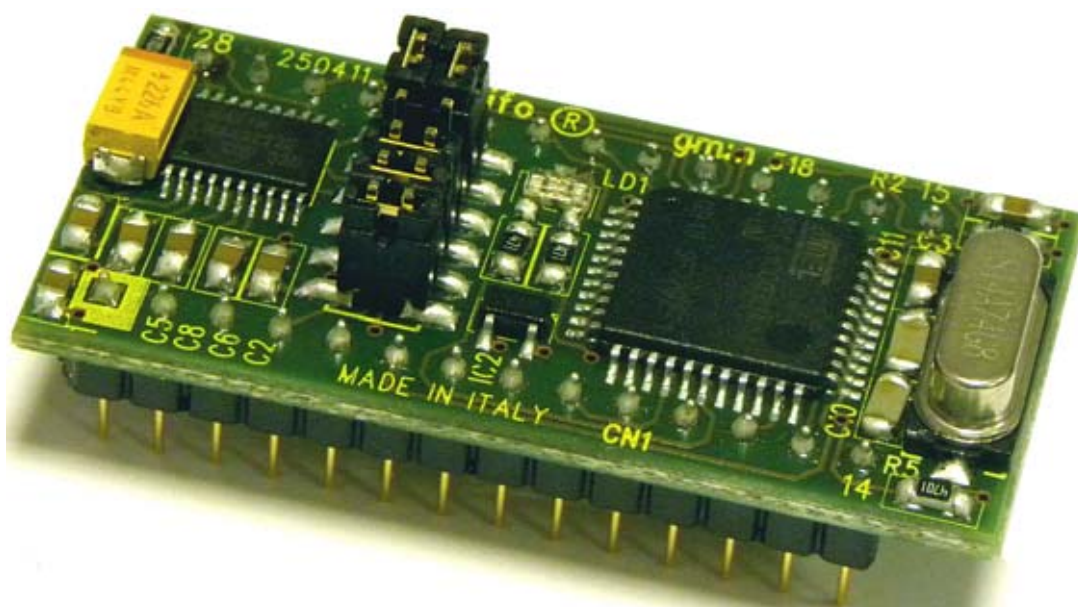


GMM 518 Zero

grifo® Mini Module Atmel AT89C51CC03

TECHNICAL MANUAL



grifo®

ITALIAN TECHNOLOGY

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

E-mail: grifo@grifo.it

<http://www.grifo.it>

<http://www.grifo.com>

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



GMM 518 Zero

Rel. 5.00 Edition 03 October 2011

 , GPC®, **grifo®**, are trade marks of **grifo®**



GMM 518 Zero

grifo® Mini Module Atmel AT89C51CC03

TECHNICAL MANUAL

Male socket with **28** pins, dual in line, **100** mils pitch, **600** mils width; very small dimension: **20,6 x 42,6 x 15 mm**; **4 layers PCB** to obtain best noisy resistance and best **EMI** performance; requires single power supply at **+5 Vdc, 10 mA**; availability of low power modes as **IDLE** and **Power Down**; **Atmel AT89C51CC03** microcontroller (**8051** code compatible) with **14.74 Mhz** crystal; programmable machine speed at **12** or **6** clock cycle; **64K FLASH** for code, **2K FLASH** for **Boot Loader**, **256** bytes **IRAM** for data, **2K** **ERAM** for data, **2K** **EEPROM** for data. **8** **A/D** converter channels with **10** bits resolution, **20 µsec** conversion time; **19** interrupt sources with **4** priority levels; **3** **Timers Counters** up to **16** bits; **5** **PCA** channels at **16** bits with **PWM**, compare, capture, etc. functionalities; **24** digital **I/O** lines available on connector; some of these have multifunctions capabilities. Hardware serial line with **Baud Rate** programmable up to **115.200 Baud**, at **TTL** level or buffered in **RS 232** with protection against **±15 KV** discharges. **Reset** and power supply control circuit. Firmware **I2C BUS** line, available on connector; **CAN** controller compatible with **2.0A** and **2.0B** standards. **5** configuration **Jumpers** with one acquired by firmware; **1** signal **LED** managed by software, through digital **I/Os**.

Internal **FLASH** and **EEPROM** can be managed through the **In System Programming**, by using only the serial communication line, even when the module is already mounted. **Free** software for **PC** that supports the **ISP** modality in a fast and comfortable way. Wide range of development tools as: **C** compilers (**µC/51**, **SYS51CW**, **DDS Micro C51**); **BASIC** compilers (**BASCOM 8051**); **PASCAL** compilers (**SYS51PW**); **Contact Logic** (**LADDER WORK**); etc. Long list of demo programs and use examples supplied under source (duly remarked) and executable format, for the available development tools.

grifo®

ITALIAN TECHNOLOGY

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

E-mail: grifo@grifo.it

<http://www.grifo.it>

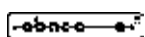
<http://www.grifo.com>

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



GMM 518 Zero

Rel. 5.00 Edition 03 October 2011



, GPC®, grifo®, are trade marks of grifo®

DOCUMENTATION COPYRIGHT BY **grifo®**, ALL RIGHTS RESERVED

No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, either electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written consent of **grifo®**.

IMPORTANT

Although all the information contained herein have been carefully verified, **grifo®** assumes no responsibility for errors that might appear in this document, or for damage to things or persons resulting from technical errors, omission and improper use of this manual and of the related software and hardware.

grifo® reserves the right to change the contents and form of this document, as well as the features and specification of its products at any time, without prior notice, to obtain always the best product.

For specific informations on the components mounted on the card, please refer to the Data Book of the builder or second sources.

SYMBOLS DESCRIPTION

In the manual could appear the following symbols:



Attention: Generic danger



Attention: High voltage



Attention: ESD sensitive device

Trade Marks

, GPC®, **grifo®** : are trade marks of **grifo®**.

Other Product and Company names listed, are trade marks of their respective companies.

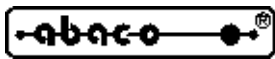
GENERAL INDEX

INTRODUCTION	1
VERSION	3
NOTES ABOUT MINI MODULE NAME	3
GENERAL INFORMATION	4
SERIAL COMMUNICATION	5
DIGITAL I/O LINES	6
I2C BUS LINE	6
ANALOG INPUTS AND A/D CONVERTER	6
MEMORY DEVICES	8
WATCH DOG	8
CLOCK	8
BOARD CONFIGURATION	9
CAN CONTROLLER	10
POSSIBLE APPLICATIONS	10
TECHNICAL FEATURES	12
GENERAL FEATURES	12
PHYSICAL FEATURES	12
ELECTRIC FEATURES	13
INSTALLATION	14
VISUAL SIGNALATIONS	14
POWER SUPPLY	14
CONNECTIONS	15
CN1 - CONNECTOR WITH MINI MODULE SIGNALS	16
RESET, POWER SUPPLY CONTROL, WATCH DOG	17
JUMPERS	18
INTERRUPTS	19
CONNECTOR SIGNALS INTERFACEMENT	20
SERIAL COMMUNICATION SELECTION	20
OPERATING MODE SELECTION	22
CAN INTERFACE	22
SUPPORT CARDS	23
USE WITH GMB HR84 MODULE	23
USE WITH GMM TST AND GMM TST 3 BOARDS	27
HOW TO START	28
A) CONNECTIONS ARRANGEMENT	28
B) TEST OF DEMO PROGRAM SAVED ON MINI MODULE	29
C) FLASH REPROGRAMMING WITH DEMO PROGRAM	29
D) GENERATE EXECUTABLE CODE OF DEMO PROGRAM	33
D3A) RECOMPILE WITH BASCOM 8051	33

