vaschetta **D9** femmina;
- impostare e visualizzare lo stato della 2 linee di **I/O**, del **Mini Modulo** e tramite pulsanti e **LED** di colori differenti;
- generare feedback sonori mediante il **Buzzer** di bordo;
- fornire la tensione di riferimento per la sezione **A/D**;
- azzerare il **Mini Modulo** montato tramite un comodo pulsante di **Reset**;

- 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;
- sviluppare una scheda di supporto secondo le esigenze dell'utente, partendo dallo schema elettrico fornito;
- gestire la programmazione ed il **Debugger** del **Mini Modulo** tramite appositi programmatori ed interfacce esterni (se interessati potete contattare la **grifo®**).

La seguente configurazione consente di usare l'accoppiata **GMM TST3** + **GMM 518 Zero** in modalità **RUN**, tramite la linea seriale in **RS 232**:

*Configurazione GMM 518 Zero*

J1.A = 2 - 3  
J1.B = 2 - 3  
J1.a = Connesso  
J1.b = Non Collegato  
J1.c = Indifferente

*Configurazione GMM TST 3*

J1 = 2-3  
J2 = Non connesso  
J3 = 1-2  
J4 = 2-3  
J5 = 2-3  
J6 = 2-3  
J7 = 2-3  
J8 = 2-3  
J9 = 2-3  
J10 = 1-2  
J11 = 1-2  
J12 = 1-2  
J13 = 1-2

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

## COME INIZIARE

In questo capitolo vengono illustrate le operazioni da effettuare per iniziare ad usare la **GMM 518 Zero** in maniera rapida e lineare, senza dover affrontare alcun problema tipico del primo uso.

Il capitolo é composto da alcune parti comuni ed altre che invece si differenziano a seconda dell'ambiente di sviluppo usato.

Inoltre si ipotizza che l'utente disponga degli accessori (di alimentazione e collegamento seriale) e di un **PC** provvisto dei requisiti minimi necessari ad eseguire i programmi descritti nei seguenti punti; tale **PC** viene identificato con il nome **PC** di sviluppo ed i suoi requisiti sono riportati nella documentazione degli stessi programmi usati.

### A) PREDISPOSIZIONE COLLEGAMENTI

A1) Per prima cosa dovete effettuare il collegamento seriale tra il **Mini Modulo GMM 518 Zero** ed il **PC** di sviluppo, ovvero collegare i due segnali di comunicazione (**TX RS232**, **RX RS232**) e la massa di riferimento (**GND**). Naturalmente questa connessione varia a seconda dell'eventuale scheda di supporto usata ed in generale é quella riportata nella seguente figura.

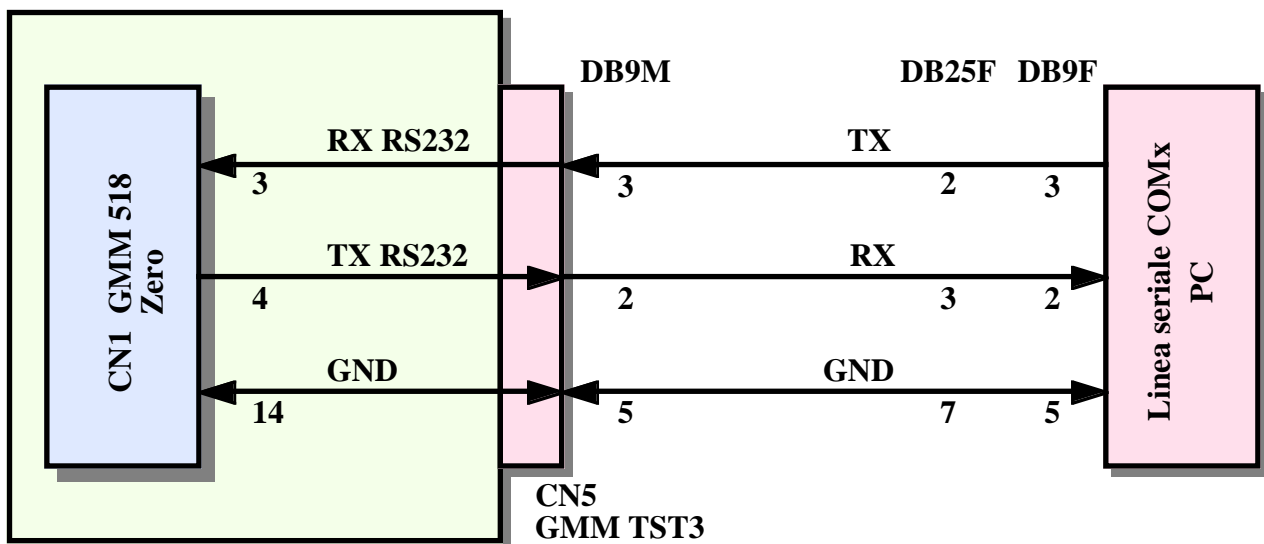


FIGURA 17: COLLEGAMENTO SERIALE RS 232 TRA GMM 518 ZERO E PC

Come si può notare dalla figura 17, in caso di utilizzo dell'accoppiata **GMM TST3 + GMM 518 Zero**, il cavo di collegamento seriale coincide con un normale cavo dritto con vaschetta **D9** (quelli ad esempio usati per la comunicazione con **Modem RS 232**) che può essere anche ordinato alla **grifo®** specificando il codice **CCR 9+9E**.

A2) Provvedere un'alimentazione adeguata: in caso di utilizzo del solo mini modulo tale alimentazione deve essere fornita come descritto nel paragrafo **ALIMENTAZIONE**, ad esempio usando un alimentatore da laboratorio. Qualora si utilizzi una accoppiata si possono sfruttare diverse sorgenti di alimentazione, come descritto nel manuale tecnico della scheda di supporto usata.

## B) PROVA PROGRAMMA DEMO SALVATO NEL MINI MODULO

- B1) Avviare il programma di emulazione terminale **HYPERTERMINAL** sul **PC** di sviluppo ed impostare la comunicazione a:

<i>Connetti:</i>	<i>direttamente a COMx (quella usata al punto A1)</i>
<i>Bit per secondo:</i>	<i>19200</i>
<i>Bit di Dati:</i>	<i>8</i>
<i>Parità:</i>	<i>Nessuna</i>
<i>Bit di Stop:</i>	<i>1</i>
<i>Controllo di flusso:</i>	<i>Nessuno</i>

Si ricorda che **HYPERTERMINAL** é un programma fornito assieme al sistema operativo Windows e può essere facilmente lanciato seguendo il percorso: *Avvio / Programmi / Accessori / Comunicazioni / HyperTerminal*. In alternativa si possono usare anche altri programmi, come l'emulatore terminale incorporato nel compilatore **BASCOM 8051**.

- B2) Impostare la modalità **RUN**, ovvero posizionare il Jumper **J1.b** della **GMM 518 Zero** in posizione Non Connesso (Jumper Aperto).
- B3) Fornire l'alimentazione o resettare il **Mini Modulo** e verificare che il **LED** lampeggi continuamente e che sul monitor del **PC** di sviluppo compaia la presentazione del programma demo. Ogni **GMM 518 Zero**, in caso di primo acquisto, viene fornita con il rispettivo programma demo già programmato nella **FLASH** interna e configurato per farlo partire all'accensione con le funzionalità descritte.  
Se non vedete accadere quanto descritto, riverificate l'alimentazione, la connessione seriale e che il Jumper **J1.b** sia in posizione Non Connesso (Jumper Aperto)
- B4) Eseguire le funzioni del demo in modo da provare le risorse disponibili sul **Mini Modulo** e verificarne gli effetti: l'utente può interagire con il demo usando il **PC** di sviluppo come una console.
- B5) Terminata la prova del demo, togliere alimentazione alla **GMM 518 Zero**.
- B6) Uscire dal programma **HYPERTERMINAL** sul **PC** di sviluppo.

## C) RIPROGRAMMAZIONE DELLA FLASH CON PROGRAMMA DEMO

- C1) Sul **CD grifo®** ricevuto, localizzare e quindi installare sul disco rigido del **PC** di sviluppo il programma di utility **FLIP**. Questo gestisce la programmazione **ISP** della **FLASH** presente sulla **GMM 518 Zero** e comunica con il **Boot Loader** presente sullo stesso modulo.  
In alternativa potete trovare l'ultima versione gratuita di questo programma sul sito **Atmel**: [www.atmel.com](http://www.atmel.com).
- C2) Creare sul disco rigido del **PC** di sviluppo una cartella in cui intendete salvare tutto il lavoro svolto.

- C3) Individuare il programma demo della **GMM 518 Zero** sul **CD grifo®** ricevuto: tale file ha il nome rappresentato alla partenza dello stesso programma, al punto B3, ed è raggiungibile a partire dalla pagina iniziale seguendo il percorso: *Italiano / Esempio Demo Schede / Grifoi Mini Moduli / GMM\_518\_Zero*.
- C4) Copiare tutti i file presenti dalla cartella individuata al punto precedente, alla cartella di lavoro creata al punto C2. A copia ultimata assicurare che i file su disco rigido abbiano l'attributo di *Sola lettura* disattivo.



FIGURA 18: CD CON ESEMPI

- C5) Impostare la modalità **DEBUG**, ovvero posizionare il Jumper **J1.b** della **GMM 518 Zero** in posizione Connesso (Jumper Chiuso).
- C6) Togliere e, successivamente, ridare alimentazione al **Mini Modulo**, oppure resettarlo.

- C7) Lanciare il programma **FLIP** installato al punto C1. Salvo diverse impostazioni effettuate durante la stessa installazione si può usare il collegamento *Avvio / Programmi / ATMEL / FLIP x.y.z / FLIP*.
- C8) Selezionare il dispositivo da programmare, premendo il primo pulsante in alto a sinistra, poi scegliendo il nome *AT89C51CC03* nella finestra che appare e premendo *OK*:

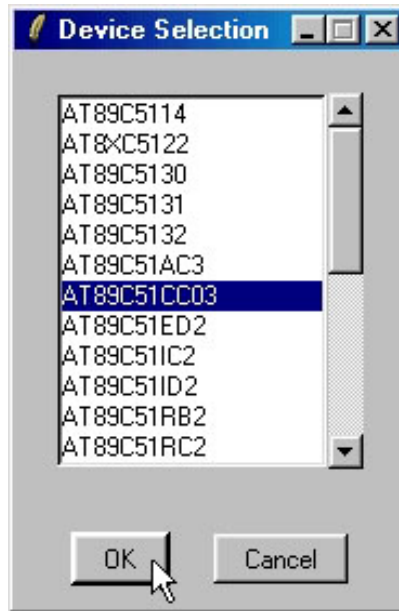


FIGURA 19: FINESTRA SETTAGGIO FLIP (1 DI 3)

- C9) Impostare la modalità di comunicazione per la programmazione **ISP** ovvero premere il secondo pulsante in alto da sinistra, scegliendo in sequenza: *RS 232*, la porta seriale del PC collegata al punto A1, *115.200 Baud* e quindi premendo il pulsante *Connect*. A questo punto il **FLIP** instaura la comunicazione con il **Boot Loader** del microcontrollore e compila una serie di dati nella sua finestra principale.
- Se la comunicazione non si instaura e dopo circa **20** secondi appare la finestra *Timeout Error*, potete abbassare la velocità di comunicazione da 115.200 a 19.200 Baud e ricontrollare i punti precedenti.