D3B) RECOMPILE WITH μ C/51	35
D3C) RECOMPILE WITH LADDER WORK	38
E) FINAL PREPARATION OF APPLICATION	40
 SOFTWARE DEVELOPMENT TOOLS	 42
 PERIPHERAL DEVICES SOFTWARE DESCRIPTION	 43
CONFIGURATION INPUTS	43
VISUALIZATION LED	43
MULTIFUNCTIONS SIGNALS	44
MEMORY ACCESS	44
 BIBLIOGRAPHY	 45
 APPENDIX A: ON BOARD DEVICES DESCRIPTION	 A-1
AT89C51CC03	A-1
 APPENDIX B: GMM TST 3 ELECTRIC DIAGRAM	 B-1
 APPENDIX C: BASE CONFIG., OPTIONS, ACCESSORIES	 C-1
 APPENDIX D: ALPHABETICAL INDEX	 D-1

FIGURES INDEX

FIGURE 1: LOCATION OF PRINTED CIRCUIT VERSION	3
FIGURE 2: BLOCK DIAGRAM	7
FIGURE 3: COMPLETE VIEW	9
FIGURE 4: COMPONENTS MAP (TOP VIEW)	11
FIGURE 5: COMPONENTS MAP (BOTTOM VIEW)	11
FIGURE 6: TOP VIEW	13
FIGURE 7: LED TABLE	14
FIGURE 8: LED, JUMPERS, CONNECTOR LOCATION	15
FIGURE 9: CN1: SOCKET WITH MINI MODULE SIGNALS	16
FIGURE 10: JUMPERS TABLE	18
FIGURE 11: JUMPERS LOCATIONS	19
FIGURE 12: TTL SERIAL LINE CONNECTION EXAMPLE	21
FIGURE 13: RS 232 SERIAL LINE CONNECTION EXAMPLE	21
FIGURE 14: GMB HR84.....	24
FIGURE 15A: GMB HR84 OPEN FRAME.....	25
FIGURE 15B: GMB HR84 OPEN FRAME + GMM 518 ZERO	25
FIGURE 16: GMM TST 3 + GMM 515 ZERO COUPLE	26
FIGURE 17: RS 232 SERIAL CONNECTION BETWEEN GMM 518 ZERO AND PC	28
FIGURE 18: CD DEMO PROGRAMS.....	30
FIGURE 19: FLIP SETTINGS WINDOW (1 OF 3)	31
FIGURE 20: FLIP SETTINGS WINDOW (2 OF 3)	31
FIGURE 21: FLIP SETTINGS WINDOWS (3 OF 3)	32
FIGURE 22: LOAD SOURCE FILE WITH BASCOM 8051	34
FIGURE 23: CONFIGURE COMPILER WITH BASCOM 8051	34
FIGURE 24: COMPILE WITH BASCOM 8051.....	35
FIGURE 25: LOAD ENVIRONMENT FILE WITH μ C/51 AND JFE	36
FIGURE 26: COMPILE WITH μ C/51 AND JFE.....	36
FIGURE 27: LOAD SOURCE FILE WITH μ C/51 AND uEDIT	37
FIGURE 28: LOAD PROJECT FILE WITH μ C/51 AND UmSHELL	37
FIGURE 29: COMPILE WITH μ C/51 AND UmSHELL	38
FIGURE 30: LOAD SCHEMATIC FILE WITH LADDER WORK	38
FIGURE 31: CONFIGURE COMPILER WITH LADDER WORK	39
FIGURE 32: COMPILE WITH LADDER WORK	39
FIGURE 33: COMPILING RESULT WITH LADDER WORK	39
FIGURE 34: CONNECTION EXAMPLES	41
FIGURE C1: DEFAULT CONFIGURATION	C-1



grifo®

ITALIAN TECHNOLOGY



INTRODUCTION

The use of these devices has turned - **IN EXCLUSIVE WAY** - to specialized personnel.

This device is not a safe component as defined in directive 98-37/CE.



Pins of module are not provided with any kind of ESD protection. Many pins of the card are directly connected to their respective pins of on board's components and these last are sensitive to electrostatic noises. So personnel who handles the product/s is invited to take all necessary precautions that avoid possible damages caused by electrostatic discharges.

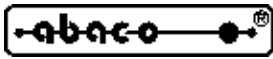
The purpose of this handbook is to give the necessary information to the cognizant and sure use of the products. They are the result of a continual and systematic elaboration of data and technical tests saved and validated from the manufacturer, related to the inside modes of certainty and quality of the information.

The reported data are destined- **IN EXCLUSIVE WAY**- to specialized users, that can interact with the devices in safety conditions for the persons, for the machine and for the environment, impersonating an elementary diagnostic of breakdowns and of malfunction conditions by performing simple functional verify operations, in the height respect of the actual safety and health norms.

The informations for the installation, the assemblage, the dismantlement, the handling, the adjustment, the reparation and the contingent accessories, devices, installation, etc. are destined - and then executable - always and in exclusive way from specialized warned and educated personnel, or directly from the **AUTHORIZED TECHNICAL ASSISTANCE**, in the height respect of the manufacturer recommendations and the actual safety and health norms.

The devices can't be used outside a box. The user must always insert the cards in a container that respect the actual safety normative. The protection of this container is not threshold to the only atmospheric agents, but specially to mechanic, electric, magnetic, etc. ones.

To be on good terms with the products, is necessary guarantee legibility and conservation of the manual, also for future references. In case of deterioration or more easily for technical updates, consult the **AUTHORIZED TECHNICAL ASSISTANCE** directly.



To prevent problems during card utilization, it is a good practice to read carefully all the information of this manual. After this reading, the user can use the general index and the alphabetical index, respectively at the beginning and at the end of the manual, to find information in a faster and more easy way.

grifo® provide this documentation "as is" without warranty of any kind. In no event shall **grifo®** be liable for indirect, special, incidental or consequential damages of any kind arising from any error in this documentation, including any loss or interruption of business, profits, use, or data. Moreover is not guaranteed the updating of the product for new computers or new operating systems, that will become available in the future.

All trademarks listed in this manual are copyright of the relative manufacturers.



VERSION

This handbook make reference to printed circuit version **250411** and following ones. The validity of the information contained in this manual is subordinated to the version number on the used card, and so the user must always verify the correct correspondence between the notations. The version number is reported in several places on the electronic part of the product, and following figure shows the most accessible one.

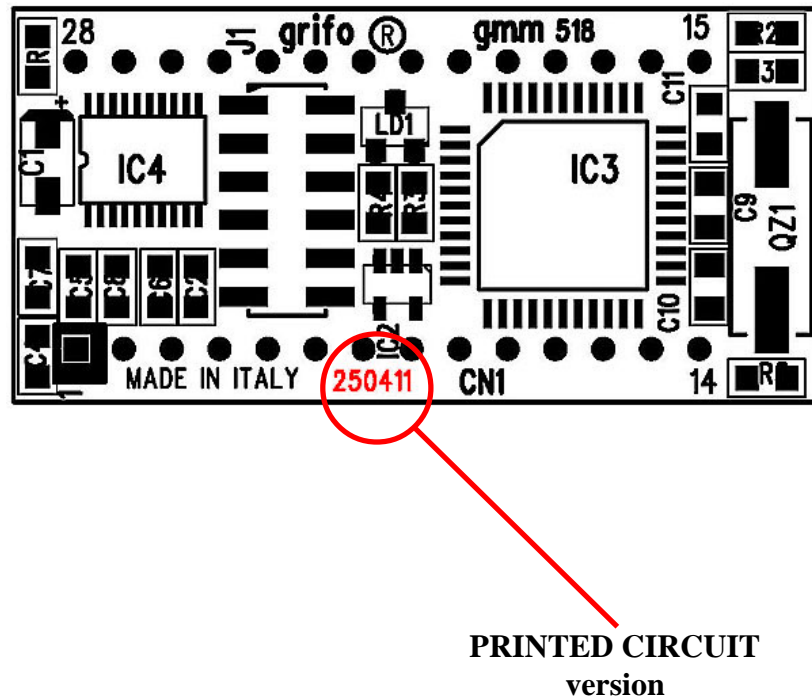


FIGURE 1: LOCATION OF PRINTED CIRCUIT VERSION

NOTES ABOUT MINI MODULE NAME

Please note that near the printed circuit revision number, just described, the card name is **GMM 518**. In fact **GMM 518 Zero** is realized by using the **GMM 518** printed circuit where it is mounted an **AT89C51CC03** microcontroller.

In order to distinguish **GMM 518** and **GMM 518 Zero** it is necessary to check the component installed on **IC3** and then use the following correspondence:

Mounted microcontroller	->	Module name
Atmel 89C51CC03	->	GMM 518 Zero
Atmel 89C51AC2	->	GMM 518

GENERAL INFORMATION

GMM 518 Zero is a **Mini Module** based on microcontroller **Atmel AT89C51CC03**, that is a powerful and complete system on chip provided with **CPU**, internal memories and a rich list of peripherals suitable for typical requirements of automation applications.

The module has already mounted inside its reduced size the components that exploit the main features of microcontroller and allows the use of all the operating modes; furthermore it includes some other components that simplify and increase the application fields.

GMM 518 Zero can be used for many installations either in the **Industrial, Domestic, Automotive** and **Didactic** environments, as described in following paragraphs.

The card use is simplified by a wide range of software development tools based either on low or high level programming languages which allow to work at the best conditions using only a standard PC. Noteworthy among these tools there are the **C, PASCAL, BASIC** compilers and a graphic programming environment based on logic contact (**LADDER**). Special care has been devoted to the application developing, by selecting tools which allow on board **FLASH** burning with user application program and **Remote debug** directly on the card, always with the simple use of a standard PC.

The **GMM 518 Zero** is equipped with a normalized standard connector that allows immediate mounting on support cards as **GMM TST 3, GMB HR84** and **GMB H844** or mounted directly on a board developed by the user, as a macro component. Both the solutions ensure a short time to market: the user can obtain a prototype or even a ready product **in one week**.

Overall features of **GMM 518 Zero** are:

- Standard format with **28** pins male socket, Dual In Line, 100 mils pitch, 600 mils width.
- **Very Small Dimension**: 20,6 x 42,6 x 15 mm.
- **4 Layers** printed circuit board to obtain best noisy resistance and best **EMI** performances.
- Single power supply required **+5 Vdc 10mA** (the current consumption may vary according to module connections and status).
- Availability of low consumption modality as **Idle** and **Power Down Modes**.
- **Atmel AT89C51CC03** microcontroller (**8051** code compatibile) with **14.74 MHz** crystal.
- Programmable machine speed at **12** or **6 clock** cycle.
- **64K FLASH** for code, **2K FLASH** for **Boot Loader**, **256 bytes IRAM** for data, **2K ERAM** for data, **2K EEPROM** for data.
- **8 A/D** converter channels with **10 bits** resolution, **20 µsec** conversion time.
- **19 interrupt** sources with **4** priority levels.
- **3 Timers Counters** up to **16** bits.
- **5 PCA** channels up to **16** bits with **PWM**, compare, capture on event, frequency generator, etc. functionalities.
- **Watch Dog** section that ensure right work of the controlled system in any operating conditions.
- **24** digital **I/O** lines available on connector. Some of these have multiple functions.
- Hardware serial line with programmable **Baud Rate** up to **115.200 Baud**, at **TTL** level or **RS 232** buffered.

- **RS 232** driver with protection against **±15 kV** discharges.
- **RERET** and power supply control circuit.
- Firmware **I2C BUS** line, available on connector.
- **CAN** controller compatible with **2.0A** and **2.0B** standards that can be connected to each line based on this protocol, through a proper external driver.
- **6** configuration **Jumpers**; **1** of them can be acquired by firmware.
- **1** status **LED** managed by software through **I/O** line.
- Internal **FLASH** and **EEPROM** can be managed through **ISP (In System Programming)**, that is with the module already mounted, by using only the serial communication line.
- **Free software** for **PC**, that supports the **ISP** programming of the generated code, inside the on board **FLASH**.
- Wide range of development tools that requires only a standard **PC**. Among the most diffused there are: **C** compilers (**μC/51**, **MCC51**, **SYS51CW**, **DDS Micro C51**); **BASIC** compiler (**BASCOM 8051**); **PASCAL** compiler (**SYS51PW**); language with **Contact Logic** (**LADDER WORK**); etc.
- Long list of demo programs and user examples supplied under source form, duly remarked, for the available development tools.

Here follows a description of the board's functional blocks, with an indication of the operations performed by each one. To easily locate such sections and verify their connections please refer to figure 2.

SERIAL COMMUNICATION

On **GMM 518 Zero** it is always available an hardware serial line that is completely software configurable for physical protocol (baud rate, stop bits number, lenght of character, etc) by simply programming some microprocessor's internal registers.

The serial line is connected to **CN1** connector at **TTL** or **RS 232** level, thanks to some on board Jumpers for configuration. When the card must be connected in a network or at long distance or with other systems that use different electric protocols, the user must provide proper external drivers (i.e. **RS 232**, **RS 422**, **RS 485**, **Current Loop**, etc.). In addition to standard receive and transmit signals on **CN1** connector there are also other **I/O** signals that can be driven by software; these signals can be used to define the **RS 485** line direction, to enable the **RS 422** transmit drive or to generate the **RS 232** hardware handshakes. For example the **MSI 01** module can convert a **TTL** serial line in any other electric standards, in a pratical and inexpensive way.

Please read **SERIAL COMMUNICATION SELECTION** paragraph of this manual or contact directly **grifo®** technicians, for further explanations or any other necessary information about electric protocols. Viceversa please read the manufacturer documentation in **APPENDIX A** of this manual and the numerous supplied examples, for detailed description of the internal registers that defines physical protocol and manages basic aspects of communication.

As described in following pages it is important remind that the module supports also other different types of serial communication, as the **I2C BUS** and the **CAN BUS**.

DIGITAL I/O LINES

The **Mini Module GMM 518 Zero** provides **24** digital **I/O** lines at **TTL** level, of the microprocessor Atmel **AT89C51CC03**, grouped in four 8 bits ports (P0, P1, P2 and P3) and in one 2 bits port (P4). A standard rules defines the names of all the signals on the Port as P0.0÷7, P1.0÷7, P2.0÷7, P3.0÷7 and P4.0÷1.

These lines are connected directly to **28** pins connector with a standard pin out, common for all the **grifo® Mini Module**, by allowing a direct connection to several interface cards.

By software it is possible to define and acquire the function, the direction and the status of these lines, and also to match them to peripheral devices (i.e. **Timer**, **Counter**, **Interrupt**, **A/D**, **PCA**, **CAN** controller, etc.), through a simple programming of some microcontroller internal registers.

For further information please refer to paragraph **CONNECTIONS** and **PERIPHERAL DEVICES SOFTWARE DESCRIPTION**.

I2C BUS LINE

On standard connector of **GMM 518 Zero** there are the signals of an **I2C BUS** interface emulated by software, and managed by two **I/O** lines of microcontroller.