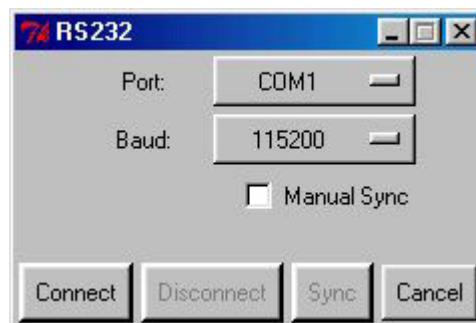


FIGURA 20: FINESTRA SETTAGGIO FLIP (2 DI 3)

- C10) Caricare il file da programmare nella **FLASH** (ovvero quello con estensione **.HEX** copiato al punto C4) premendo il terzo pulsante da destra e selezionando il file tramite la finestra di dialogo rappresentata. Nel riquadro *FLASH Buffer Information* appaiono varie informazioni sull'operazione appena effettuata; in particolare la voce *HEX File:* deve riportare il nome del file.

C11) Spuntare tutte le caselle del riquadro *Operations Flow* come in figura 21, in modo che il **FLIP** esegua sequenzialmente le quattro operazioni di: cancellazione, verifica di cancellazione, programmazione e verifica di programmazione.

C12) A questo punto assicurarsi che la finestra principale del **FLIP** si presenti come in figura 21, in particolare per le voci *Size:BLJB, X2; Device SSB e BSB / EB / SBV*.

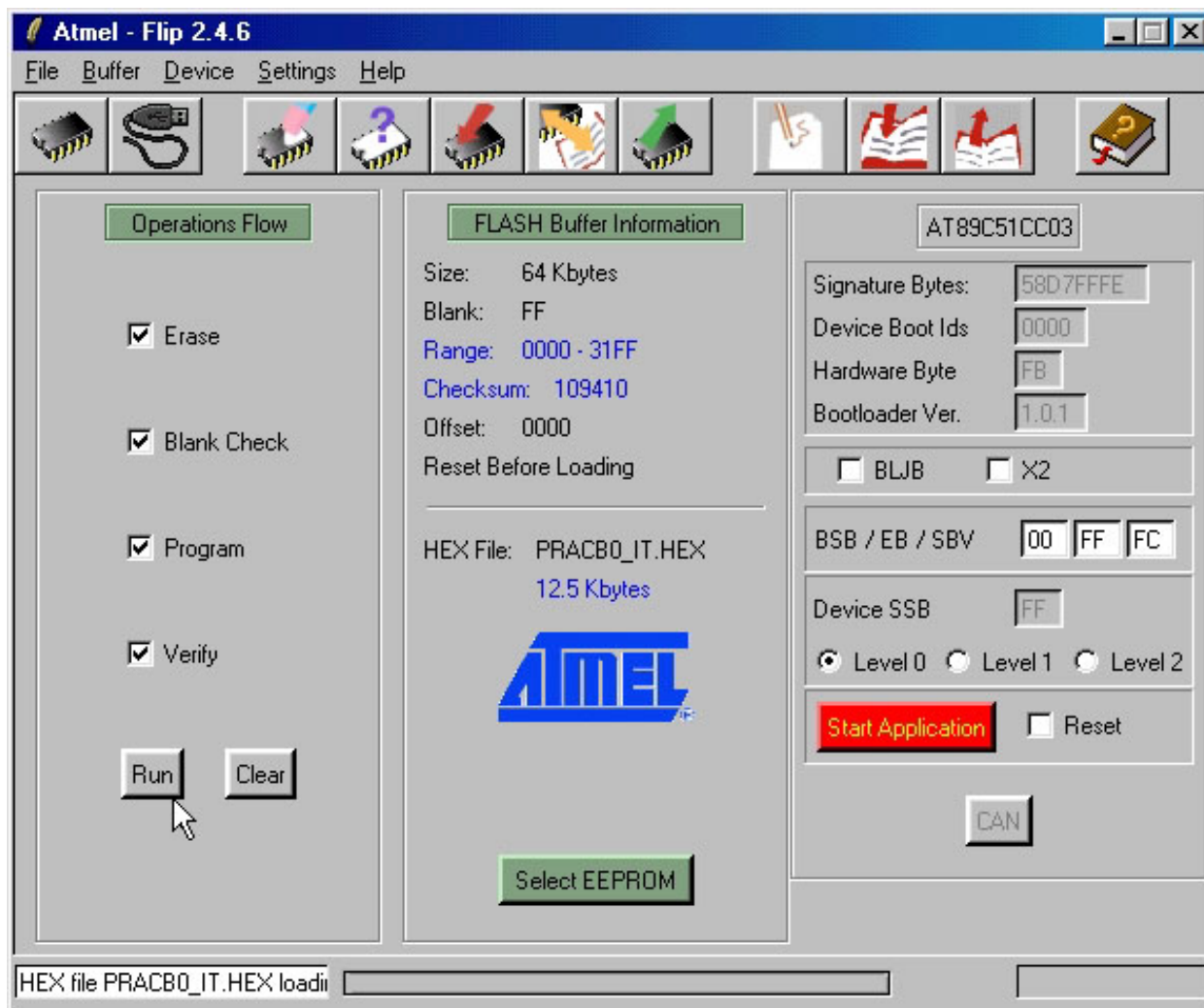


FIGURA 21: FINESTRA SETTAGGIO FLIP (3 DI 3)

C13) Premere il pulsante **Run** nella finestra principale per avviare le operazioni **ISP** impostate, che programmeranno la **FLASH** interna con il codice caricato.

C14) Attendere la fine delle operazioni **ISP**. Nel riquadro in basso a sinistra si può vedere l'operazione in corso affiancata da una barra a scorrimento che indica il suo stato di avanzamento; le caselle di spunto diventano rosse durante l'esecuzione e poi verdi man mano che la rispettiva operazione viene completata. Si deve quindi aspettare che la casella *Verify* diventi verde.

C15) A questo punto la **FLASH** è programmata ed il **FLIP** può essere chiuso.

C16) Riprovare il programma demo appena riprogrammato ripetendo i passi B1÷B6.



Al fine di velocizzare le operazioni sopra descritte, che devono essere ripetute ad ogni prova del programma applicativo, si può conveniente usare la modalità **BatchISP** del **FLIP** corredata di un adeguato file di comando, con le seguenti istruzioni:

```
-device AT89C51CC03
-hardware RS232
-port COM1
-baudrate 115200
-operation
memory FLASH
erase F
loadbuffer "<file utente>.HEX"
addrange 0x0000 0xFFFF
program
verify
```

Per ulteriori informazioni sulla programmazione **ISP** e sull'uso del programma **FLIP** si prega di consultare la specifica documentazione rilasciata dalla **ATMEL**.

## D) CREAZIONE DEL CODICE ESEGUIBILE DEL PROGRAMMA DEMO

- D1) Installare l'ambiente di sviluppo scelto per realizzare programma applicativo, sul disco rigido del **PC**. Come descritto nel capitolo **AMBIENTI DI SVILUPPO** sono disponibili diversi pacchetti in modo da soddisfare le richieste di ogni utente, ma qui si presentano quelli più diffusi, ovvero il **BASCOM 8051**, **µC/51** e **LADDER WORK**. Per informazioni dettagliate su questi prodotti si faccia riferimento al relativo manuale d'uso oppure all'aiuto in linea.
- D2) Accertarsi che al punto C4 dal **CD grifo®** siano stati copiati tutti i file e non solo l'eseguibile **.HEX**, del demo. In dettaglio nella cartella di lavoro devono essere presenti il/i file sorgenti, i file di progetto, i file di definizione provvisti delle seguenti estensioni a seconda dell'ambiente di sviluppo usato:

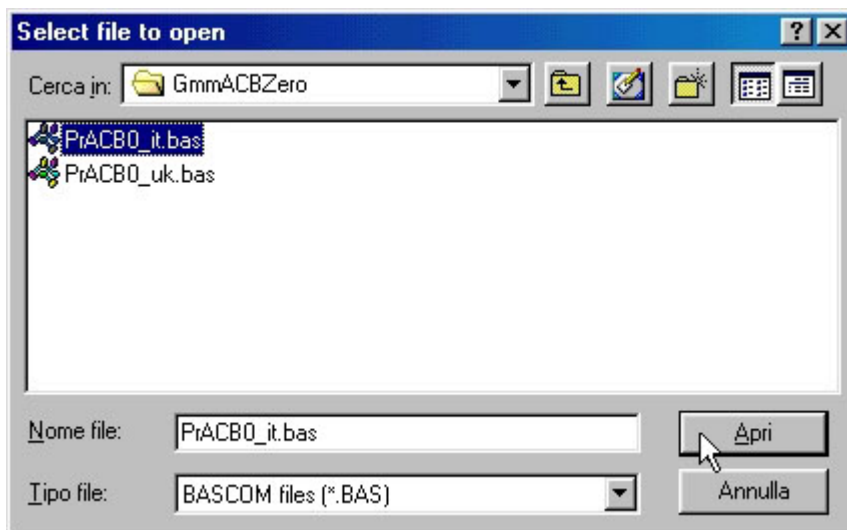
Ambiente di sviluppo	Linguaggio	Sorgente	Progetto	Definizioni	Ambiente
<b>BASCOM 8051</b>	BASIC	.BAS	-	.DAT	-
<b>µC/51</b>	C	.C	.MAK	.H	.WSP
<b>LADDER WORK</b>	LADDER	.PJM	.PJM	-	-

- D3) Ricompilare il sorgente usando l'ambiente di sviluppo scelto, in modo da ottenere il file **.HEX** identico a quello ricevuto e già usato nel punto C. Questa operazione si differenzia notevolmente a seconda dell'ambiente di sviluppo utilizzato, pertanto di seguito vengono esposti i passi dettagliati opportunamente suddivisi:

### D3A) RICOMPILAZIONE CON BASCOM 8051

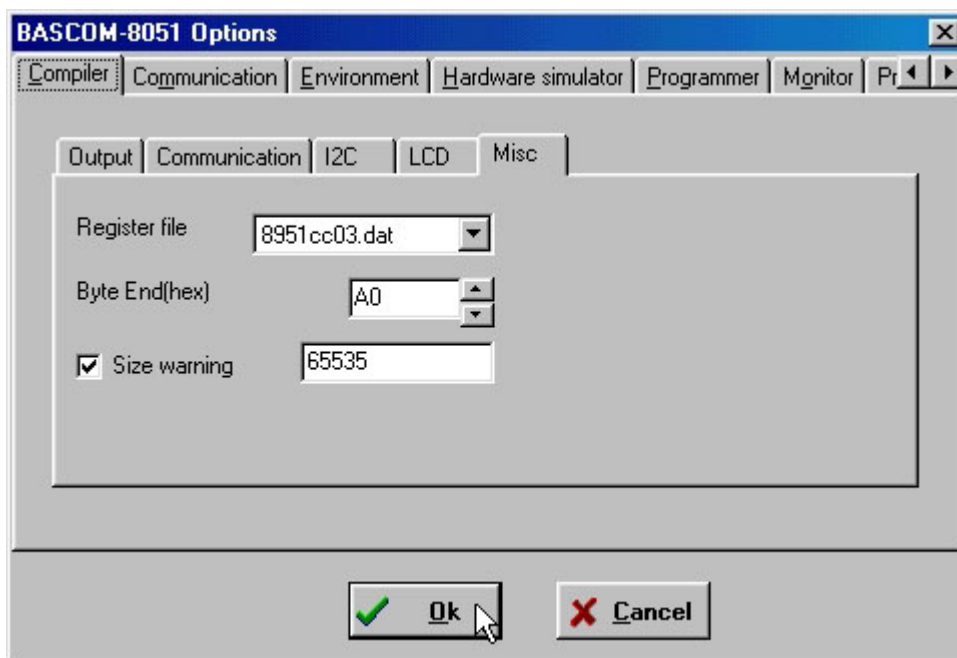
- D3A1) Copiare il file di definizione **.DAT** descritto al punto D2 nella cartella di installazione del **BASCOM 8051**: `\MCS Electronics\BASCOM8051\`

- D3A2) Eseguire il **BASCOM 8051** ed una volta entrati nel suo **IDE**, caricare il file sorgente (con estensione **.BAS**), tramite il menu *File / Open*:



**FIGURA 22: CARICAMENTO SORGENTE CON BASCOM 8051**

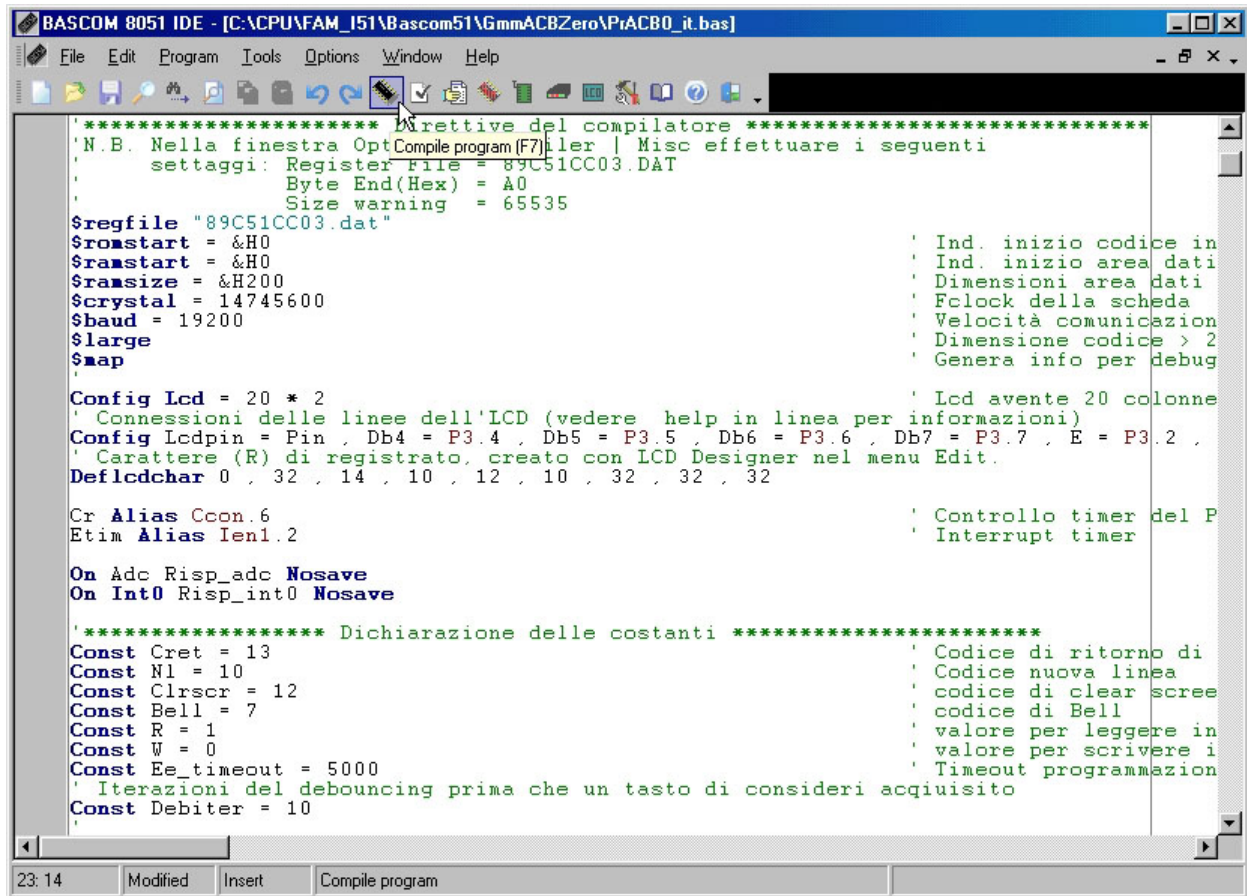
- D3A3) Aprire la finestra di configurazione del compilatore **BASCOM 8051**, tramite il comando *Option/Compiler/Misc*, effettuare le impostazioni riportate nella seguente figura ed al termine confermare con il pulsante *Ok*.  
Si ricorda che l'impostazione del campo *Register file* deve coincidere con il microprocessore usato ed é possibile solo se il file di definizione é stato correttamente copiato, come descritto al punto D3A1.



**FIGURA 23: CONFIGURAZIONE COMPILATORE BASCOM 8051**

- D3A4) Ricompilare il sorgente del programma demo con una semplice pressione del tasto rapido *F7*, oppure selezionare il comando *Program / Compile*, e verificare che non avvengano errori. In questo modo si deve ottenere il file **.HEX**, identico a quello presente sul disco **grifo®** e già usato nei punti C.

La compilazione impiega un tempo che dipende dal PC usato; in ogni caso si deve attendere il completamento di entrambe le passate opportunamente segnalate da una finestra di stato che compare durante la stessa compilazione, e verificare che al termine non siano riportati errori nella parte bassa dell'IDE. In altre parole al termine si deve presentare una situazione simile a quella riportata nella seguente figura.



```

***** Direttive del compilatore *****
' N.B. Nella finestra Opt Compile program (F7) | Misc effettuare i seguenti
  settaggi: Register File = 89C51CC03.DAT
            Byte End(Hex) = A0
            Size warning = 65535
$regfile "89C51CC03.dat"
$romstart = &H0
$ramstart = &H0
$ramsize = &H200
$crystal = 14745600
$baud = 19200
$large
$map
' Ind. inizio codice in
' Ind. inizio area dati
' Dimensioni area dati
' Fclock della scheda
' Velocità comunicazione
' Dimensione codice > 2
' Genera info per debug

Config Lcd = 20 * 2
' Connessioni delle linee dell'LCD (vedere help in linea per informazioni)
' Lcd avente 20 colonne
Config Lcdpin = Pin , Db4 = P3.4 , Db5 = P3.5 , Db6 = P3.6 , Db7 = P3.7 , E = P3.2 ,
' Carattere (R) di registrato, creato con LCD Designer nel menu Edit.
Deflcdchar 0 , 32 , 14 , 10 , 12 , 10 , 32 , 32 , 32

Cr Alias Coon.6
Etim Alias Ien1.2
' Controllo timer del P
' Interrupt timer

On Adc Risp_adc Nosave
On Int0 Risp_int0 Nosave

***** Dichiarazione delle costanti *****
Const Cret = 13
Const N1 = 10
Const Clrscr = 12
Const Bell = 7
Const R = 1
Const W = 0
Const Ee_timeout = 5000
' Iterazioni del debouncing prima che un tasto di consideri acquisito
Const Debiter = 10
' Codice di ritorno di
' Codice nuova linea
' codice di clear scree
' codice di Bell
' valore per leggere in
' valore per scrivere i
' Timeout programmazione

```

FIGURA 24: COMPILAZIONE CON BASCOM 8051

### D3B) RICOMPILAZIONE CON $\mu$ C/51

D3B1) Il pacchetto di sviluppo  $\mu$ C/51 è provvisto di due ambienti distinti. Il primo, coincide con l'IDE JFE editor ovvero un programma che comprende un editor, un rappresentatore di dipendenze ad albero ed un esecutore di programmi esterni. Questo può essere configurato con appositi programmi in modo da effettuare tutte le operazioni necessarie allo sviluppo del programma applicativo.

Il secondo comprende un editor denominato **uEdit** con cui si può esaminare e modificare il sorgente ed un gestore di progetti, denominato **UmShell**, che può compilare il sorgente ed ottenere quindi il codice eseguibile per la **GMM 518 Zero**.

I seguenti passi illustrano le operazioni basilari per entrambi gli ambienti del  $\mu$ C/51 mentre per informazioni dettagliate sull'uso dei vari programmi descritti si rimanda alla documentazione fornita con il pacchetto.

D3B2) Usando il **JFE** editor si deve prima lanciare l'omonimo programma dal menù di avvio di **Windows** e quindi aprire il file di ambiente relativo al demo, con estensione **.WSP**, tramite il menu *File / Open.Workspace*. Qualora questo file non sia disponibile l'utente può generarlo, oppure usare gli altri programmi uEdit e UmShell descritti dopo.

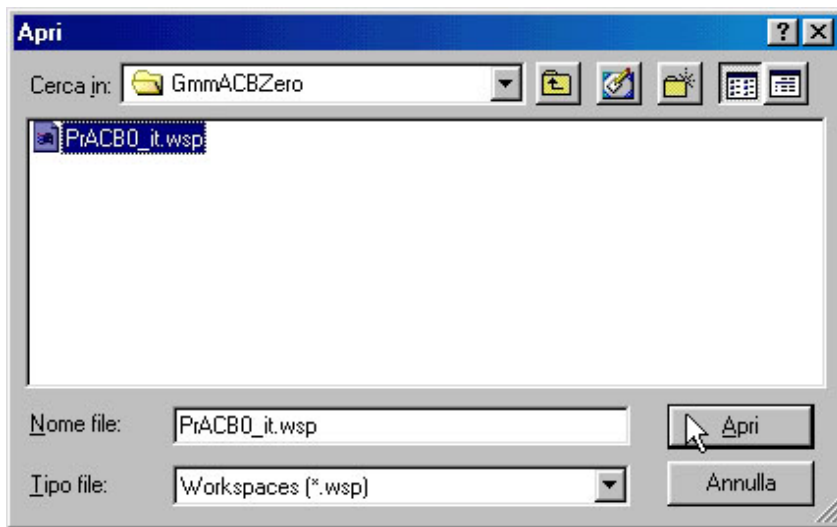


FIGURA 25: CARICAMENTO AMBIENTE CON  $\mu$ C/51 E JFE

A questo punto tutto é già predisposto e per creare il codice eseguibile del programma ed é sufficiente premere il pulsante **RE-MAKE** sulla riga dei pulsanti del **JFE**, verificando che non avvengano errori, come illustrato nella seguente figura.