This interface allows to connect the devices featuring the same communication standard in order to expand local potentialities of module.

There is wide range of software examples that manage most common devices with **I2C BUS** interface like **A/D** and **D/A** converters, display drivers, memories, temperature sensors, etc.

If you are intrested in this interfaces, it can be useful to consider the **K51-AVR** card, for which both technical manual featuring electric diagram and a complete set of examples programs are available.

In addition remind also some models of **QTP** operator panel that can be driven with an **I2C BUS** line as an alternative to asynchronous communication line.

ANALOG INPUTS AND A/D CONVERTER

Mini Module **GMM 518 Zero** provides 8 analog inputs connected to **A/D** converter section of microcontroller. These inputs can be connected to external sensors that convert many physic quantities (as temperature, pressure, speed, weight, etc.) or any other system that supplies a compatible voltage signal.

Main feautes of this section are: resolution **10** bits, **8** indipendent analog inputs, maximum accepted range **0÷3 V**, conversion time on a single channel **20 µsec**, very easy software management, end of conversion interrupt; external reference signal (**Vref**) that defines full scale voltage value.

A/D conversions are performed using the successive approximations technique and are made through opportune manipulation of specific microcontroller internal registers.

In order to simplify the **A/D** converter management, some software packages are provided with specific procedures that manage all details of this section.

For further information please refer to data sheet of **APPENDIX A** of this manual or paragraph **CONNECTIONS**.

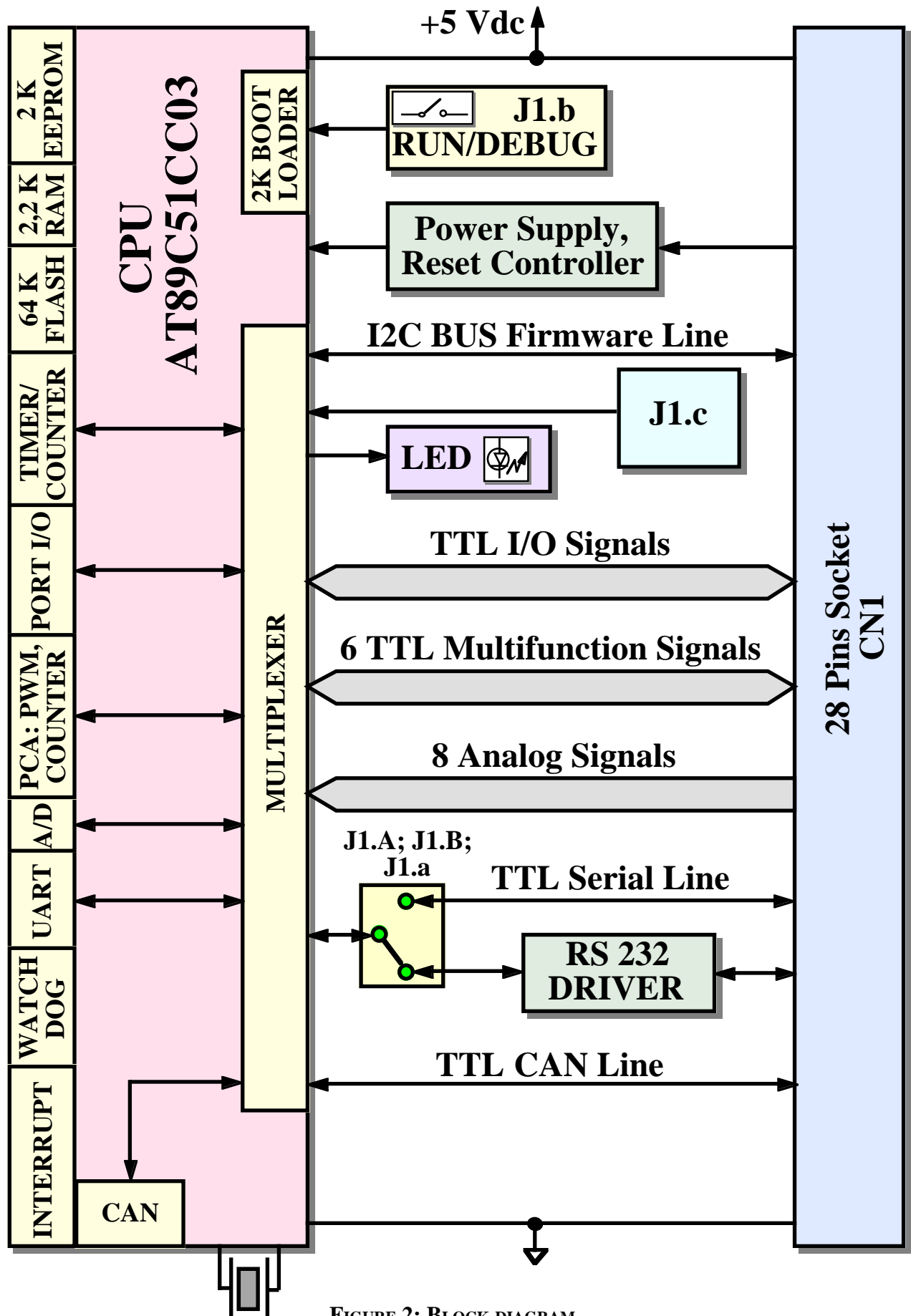


FIGURE 2: BLOCK DIAGRAM

MEMORY DEVICES

The card is provided of **70.25K** of memory divided in the following types:

- **64K** Bytes of FLASH EPROM;
- **2K** Bytes of FLASH EPROM for Boot Loader;
- **256** Bytes of internal IRAM;
- **2K** Bytes of external ERAM;
- **2K** Bytes of EEPROM.

All these memories, except the **FLASH** for **Boot Loader**, are completely available for user requirements, both for code and data of his application program.

Thanks to on board **EEPROM** there is the possibility to keep data also when power supply is failed; in this way the card is always able to maintain parameters, logged data, system status, configurations and so on, in each working conditions.

Whenever the amount of memory for data is not sufficient (i.e. for data loghin systems or special applications), it is always possible to connect external memory devices (with SRAM, EEPROM, FLASH, EPROM technologies) through the comfortable **I2C BUS** interface of the card (please see propers paragraphs).

The addressing and management of memory devices is controlled by microcontroller as described in paragraph MEMORY ACCESS and inside the component data sheet or in APPENDIX A of this manual.

WATCH DOG

Microcontroller **AT89C51CC03** includes an hardware **Watch Dog** capable to reset the **CPU** if the user program doesn't retrigger it before than the selected intervent time elapses. The main purpose of this section is to supply a real security to the system controlled by **Mini Module** in order to work correctly in any operating conditions and to avoid dangerous malfunctions not examined during develop phase.

The intervent time range of **Watch Dog** is rather wide, from a minimum of about **6** millisecond to a maximum of **1.7** second.

For further information refer to microcontroller data sheet or to APPENDIX A of this manual.

CLOCK

On **GMM 518 Zero** module there is one clock circuitry that generates the clock signal for the microcontroller. Such circuitry is based on a crystal that generates a **14.7456 MHz** frequency that is used, directly or indirectly, by all the peripherals of the module that requires timings as the asynchronous serial line, the Timers, the **Watch Dog**, the **PCA** channel, etc.

In order to improve speed performance, on **GMM 518 Zero** the user can also set the machine cycle duration to **12** or **6** clock cycles (**X2 Mode**). In X2 mode code execution is two times faster than a classic **8051** architecture.

BOARD CONFIGURATION

The **GMM 518 Zero** is provided with a Jumpers with **5** ways that allow to configure the module and its application program. Three of these dips can be acquired by software and they let the user manages different conditions through a single program, with no renounce to other input lines (the typical applications are: languages selection, definition of program parameters, operating modes selection, etc.).

Moreover one dip is used at power on or after a reset to determine which working modality, **RUN** or **DEBUG**, is used, that is respectively, whether the microcontroller has to run the user application program or the **Boot Loader**.

In addition, the board is also provided with **1** signalation **LED**; these are firmware manageable, and can be used to signal the board status and configurations in a visual ways, as described in the specific paragraphs.

All the configuration resources described are completely software manageable by simply using few specific registers of the microcontroller.

For further information refer to paragraphs DIP SWITCH, VISUAL SIGNALATIONS and VISUALIZATION LEDS.

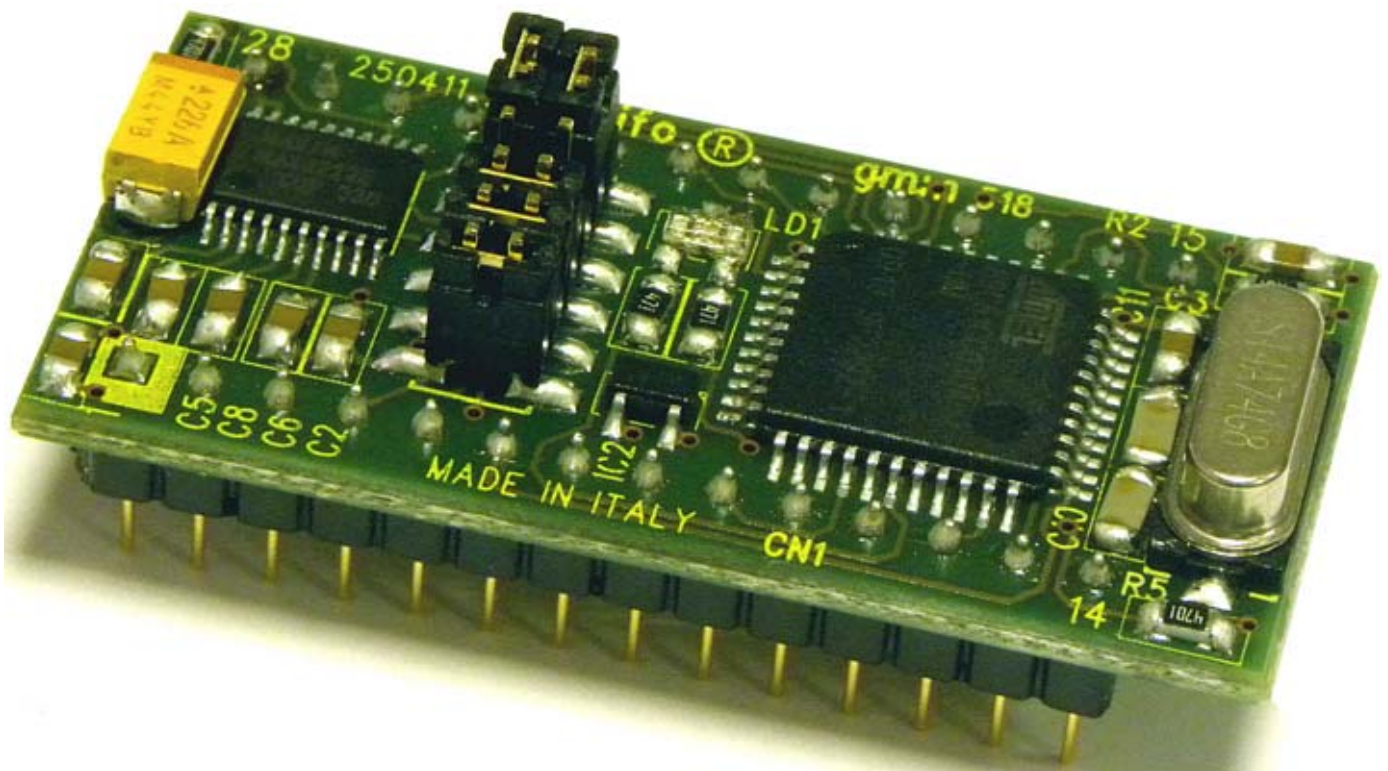


FIGURE 3: COMPLETE VIEW

CAN CONTROLLER

The **CAN** controller of **GMM 518 Zero** is charged of the complete management of the homonymous protocol in all its modalities and aspects. In details the overall features of this section are briefly described below:

- supports for **BasicCAN** protocol (**2.0 A**);
- supports for **PeliCAN** protocol (**2.0 B**);
- identifiers lengths supported: **11** and **29** bits;
- **15** independent messages managers;
- separated identifiers, masks and commands for each manager;
- programmable receive buffer size;
- programmable timer for triggers and synchronizations;
- readable errors counters;
- generates some different interrupts according with current controller status;
- completely managed through microcontroller's internal registers;
- etc.

For further information please refer to **CAN INTERFACE** paragraph and to APPENDIX A of this manual.

POSSIBLE APPLICATIONS

Possible applications of **GMM 518 Zero** modules are several. The following example can be listed:

- **Smart Intelligent Modes** with local functionalities as **PID** algorithms for Controlling Temperatures, Motors, Valves, etc.
- **Decentralized systems** as robots, automation of production line machines, big factory automations, etc.
- **Teleacquisition** and **Telecontrol** on medium and low distances.
- Application in **Home Automation** where it performs tasks like lights turning **ON/OFF**, heating and cooling systems control, supervision of electric devices, security and acces control systems, gardens sprinkle, etc.
- **Car Automations** (automotive) as lights turning **ON/OFF**, heating and cooling systems control, supervision services for drivers, anti-teft and acces control systems, functionality checks, etc.
- Also the **CAN Applications** are possible, but only when the **Mini Module** is integrated with an external line driver. So it can be used again in automotive applications, in the connection to **CAN** networks based on proprietary protocols or standard protocols as **CANopen**, **DeviceNet**, **SDS**, **CAN Kingdom J1939**, etc.
- Last but not least, the **Didactics** use in fact **GMM 518 Zero** offers a very low cost system suitable to learn microcontroller with famous **8051** core and to develop the typical start applications for the students. For this purpose it is likewise interesting the **GMM TST 3** support cards that solve the problems of power supply, of serial connection to development **PC** and of module lines connection. In the same support cards there are a matrix keyboard and a **LCD** display that allow to study and test some low cost user interface solutions.
- All the applications where it is necessary to reduce developing time and price: in fact the module can be mounted directly on the board developed by the user, as a ready to use **Macro Component**.