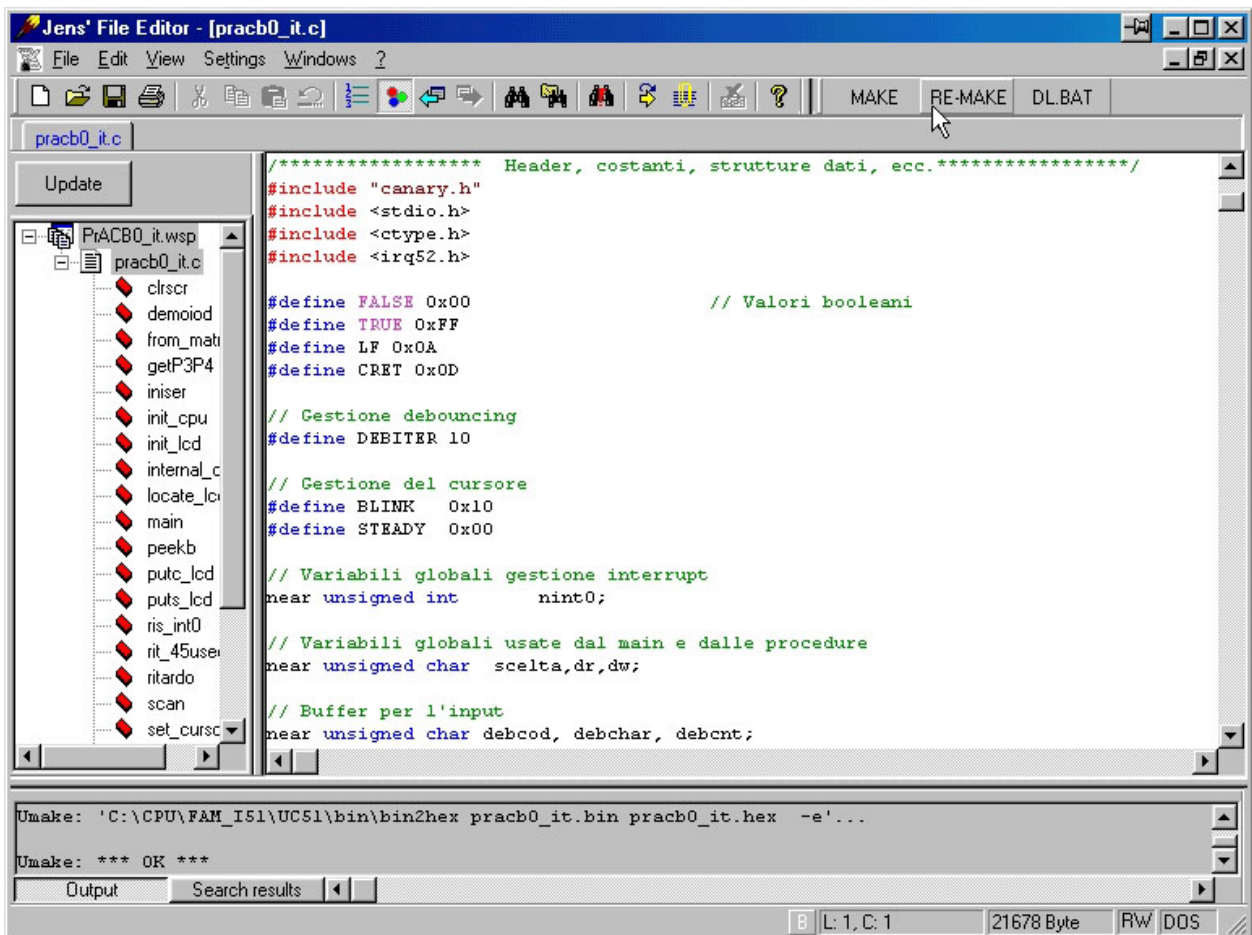


FIGURA 26: COMPILAZIONE CON  $\mu$ C/51 E JFE

Al termine della compilazione si ottiene il file **.HEX**, identico a quello presente sul disco **grifo®** e già usato nei punti C.

- D3B3) Usando invece il **uEdit** e l'**UmShell** si deve innanzitutto lanciare il primo programma dal menù di avvio di Windows e quindi aprire il file sorgente del demo, con estensione **.C**, tramite il menu *File / Open*:

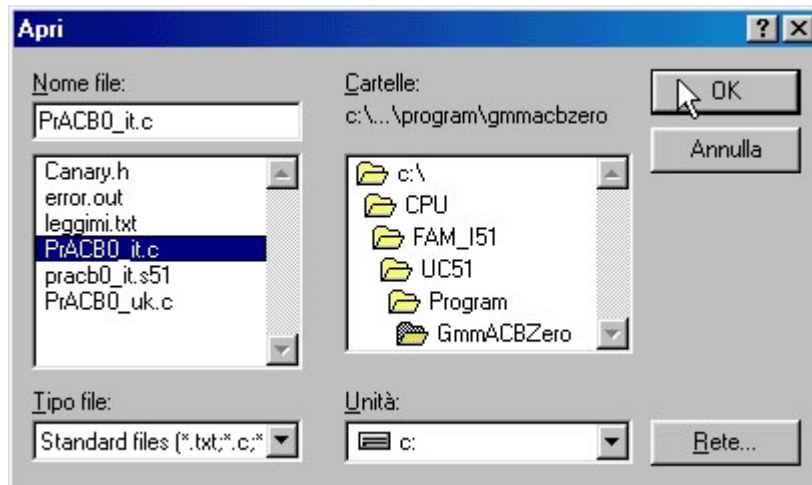


FIGURA 27: CARICAMENTO SORGENTE CON  $\mu$ C/51 ED UEDIT

- D3B4) Continuando ad usare il **uEdit** e l'**UmShell** si deve poi lanciare il secondo programma dal menù di avvio di Windows e quindi aprire il file di progetto del demo, con estensione **.MAK**, tramite il menu *File / Open*:

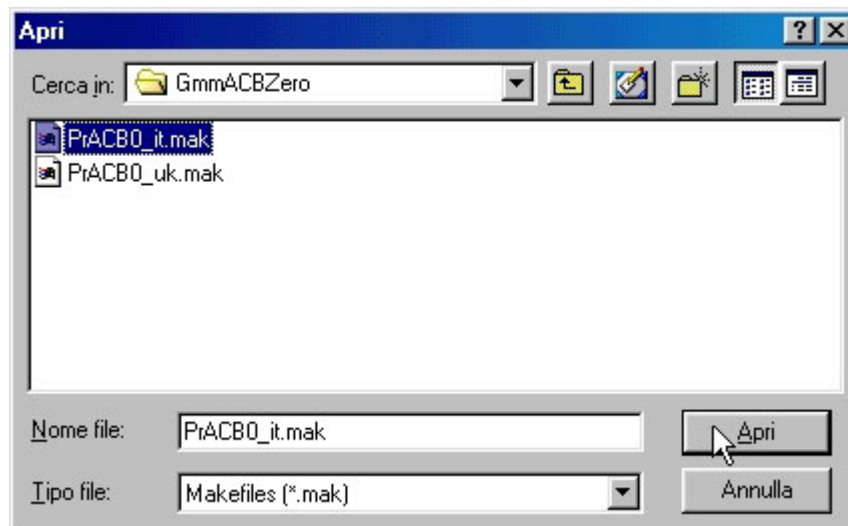


FIGURA 28: CARICAMENTO PROGETTO CON  $\mu$ C/51 ED UMSHELL

Il file di progetto definisce tutte le modalità di compilazione del sorgente e stabilisce quindi le caratteristiche del programma eseguibile generato, in termini di allocazione codice, allocazione area dati, ottimizzazioni, gestione della console, ecc.

- D3B5) Usando sempre il **uEdit** e l'**UmShell**, a questo punto si deve compilare il sorgente premendo il secondo pulsante da destra dell'**UmShell**, oppure usando il suo menù *Make / Remake Target*. Durante la compilazione si deve verificare che non avvengano

errori ed al termine si ottiene il file **.HEX**, identico a quello presente sul disco **grifo®** e già usato nei punti C.

La compilazione impiega un tempo che dipende dal **PC** usato, in ogni caso si deve attendere il completamento di tutte le fasi opportunamente segnalate nella finestra dell'**UmShell** in modo da ottenere una condizione simile a quella riportata nella seguente figura.

```

UmShell - C:\CPU\FAM_I51\UC51\Program\GmmACBZero\PrACB0_it.mak
File Edit Search Make Options Help
pracb0_it.hex
Umake: 'C:\CPU\FAM_I51\UC51\bin\uc51 pracb0_it.c pracb0_it.s51 -IC:\CPU\FAM_I51\UC51\include
Umake: 'C:\CPU\FAM_I51\UC51\bin\as51 -e pracb0_it.s51 -iC:\CPU\FAM_I51\UC51\include -g'...
Umake: 'C:\CPU\FAM_I51\UC51\bin\farg link.lnk pracb0_it.obj'...
Umake: 'C:\CPU\FAM_I51\UC51\bin\l51 -e -opracb0_it.bin @link.lnk -IC:\CPU\FAM_I51\UC51\lib\sm
Binary 'pracb0_it.bin': start: $0, end: $1850, total size: $1851/(dec.)6225 bytes, used (dec.
Sclass 'text': start: $0, end: $1850 size: $1851/(dec.)6225 Bytes
Sclass 'dram': start: $20, end: $40 size: $21/(dec.)33 Bytes
Sclass 'iram': start: $41, end: $49 size: $9/(dec.)9 Bytes
Umake: 'C:\CPU\FAM_I51\UC51\bin\bin2hex pracb0_it.bin pracb0_it.hex -e'...

Umake: *** OK ***

Complete Remake of the main Target
    
```

FIGURA 29: COMPILAZIONE CON  $\mu$ C/51 ED UMShell

### D3C) RICOMPILAZIONE CON LADDER WORK

D3C1) Una volta lanciato l'**IDE** del **LADDER WORK**, dal menù di avvio di Windows, aprire lo schematico del programma demo (con estensione **.PJM**) tramite il menu *File / Open*:

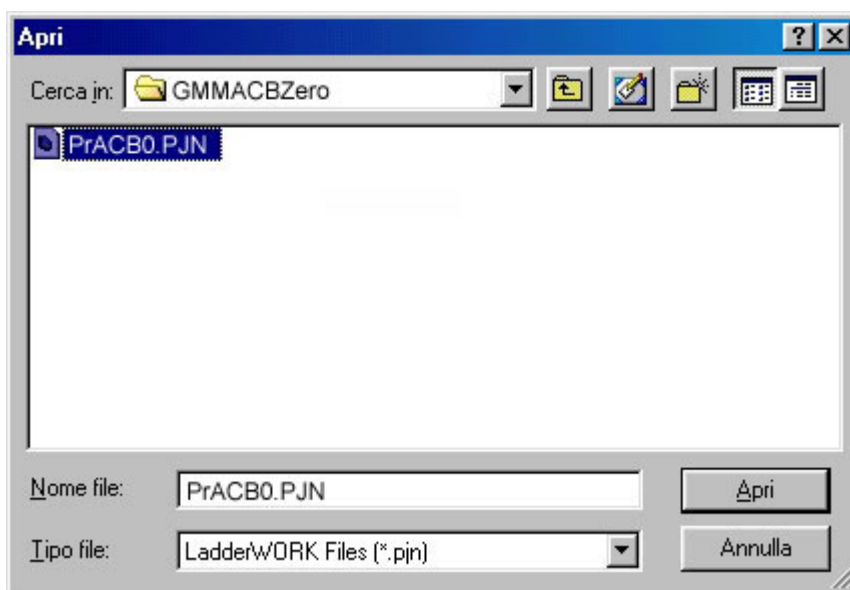


FIGURA 30: CARICAMENTO SCHEMATICO CON LADDER WORK

D3C2) Assicurarsi che il profilo scelto per la compilazione sia quello relativo al **Mini Modulo GMM ACB Zero**:



FIGURA 31: CONFIGURAZIONE COMPILATORE CON LADDER WORK

D3C3) Compilare il sorgente premendo il primo pulsante da destra:



FIGURA 32: COMPILAZIONE CON LADDER WORK

D3C4) Durante la compilazione non si devono presentare errori come nella seguente figura:

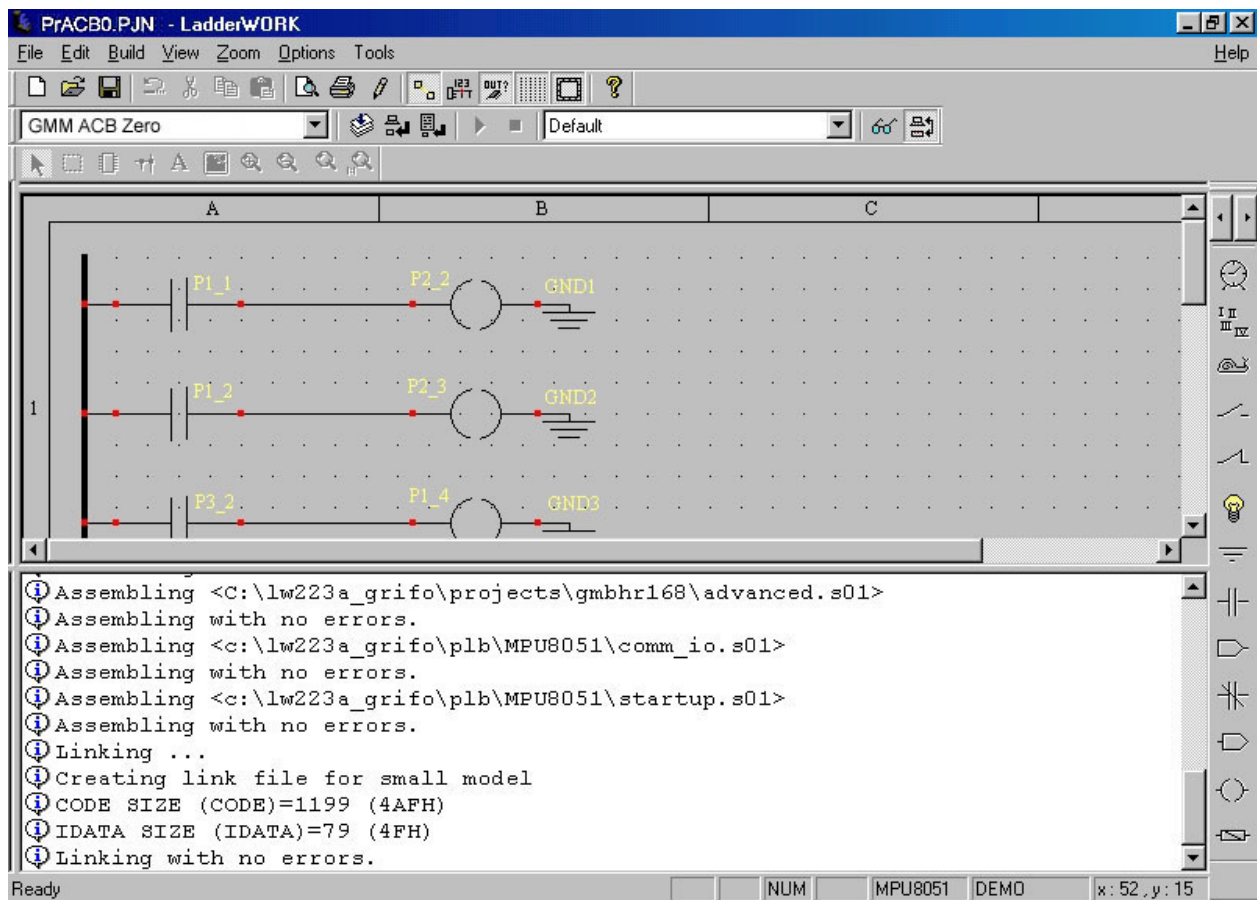


FIGURA 33: RISULTATO COMPILAZIONE CON LADDER WORK

La compilazione impiega un tempo che dipende dal **PC** usato, in ogni caso si deve attendere il completamento di tutte le fasi opportunamente segnalate nella finestra in basso dell'**IDE**. Al termine si ottiene il file **.HEX**, identico a quello presente sul disco **grifo®** e già usato nei punti C.

D4) Rieffettuare il salvataggio e la prova del file **.HEX** ottenuto dalla compilazione, ripetendo i punti **C5÷C16**.

D5) 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 518 Zero**. A questo punto è possibile modificare il sorgente del programma demo in modo da soddisfare le richieste dell'applicazione da realizzare e ricompilarlo, riprogrammarlo e riprovarlo (passi da B1 a D4) in modo ciclico, fino a quando si ottiene la funzionalità desiderata.

In merito alle impostazioni del **FLIP** si ricorda che queste possono essere effettuate solo la prima volta infatti lo stesso programma mantiene gli ultimi settaggi utilizzati; in alternativa si può usare la modalità batch, che oltretutto può essere anche integrata nell'ambiente di sviluppo, in modo da velocizzare la fase di programmazione e prova del programma.

Raggiunto questo obiettivo si può eliminare il **PC** di sviluppo, ovvero:

## E) PREPARAZIONE DEFINITIVA DELL'APPLICAZIONE

E1) Impostare modalità **RUN** (Jumper **J1.b** della **GMM 518 Zero** in posizione **Non Connesso** = Jumper Aperto) e scollegare il **PC** di sviluppo, se non richiesto dall'applicazione.



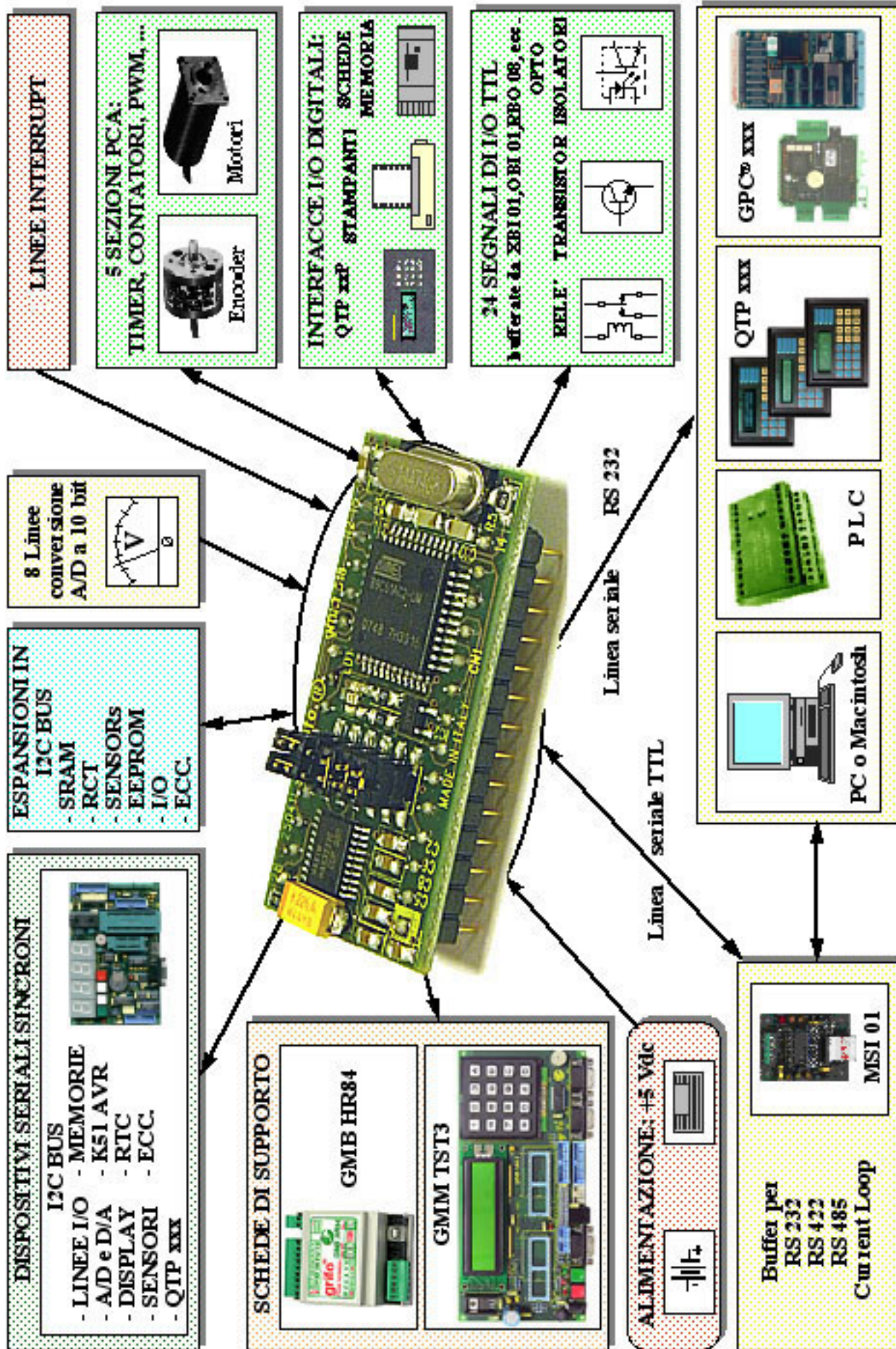


FIGURA 34: ESEMPI DI CONNESSIONE

## AMBIENTI DI SVILUPPO

La scheda usufruisce di una ricca serie di strutture software che consentono di utilizzarne al meglio le caratteristiche e di sviluppare le applicazioni necessarie in un tempo veramente corto. In generale il **Mini Modulo** può sfruttare tutte le risorse software per il microprocessore montato, ovvero i numerosi pacchetti ideati per la famiglia **8051**, sia ad alto che a basso livello. Tutti gli ambienti di sviluppo software forniti dalla **grifo®** sono sempre accompagnati da esempi che illustrano come gestire le sezioni della scheda, da una completa documentazione d'uso e da abbondanti commenti. Tra questi ricordiamo:

**SYS51CW**: Cross compilatore per programmi scritti in **C**, disponibile in ambiente **WINDOWS** con un comodo **IDE** che mette a disposizione: editor, **Compilatore C**, Assemblatore, Ottimizzatore, Linker, Librerie ed un Debugger Simbolico Remoto.