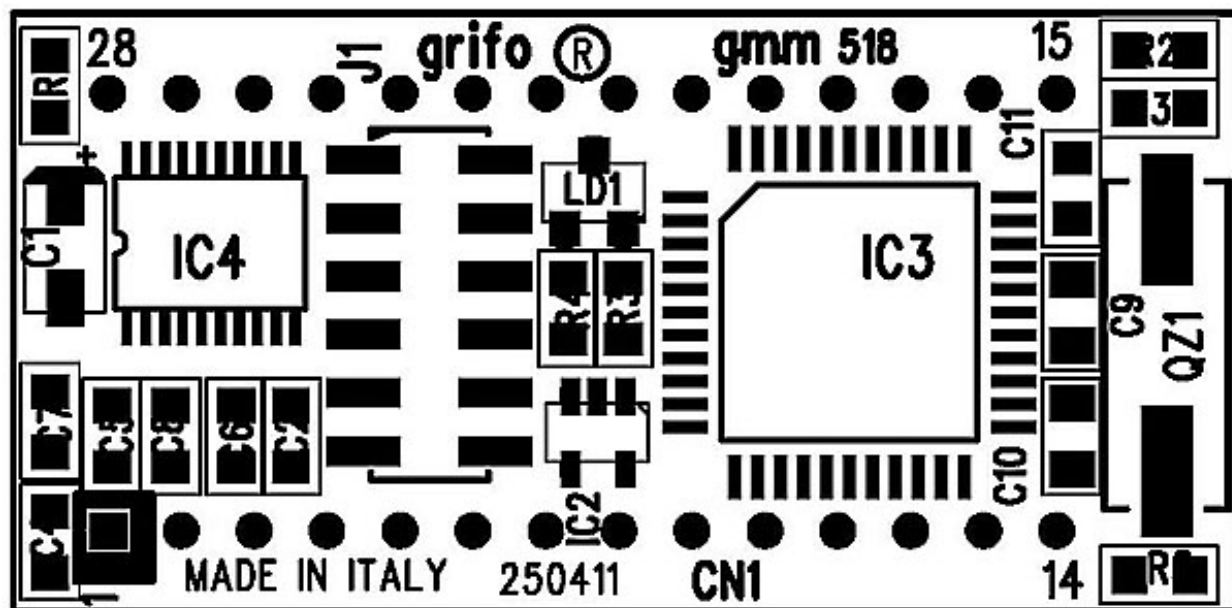


FIGURE 4: COMPONENTS MAP (TOP VIEW)

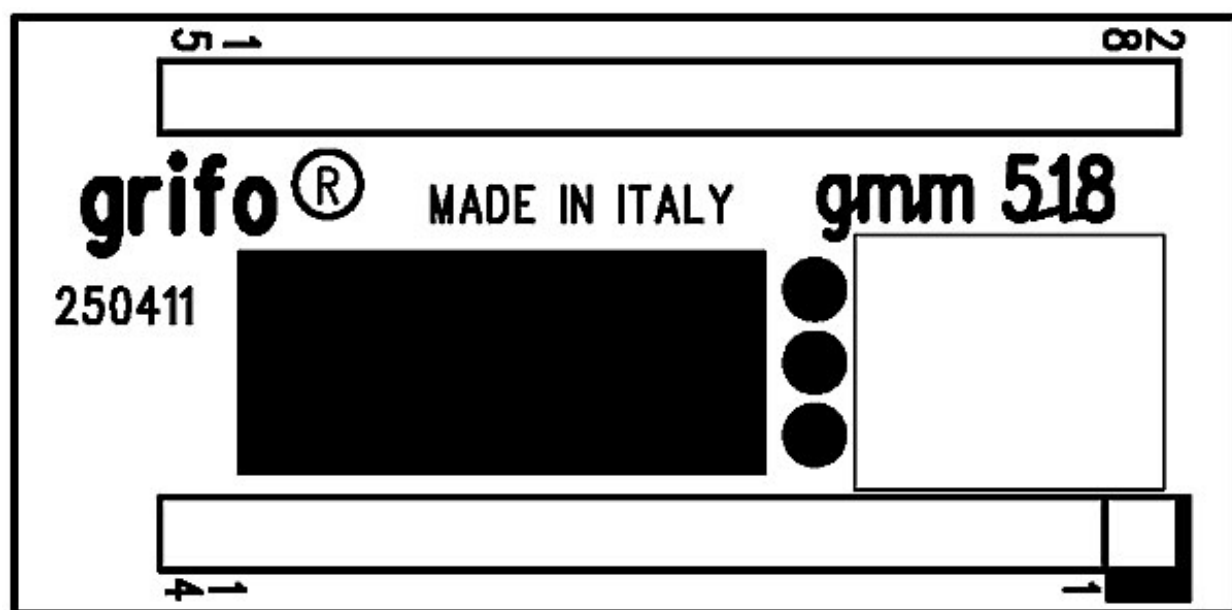


FIGURE 5: COMPONENTS MAP (BOTTOM VIEW)

TECHNICAL FEATURES

GENERAL FEATURES

Devices:	24 lines of TTL digital I/O 8 analog inputs on A/D converter 5 PCA sections for compare, capture, PWM, frequency signals 1 Watch Dog section 3 Timers, Counters multifunctions 14 interrupt sources with 4 levels 1 reset and power supply controller circuit 1 RS 232 or TTL serial line 1 CAN controller 5 configuration Jumpers 1 visualization LED	
Memories:	64K Byte FLASH	user program
	2K Byte FLASH	boot loader
	2K Byte EEPROM	user data (last 8 Bytes are reserved)
	2K Byte ERAM	user data
	256 Bytes IRAM	user data
Microcontroller:	Atmel AT89C51CC03	
Clock frequency:	14.7465 MHz	
A/D resolution:	10 bits	
A/D conversion time:	20 µsec	
Reset time:	typical 200 msec	
Watch Dog intervent time:	programmable from about 6 msec to 1.7 sec	

PHYSICAL FEATURES

Size:	20,6 x 42,6 x 15 mm	
Weight:	9 g	
Connectors:	CN1: 28 pins male socket DIL, 100 mils pitch, 600 mils width.	
Temperature range:	0÷50 °C	
Relative humidity:	20%÷90%	(without condense)

ELECTRIC FEATURES

Power supply voltage: +5 Vdc \pm 5%

Current consumption: 10 mA * (normal)
14 mA * (maximum)

Analog inputs voltage range: 0÷Vref

A/D reference voltage Vref: 0÷3 V

Analog inputs impedance: high

Power failure threshold: typical 4.65 Vdc

RS 232 protection: \pm 15 KV

(*) The reported values are referred to 20 C° environment temperature (for further information please refer to chapter POWER SUPPLY).

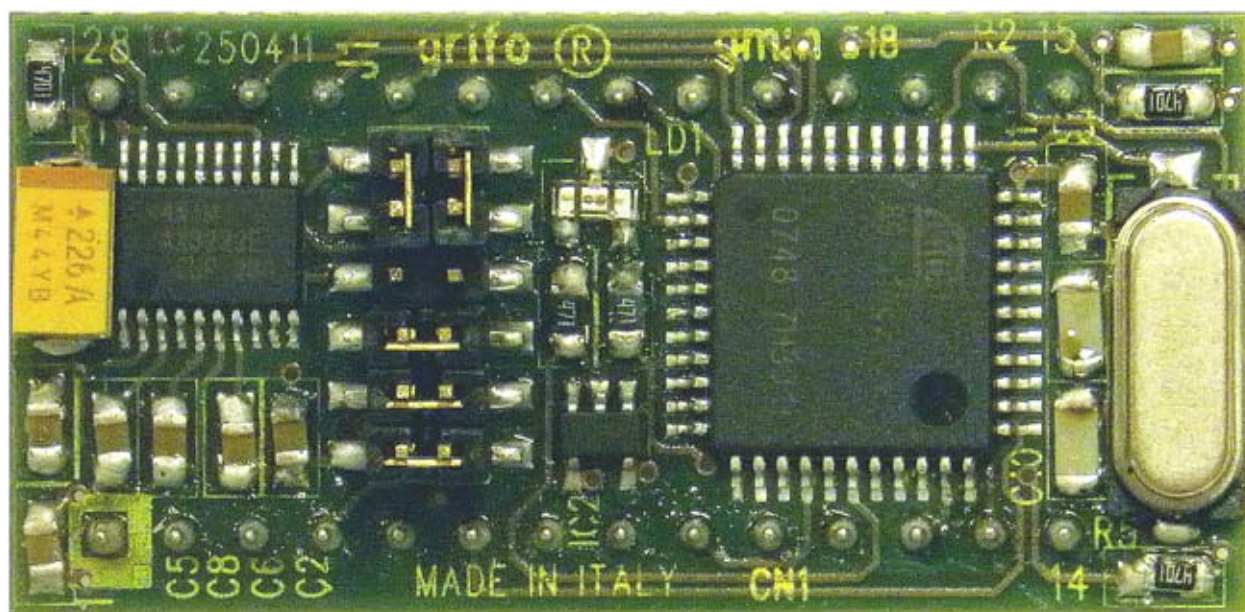


FIGURE 6: TOP VIEW

INSTALLATION

In this chapter there are the information for a right installation and correct use of the card **GMM 518 Zero**. In detail there are the locations and functions of each connector, of the user settable Jumpers, **LED**, and so on.