**SYS51PW**: Cross compilatore per programmi scritti in **PASCAL**, disponibile in ambiente **WINDOWS** con un comodo **IDE** che mette a disposizione: Editor, Compilatore **PASCAL**, Assemblatore, Ottimizzatore, Linker, Librerie ed un Debugger Simbolico Remoto.

**DDS MICRO C 51**: E' un comodo pacchetto software, a basso costo, che tramite un completo **IDE** permette di utilizzare un editor, un **Compilatore C** (integer), un Assemblatore, un Linker e un Remote Debugger abbinato ad un Monitor. Sono inclusi i sorgenti delle librerie ed una serie di utility.

**BASCOM 8051**: 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 e numerose istruzioni dedicate alle tipiche risorse hardware usate nell'automazione industriale.

**µC/51**: E' un comodo pacchetto software, a basso costo, che tramite un completo **IDE** permette di utilizzare un editor, un **Compilatore ANSI C**, un Assemblatore, un Linker e un Debugger Remoto configurabile da utente, a livello Sorgente. Sono inclusi i sorgenti delle librerie fondamentali e del deDugger Remoto, alcuni esempi di utilizzo e vari programmi di utilità.

**LADDER WORK**: E' un semplice sistema per creare programmi di automazione con la conosciuta e diffusa logica a contatti. Include un editor grafico che consente di posizionare e collegare i componenti hardware della scheda (input, output, contatori, **A/D**, ecc) come su uno schema elettrico e di definirne le proprietà, un efficiente compilatore che converte lo schema in codice eseguibile ed utilità per lo scaricamento di tale codice verso la scheda. Il tutto integrato in un comodo **IDE** per **Windows**. Viene fornito sotto forma di **CD** che comprende esempi e manuale d'uso e relativa chiave di abilitazione.

## DESCRIZIONE SOFTWARE 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 microcontrollore (Timer, Counter, PCA, Port I/O, A/D, UART seriale, Interrupt, Watch Dog, controllore **CAN**, ecc.); per la gestione di quest'ultime si faccia riferimento all'APPENDICE A di questo manuale.

Nei paragrafi successivi si usano le indicazioni **D0÷D7** e **.0÷7** per fare riferimento ai bits della combinazione utilizzata nelle operazioni di **I/O**.

### INGRESSI DI CONFIGURAZIONE

Una delle vie del Jumper montato a bordo della **GMM 518 Zero**, in particolare il **J1.c**, può essere acquisite via software, effettuando una semplice operazione di input dai relativi bit del Port **0**:

**J1.c -> P0.0**

L'acquisizione è in logica negata, ovvero il Jumper Chiuso fornisce lo stato logico **0 (L)** mentre il Jumper Aperto fornisce lo stato logico **1 (H)**.

**J1.b -> Connesso = DEBUG**

**J1.b -> Non Connesso = RUN**

Il Jumper **J1.b** svolge la funzione di selettore delle modalità **RUN** o **DEBUG**, ovvero se al momento dell'accensione, o dopo un **Reset**, è Connesso (= Jumper Chiuso) viene eseguito il **Boot Loader**.

Se il Jumper **J1.b** è Non Connesso (= Jumper Aperto) viene eseguito il programma utente a partire dall'indirizzo **0** dell'area codice, allocata nella **FLASH** interna.

### LED DI SEGNALAZIONE

Il **LED** può essere pilotato dal software ed il suo stato può essere definito mediante semplici operazioni sui relativi bit del **Port 0**:

**LD1 Spento (rosso) -> P2.7 = 1**

**LD1 Acceso (rosso) -> P2.7 = 0**

Il pilotaggio del **LED** è in logica negata, ovvero, il **LED** si accende ponendo a **0 (Logica L)** il corrispondente bit del **Port 0** mentre si spegne ponendo ad **1 (Logica H)** lo stesso bit.

Tutti i segnali del port **P0** vengono mantenuti alti in fase di **Reset** od accensione, di conseguenza durante ed in seguito ad una di queste fasi, il **LED** è spento.

## SEGNALI MULTIFUNZIONE

Numerosi segnali presenti su **CN1** possono essere utilizzati dall'utente con diverse funzionalità selezionabili via software dal programma applicativo sviluppato. Questa specializzazione dei segnali multiplexati avviene tramite appositi settaggi di registri interni al microcontrollore, come descritto nella documentazione del componente. Ad esempio i piedini **20÷27** possono operare come ingressi analogici (**ANn**) oppure come **I/O** digitali (**P1.n**) a seconda del settaggio del registro **ADCF**; quando settati come **I/O** possono poi operare come segnali del **Timer 2** o del **PCA**, attivando opportunamente queste due periferiche.

## ACCESSO ALLE MEMORIE

Sul **Mini Modulo GMM 518 Zero** sono disponibili diversi tipi di memorie che possono essere gestite dal programma applicativo dell'utente, come indicato di seguito:

<i>Memoria</i>	<i>Allocazione</i>	<i>Accesso</i>	<i>Gestione</i>
64K bytes di FLASH	Area codice	R W	Istruzioni lettura area codice Chiamate a funzioni del Boot Loader
256 bytes di IRAM	Area dati interna	R/W	Istruzioni per RAM interna sia ad accesso diretto che indiretto
2K bytes di ERAM	Area dati esterna	R/W	Istruzioni per accesso a RAM esterna
2K bytes di EEPROM	Area dedicata	R/W	Tramite appositi registri di gestione (SFR)

Le modalità di indirizzamento e le istruzioni di accesso alle memorie del microcontrollore vengono spiegate nei manuali del componente, pertanto si prega di consultare questi ultimi o l'APPENDICE A di questo manuale. Questa conoscenza diventa necessaria quando il programma dell'utente é sviluppato in assembly, mentre utilizzando un ambiente di sviluppo ad alto livello il tutto avviene automaticamente e comodamente, ad esempio aggiungendo degli identificatori nella dichiarazione delle variabili, a seconda di dove devono essere allocate.

E' importante ricordare che gli ultimi **8** byte della **EEPROM** di bordo sono riservati per dati di calibrazione e non devono essere modificati. Sia il programma applicativo utente, che la gestione della **EEPROM** in **ISP**, devono salvaguardare questi bytes pena il malfunzionamento del **Mini Modulo**.

## 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 518 Zero**.

Manuale ATMEL:	<i>Data sheet AT89C51CC03</i>
Manuale MAXIM:	<i>New Releases Data Book - Volume IV</i>
Manuale MAXIM:	<i>New Releases Data Book - Volume V</i>
Documentazione tecnica MAXIM:	<i>True RS 232 Transceivers</i>
Manuale PHILIPS:	<i>80C51 - Based 8-Bit Microcontrollers</i>
Manuale PHILIPS:	<i>Application notes and development tools for 80C51 microcontrollers</i>

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.

Molti manuali in formato elettronico sono reperibili anche presso il nostro archivio di documentazione tecnica, come indicato in APPENDICE A.



## APPENDICE A: DESCRIZIONE COMPONENTI DI BORDO

La **grifo®** fornisce un servizio di documentazione tecnica totalmente gratuito attraverso il proprio sito internet da cui possono essere scaricate le documentazioni tecniche complete dei componenti usati a bordo scheda. Si rimanda quindi l'utente a tali manuali scaricabili dalla pagina "Servizio documentazione Tecnica" di cui in questo capitolo vengono riportate solo le prime pagine.

### AT89C51CC03

#### Features

- 80C51 Core Architecture
- 256 Bytes of On-chip RAM
- 2048 Bytes of On-chip ERAM
- 64K Bytes of On-chip Flash Memory
  - Data Retention: 10 Years at 85°C
  - Read/Write Cycle: 100K
- 2K Bytes of On-chip Flash for Bootloader
- 2K Bytes of On-chip EEPROM
  - Read/Write Cycle: 100K
- 14-sources 4-level Interrupts
- Three 16-bit Timers/Counters
- Full Duplex UART Compatible 80C51
- Maximum Crystal Frequency 40 MHz
  - In X2 Mode, 20 MHz (CPU Core, 20 MHz)
- Five Ports: 32 + 4 Digital I/O Lines
- Five-channel 16-bit PCA with
  - PWM (8-bit)
  - High-speed Output
  - Timer and Edge Capture
- Double Data Pointer
- 21-bit WatchDog Timer (7 Programmable Bits)
- A 10-bit Resolution Analog to Digital Converter (ADC) with 8 Multiplexed Inputs
- SPI Interface, (PLCC52, VPPF64 and CABGA 64 packages only)
- Full CAN Controller
  - Fully Compliant with CAN Rev 2.0A and 2.0B
  - Optimized Structure for Communication Management (Via SFR)
  - 15 Independent Message Objects
    - Each Message Object Programmable on Transmission or Reception
    - Individual Tag and Mask Filters up to 29-bit Identifier/Channel
    - 8-byte Cyclic Data Register (FIFO)/Message Object
    - 16-bit Status and Control Register/Message Object
    - 16-bit Time-Stamping Register/Message Object
    - CAN Specification 2.0 Part A or 2.0 Part B Programmable for Each Message Object
    - Access to Message Object Control and Data Registers Via SFR
    - Programmable Reception Buffer Length Up To 15 Message Objects
    - Priority Management of Reception of Hits on Several Message Objects at the Same Time (Basic CAN Feature)
    - Priority Management for Transmission
    - Message Object Overrun Interrupt
  - Supports
    - Time Triggered Communication
    - Autobaud and Listening Mode
    - Programmable Automatic Reply Mode
  - 1-Mbit/s Maximum Transfer Rate at 8 MHz<sup>(1)</sup> Crystal Frequency in X2 Mode
  - Readable Error Counters
  - Programmable Link to On-chip Timer for Time Stamping and Network Synchronization
  - Independent Baud Rate Prescaler
  - Data, Remote, Error and Overload Frame Handling
- On-chip Emulation Logic (Enhanced Hook System)
- Power Saving Modes
  - Idle Mode
  - Power-down Mode

1. At BRP = 1 sampling point will be fixed.



**Enhanced 8-bit  
MCU with CAN  
Controller and  
Flash Memory**

**AT89C51CC03**

**Advance  
Information**

Rev. 4182A-CAN-07/03





- Power Supply: 3 volts to 5.5 volts
- Temperature Range: Industrial (-40° to +85°C)
- Packages: VQFP44, PLCC44, VQFP64, PLCC52, CA-BGA64

## Description

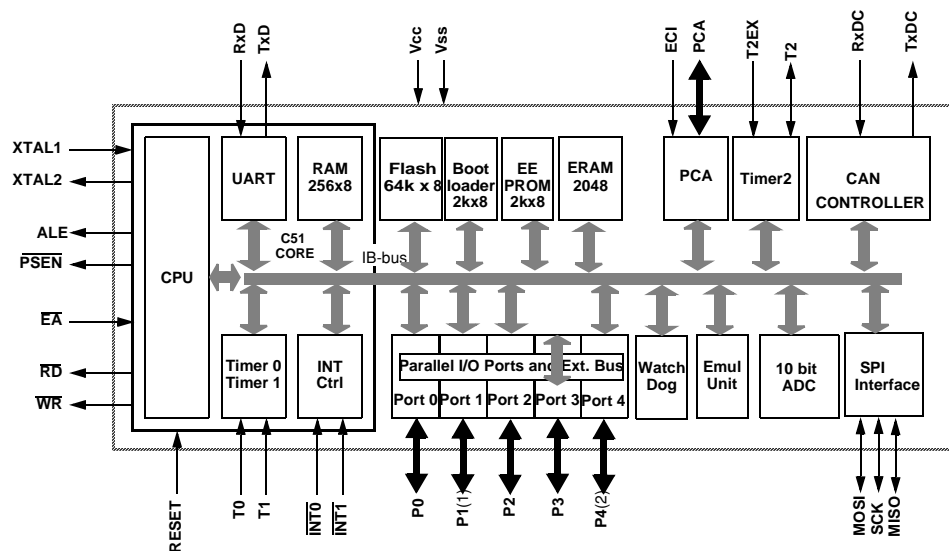
The AT89C51CC03 is the third member of the CANary™ family of 8-bit microcontrollers dedicated to CAN network applications.

In X2 mode a maximum external clock rate of 20 MHz reaches a 300 ns cycle time.

Besides the full CAN controller AT89C51CC03 provides 64K Bytes of Flash memory including In-System Programming (ISP), 2K Bytes Boot Flash Memory, 2K Bytes EEPROM and 2048 byte ERAM.

Primary attention is paid to the reduction of the electro-magnetic emission of AT89C51CC03.

## Block Diagram

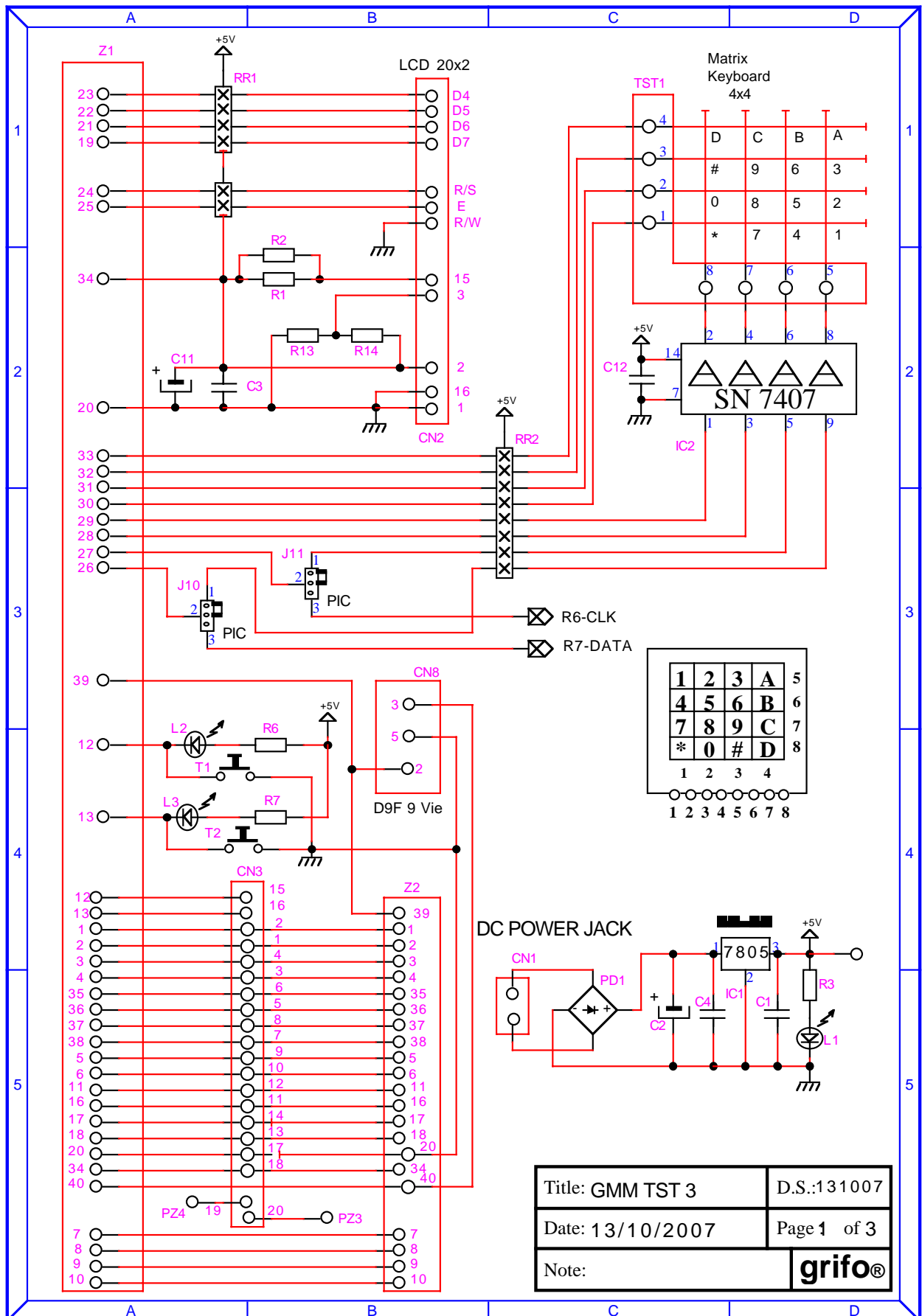


- Notes:
1. 8 analog Inputs/8 Digital I/O
  2. 5-Bit I/O Port

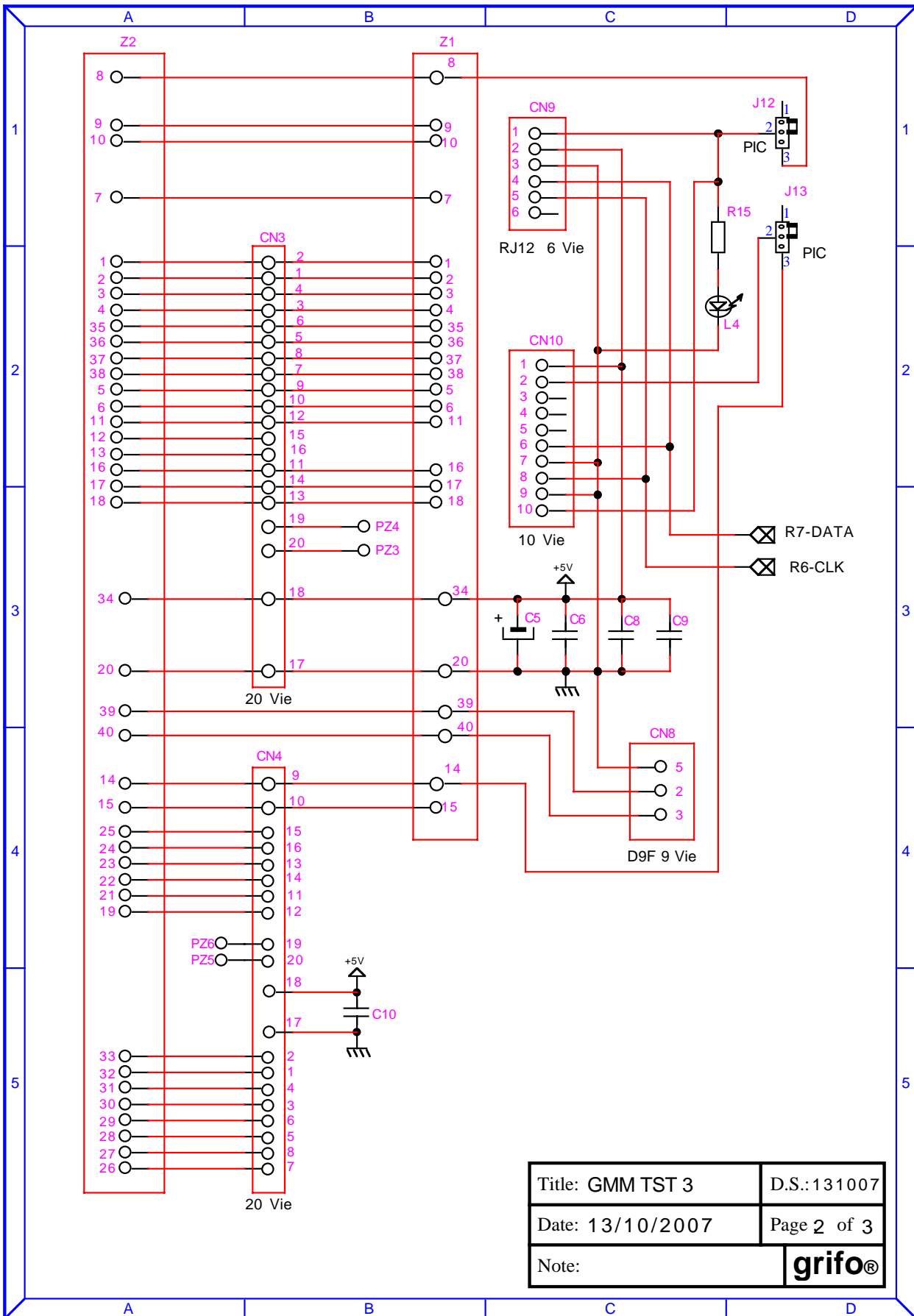


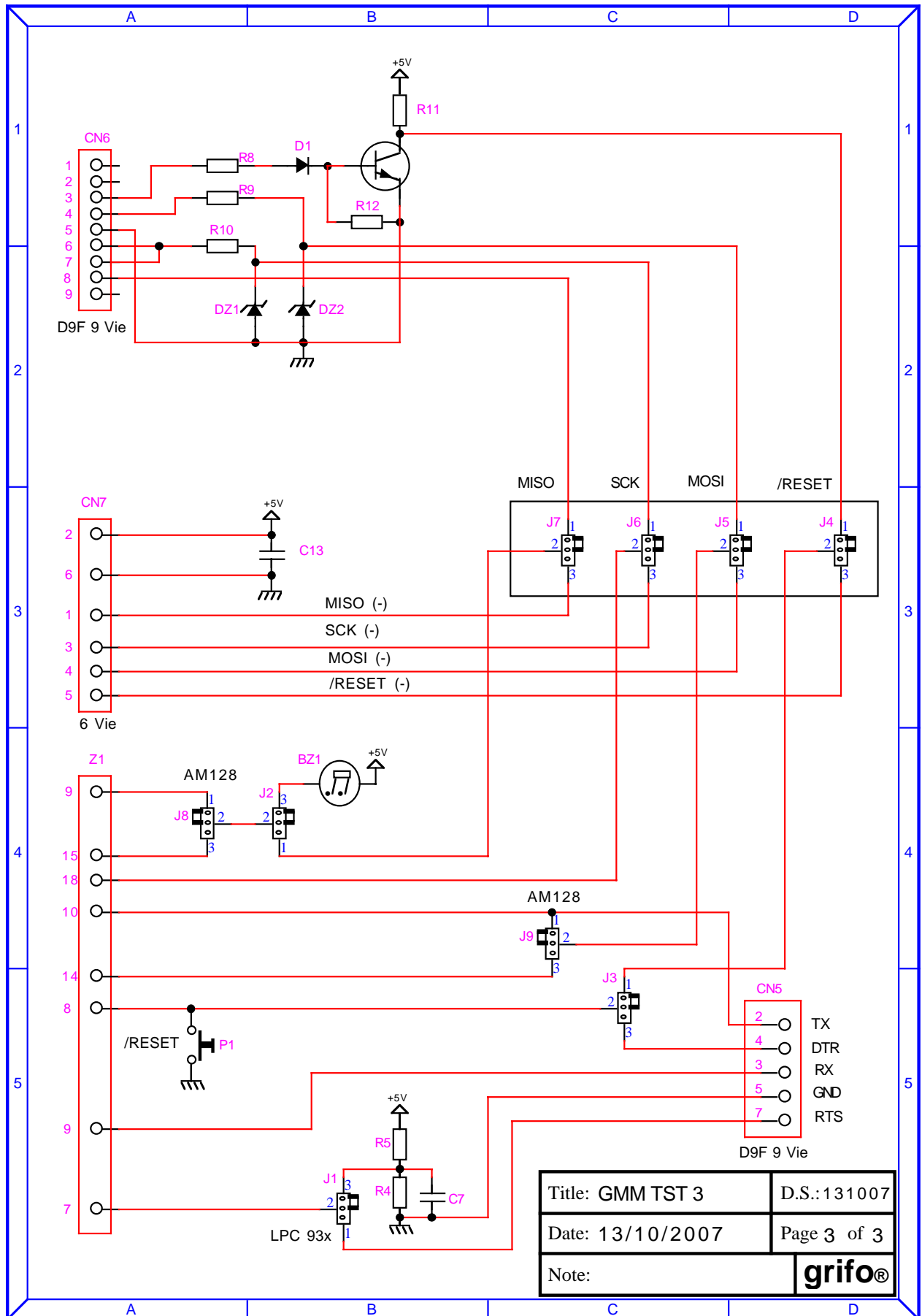


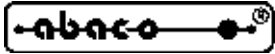
APPENDICE B: SCHEMA ELETTRICO GMM TST 3



Title: GMM TST 3	D.S.:131007
Date: 13/10/2007	Page 1 of 3
Note:	<b>grifo®</b>







ITALIAN TECHNOLOGY



## APPENDICE C: CONFIG. BASE, OPZIONI, ACCESSORI

In corrispondenza di un primo acquisto o di una eventuale riparazione, la **GMM 518 Zero** viene fornita nella sua configurazione base. Alcune caratteristiche di tale configurazione sono state descritte più volte in questo manuale (usando anche il nome configurazione di default) ed in questa appendice vengono riassunte, opportunamente raggruppate nella seguente tabella.

DISPOSITIVO	CONFIG.	FUNZIONE
<b>J1.A</b> <b>J1.B</b> <b>J1.a</b>	<b>2-3</b> <b>2-3</b> <b>Connesso</b>	<i>Configura</i> la Linea Seriale Asincrona, su <b>CN1</b> , per lo standard di comunicazione <b>RS232</b> .
<b>J1.b</b>	<b>NON</b> <b>Connesso</b>	<i>Non Collega</i> la linea <b>/PSEN</b> del microcontrollore selezionando la modalità <b>RUN</b> . A seguito di ogni accensione, e/o <b>RESET</b> del <b>Mini Modulo</b> , parte il programms scritto nella <b>FLASH</b> di bordo del microcontrollore.
<b>J1.c</b>	<b>Non</b> <b>Connesso</b>	<i>Lascia Libero</i> l'ingresso di Configurazione, acquisibile da firmware, che quindi assume il livello logico <b>1 (H)</b> .
<b>FLASH per</b> <b>Boot Loader</b>	<b>Boot</b> <b>Loader</b>	Quando é selezionata la modalità <b>DEBUG</b> (J1.b=Collegato) consente di comunicare con un PC che esegue il <b>FLIP</b> per gestire la programmazione <b>ISP</b> delle memorie <b>FLASH</b> ed <b>EEPROM</b> di bordo.
<b>FLASH</b> <b>Interna</b> <b>per Utente</b>	<b>Programma</b> <b>Demo</b>	Effettua il lampeggia del <b>LED</b> di segnalazione ed interagisce con una console sulla seriale <b>RS 232</b> impostata a 19.200 Baud, 8 bit x chr, Nessuna parità, 1 Stop bit, Nessun handshake.
<b>EEPROM</b> <b>Interna per</b> <b>Utente</b>	<b>Dati</b> <b>Calibrazio-</b> <b>ne</b>	Gli ultimi <b>8</b> byte di questa memoria sono riservati per dati di calibrazione e non devono essere modificati. Sia il programma applicativo utente che la gestione <b>ISP</b> devono salvaguardare questi bytes, pena il malfunzionamento del <b>Mini Modulo</b> .

FIGURA C1: TABELLA CONFIGURAZIONE BASE

La precedente tabella può essere ulteriormente ridotta indicando che il prodotto viene fornito calibrato, con tutte le linee di **I/O** sullo zoccolo libere e pronto per la comunicazione in **RS 232**, con un protocollo fisico **19.200, 8, No, 1**.

La **GMM 518 Zero** non ha opzioni che possono essere aggiunte in fase di ordine del modulo. Come descritto nel capitolo SCHEDE DI SUPPORTO ci sono altre schede che sono in grado di alloggiare il **Mini Modulo** e di rendere immediatamente utilizzabili tutte le sue risorse.



## APPENDICE D: INDICE ANALITICO

### Simboli

.HEX 31, 33, 34, 37, 38, 40  
/RES 17  
µC/51 33, 35, 42

### A

A/D converter 6, 12, 13, 17, 20  
Accensione 17  
Accesso 44  
Accessori C-1  
Alimentazione 13, 14, 17  
Allocazione memorie 44  
Ambienti di sviluppo 42  
Area codice 17  
Area dati 17, 23, 44  
Assistenza 1  
AT89C51CC03 12, 31, A-1  