VISUAL SIGNALATIONS

GMM 518 Zero is provided of the **LED** described in the following table:

LED	COLOUR	FUNCTION
LD1	Red	When lighted, indicates that signal P2.7 (pin 16 of microprocessor) is at Low Level (L = Zero Volt). When no lighted, indicates that signal P2.7 (pin 16 of microprocessor) is at High Level (H = +5 Volt).

FIGURE 7: LEDs TABLE

The main function of **LED** is to inform the user about card status, with a simple visual indication and in addition to this, **LED** make easier the debug and test operations of the complete system.

POWER SUPPLY

Mini Module must be supplied with a stabilized **+5 Vdc \pm 5%** voltage connected to proper pins **14** and **28** of **CN1**.

On the board all the circuits and components have been chosen in order to obtain the best noisy immunity and the lowest consumption, including the possibility to use some different low power modalities; this feature is really important when the module is supplied by batteries for example in portable applications. In details it can be set the power down and the idle modes plus the machine cycle speed of the microcontroller, through the proper internal register **PCON**. The user application program can reduce supply consumption and eventually restore the normal working mode when a specific event occurs, like an interrupt, a variation on an analog or digital input, a timeout, etc. For further information please refer to paragraph **ELECTRIC FEATURES**.

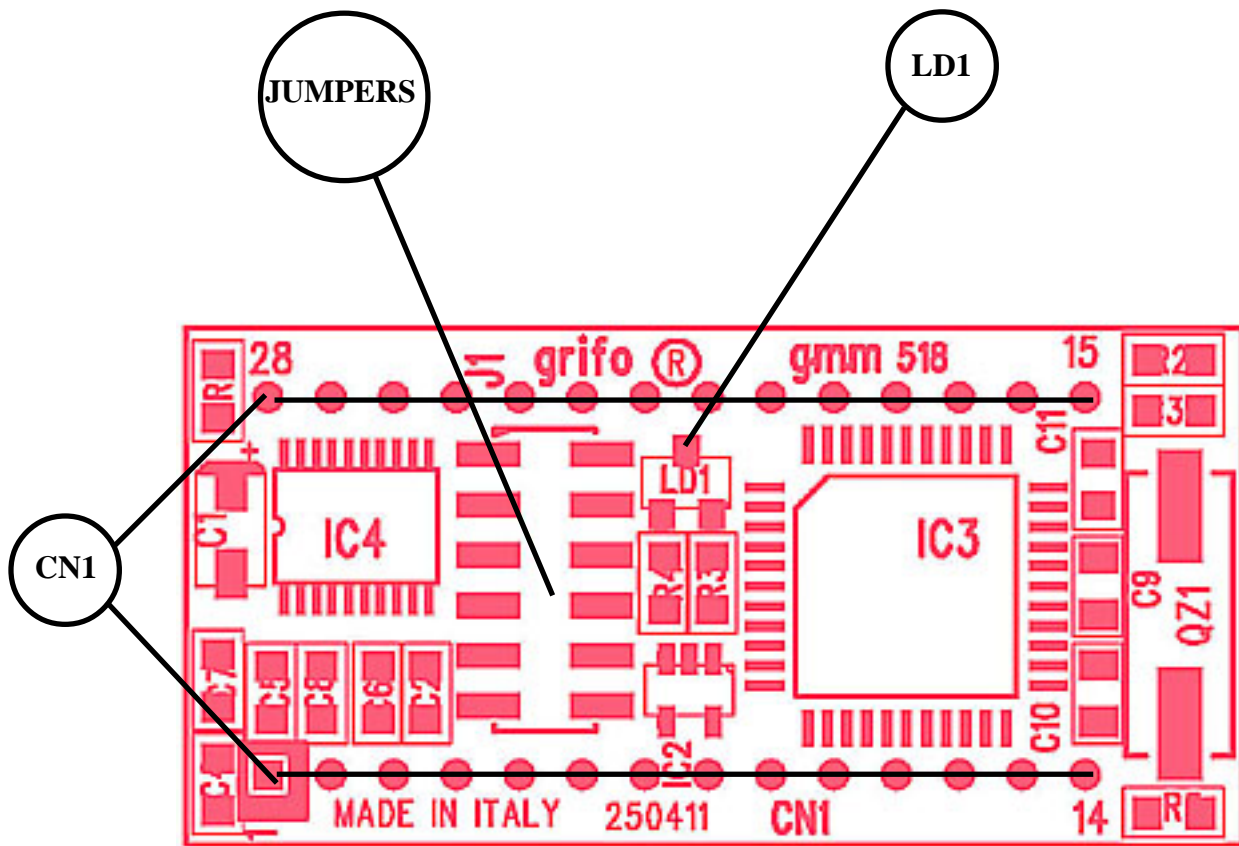


FIGURE 8: LED, JUMPERS, CONNECTOR LOCATION

CONNECTIONS

The **GMM 518 Zero** module has **1** connector that can be linkeded to other devices or directly to the field, according to system requirements.

In this paragraph there are connector pin out, a short signals description (including the signals direction) and connectors location (see figure 8) that simplify and speed the installation phase. Some additional figures shows the pins functionalities and some of the most frequently used connections.

All the connectors of **grifo®** cards follows standard pin outs in order to obtain a modular electronics where each cards can be changed with many others, of the same or different type. This reduces times and costs when modules become obsolete or insufficient for the application requirements.

CN1 - CONNECTOR WITH MINI MODULE SIGNALS

CN1 is a 28 pins, male, dual in line, socket connector with 100 mils pitch and 600 mils width.

On this connector are available all the interfacement signals of the **Mini Module** as the power supply, the I/O lines, the asynchronous and synchronous communication lines, the on board peripheral devices signals, the expansion signals, etc.

Many pins of the connector have multiple functions in fact, by software, some internal sections of microcontroller can be multiplexed with I/O signals and the following figure list all these possible functionalities. So the signals available on CN1 have different characteristics, as described in the following CONNECTOR SIGNALS INTERFACEMENT paragraph, and they follow **grifo**® Mini Module standard pin out.

In order to avoid problems on pin counting and numbers the figure 9 shows the signals directly on the top view of the **GMM 518 Zero**; moreover the serigraph reports the pins number on the four corners of the card both on bottom and top side.