Attributi 30

### B

BASCOM 8051 33, 35, 42  
Basso assorbimento 14  
BatchISP 33  
Baud rate 29, 31  
Bibliografia 45  
Bit di Dati 29  
Bit di Stop 29  
Bit per secondo 29  
Bit x chr 29  
Boot Loader 22, 43

### C

Calibrazione 44, C-1  
CAN 10, 16, 23  
Caratteristiche elettriche 13  
Caratteristiche fisiche 12  
Caratteristiche generali 12  
Caratteristiche tecniche 12  
Cariche elettrostatiche 1  
Cartella di lavoro 29  
Cattura 17  
Ciclo macchina 8  
Clock 8, 12, 17

Come iniziare 28  
Comparazione 17  
Comunicazione seriale 5, 18, 20, 28  
COMx 28, 29  
Configurazione base 19, C-1  
Configurazione scheda 9, 18  
Connessioni 15  
Connettori 12, 15  
    CN1 16  
Consumi 13, 14  
Contenitore 1  
Controllo alimentazione 17  
Controllo di flusso 29  
Corrente 13  
Creazione codice eseguibile 33

## D

**DDS MICRO C 51** 42  
DEBUG 18, 22, 30, 43  
Descrizione software 43  
Dimensioni 12  
Display 26  
Disposizione 15  
Disturbi 14  
Documentazione 45, A-1  
Driver 23

## E

EEPROM 8, 12, 22, 44, C-1  
EEPROM riservata 44  
Emulazione terminale 29  
ERAM 8, 12, 44  
Esempi di connessione 41

## F

File di ambiente 36  
File di definizione 33  
File di progetto 37  
File forniti 33  
File sorgente 34, 37  
FLASH 8, 12, 22, 32, 44, C-1  
**FLIP** 29, 32  
Frequenza clock 12



**G**

Garanzia 1

**GMB HR84** 24, 41**GMM TST** 26**GMM TST 3** 26, 41, B-1**H**

Handshake 29

**HYPERTERMINAL** 29**I**

I/O digitali 6, 16, 44

I2C BUS 6, 16, 20

Idle 14

Impedenza 20, 23

Impedenza ingressi analogici: 13

Informazioni generali 4

Ingressi optoisolati 24

Ingressi analogici 6, 13, 17, 20, 44

Ingressi configurazione 18, 43

Installazione 14

Interfacce operatore 41

Interfacciamento 20

Interrupts 16, 19

IRAM 8, 12, 44

ISP 22, 29, 32

**J****JFE** 36

Jumpers 10, 15, 18, 21, 22, C-1

**K****K51-AVR** 6, 41**L****LADDER WORK** 33, 38, 39, 42

LEDs 9, 14, 15, 16, 18, 43

**M**

Manuali 45

Massa 17

Memorie interne 8, 12, 44, C-1

Microcontrollore 12, 31, A-1

Modo operativo 22

**MSI 01** 5, 41

**N**

Normative 1

**O**

Opzioni C-1

**P**

Parità 29

Passo 12, 16

PC 41

PC di sviluppo 28

PCA 17, 44

Periferiche di bordo 43

Peso 12

Piante componenti 11

PLC 41

Port 6, 16, 43

Power down 14

Predisposizione collegamenti 28

Preparazione definitiva 40

Primo uso 28

Priorità 19

Programmazione FLASH 29

Programmi demo 29, 30, C-1

Protezione RS 232 13

Protocollo elettrico 20

Protocollo fisico 20, C-1

Prova programma demo 29

PWM 17, 20

**Q**

QTP 6, 41

Quarzo 8

**R**

Range ingressi analogici 13

Relè 24

Reset 17

Resistenze pull up 20

Risoluzione A/D 12

Risorse della scheda 12

RS 232 5, 13, 20, 21, 28

RUN 18, 22, 29, 40, 43, C-1

**S**

Scariche **20**  
Schede di supporto **24**  
Schema a blocchi **7**  
Schematico **38**  
Schemi elettrici **23, B-1**  
Segnalazioni visive **14**  
Segnali di controllo **17, 22**  
Segnali multifunzione **16, 44**  
Seriale asincrona **5, 20**  
Seriale RS 232 **5, 16, 18, 21, 28, C-1**  
Seriale TTL **5, 16, 18, 21**  
Settori applicativi **10**  
Sicurezza **1, 8, 17**  
Soglia alimentazione **13**  
**SYS51CW 42**  
**SYS51PW 42**

**T**

Tabella esempi **30**  
Tastiera **26**  
Temperatura **12**  
Tempo conversione A/D **12**  
Tempo intervento Watch Dog **12**  
Tempo reset **12**  
Tensione alimentazione **13**  
Tensione riferimento **13, 17**  
Timer **16**  
Trigger **16**  
TTL **5, 6, 20, 21**

**U**

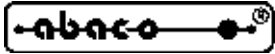
Umidità **12**  
Utilizzo **10**

**V**

Velocità **8**  
Versione **3**  
Vista complessiva **9**  
Vista dall'alto **13**  
Vref **13, 17, 20**

**W**

Watch Dog **8, 12, 17**



**Z**

Zoccolo 16