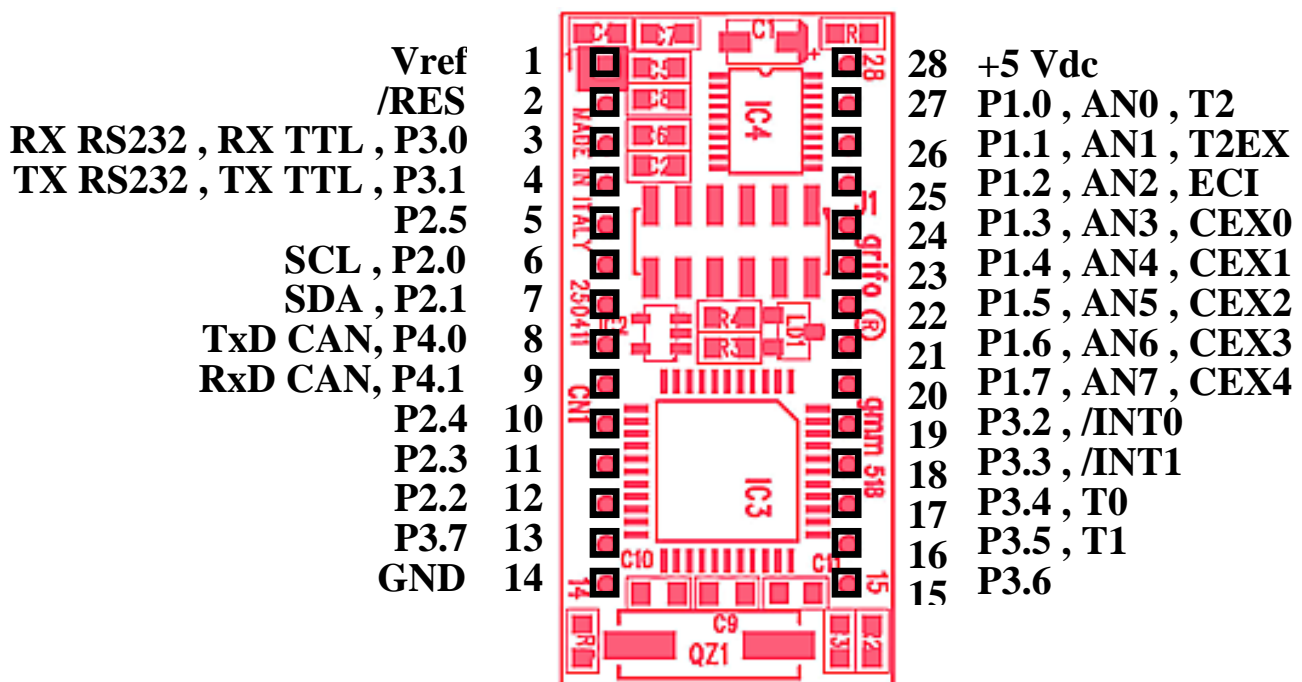


FIGURE 9: CN1: SOCKET WITH MINI MODULE SIGNALS

Signals description:

RX RS232	= I - Receive Data of serial line buffered in RS 232
TX RS232	= O - Transmit Data of serial line buffered in RS 232
RX TTL	= I - Receive Data of TTL serial line
TX TTL	= O - Transmit Data of TTL serial line
Px.n	= I/O - Signal n of Port x of microcontroller digital I/Os
/INTn	= I - Interrupt lines n of microcontroller
Tn	= I - Signal connected to section Timern of microcontroller
T2EX	= I - External trigger signal for Ttimer 2
SCL	= I/O - Clock lines of software I2C BUS interface
SDA	= I/O - Data lines of software I2C BUS interface
RxD CAN	= I - Receive Data of CAN controller

TxD CAN	= O - Transmit Data of CAN controller
Vref	= I - A/D converter reference voltage
ANn	= I - Analog inputs connected to A/D converter section
CEXn	= I/O - Signal connected to section PCAn (PWM outputs, capture and compare inputs, etc.)
ECI	= I - External clock signal of PCA sections
/RES	= I - Reset signal
+5 Vdc	= I - +5 Vdc power supply voltage
GND	= - Ground

RESET, POWER SUPPLY CONTROL, WATCH DOG

In order to prevent possible connecting problems between **GMM 518 Zero** and the external systems, the user has to respect carefully all the indications and figures of the manual and he must follow these instructions:

- For **RS 232** signals the user must follow the standard specifications of this protocol, defined by CCITT normatives.
- All **TTL** signals must follow the rules of this electric standard. The connected digital signals must be always referenced to card ground (**GND**) and then the **0V** level corresponds to logic status 0, while the **5V** level corresponds to logic status 1. The connection of these lines to devices of the controlled system (encoders, switches, proximity, electric valves, relays, motors, etc.) must be performed through proper power interfaces; it is preferable to adopt opto coupled interfaces that ensure an electric insulation between **Mini Module** electronic and external noisy, typically generated by power electronic.
- The inputs for analog section must be connected to low impedance signals in the range **0÷3.0 V**, that assure greater stability and precision. This range is valid also for the reference voltage Vref, used by **A/D** section, that must be perfectly filtered and stabilized.
- The **PWM** signals generated by **Timer**, **Counter** and **PCA** sections are **TTL** type so they must be properly buffered in order to drive the power circuitry. Typical interfaces can be a simple current driver (when a **PWM** signal is still required) or an integrator circuit, when analog voltage is necessary.
- Also **I2C BUS** signals are at **TTL** level, as defined by the same standards; for completeness it is remarked that in a network with several devices and rather long, it is better to study the connection lay out and to set properly the output stage, the operational modes and the programmable bit rate: the best values of all these settings allow right communications in any condition. Each **I2C BUS** connection must have two pull up resistors at the extremes of the cable, that is near the two units at the greatest distance, as defined by standard specification of this interface.

JUMPERS

JUMPER	POSIZIONE	UTILIZZO	DEF.
J1.A	1-2	TTL Connection - They connect the reception signal of the asincronous serial line to pin n.3 of CN1 , directly to the controller. For this operation also jumpers J1.B and J1.a. has to be connected..	*
	2-3	RS 232 Connection - They connect the reception signal (RX) of the asincronous serial line to pin n.3 of CN1 to RS 232 driver. For this operation also jumpers J1.B and J1.a. has to be connected.	
J1.B	1-2	TTL Connection - They connect the transmission signal of the asincronous serial line to pin n.4 of CN1 , directly to the controller. For this operation also jumpers J1.A and J1.a. has to be connected.	*
	2-3	RS 232 Connection - They connect the transmission signal (TX) of the asincronous serial line to pin n.4 of CN1 to RS 232 driver. For this operation also jumpers J1.A and J1.a. has to be connected.	
J1.a	Not Connected	TTL Connection - It doesn't connect the reception signal of the asincronous serial line of the controller to RS 232 driver. For this operation also jumpers J1.A and J1.B are used.	*
	Connected	RS 232 Connection - It connects the reception signal of the asincronous serial line of the controller to RS 232 driver. For this operation also jumpers J1.A and J1.B are used.	
J1.b	Not Connected	No Connection - It doesn't connect the controller / PSEN line. If Mini Module is switched on or re set in such function the RUN mode is ctivated and the programmer stored into the microcontroller FLASH starts.	*
	Connected	Connection - It connectsthe microcontroller / PSEN line to the ground (Zero Volt). If the Mini Module is switched on or re set in such function the DEBUG mode is ctivated abd the Boot Loader starts.	
J1.c	Not Connected	No Connection - It leaves the CPU pin n.24 free so it goes to +5Vdc. This condition makes 1 (H) the P0.0 microcontroller signal to logic evel acting as configuration input via software acquisition.	*
	Connected	Connection - It connect pin 24 of the CPU to the ground. This condition makes 0 (L) the P0.0 microcontroller signal to logic evel acting as configuration input via software acquisition.	

FIGURE 10: JUMPERS TABLE

On **GMM 518 Zero** module there is an 5 ways Jumpers, named J1., that defines some configurations of the card. In the following figures is reported their list and their functions in all the available connection modes.

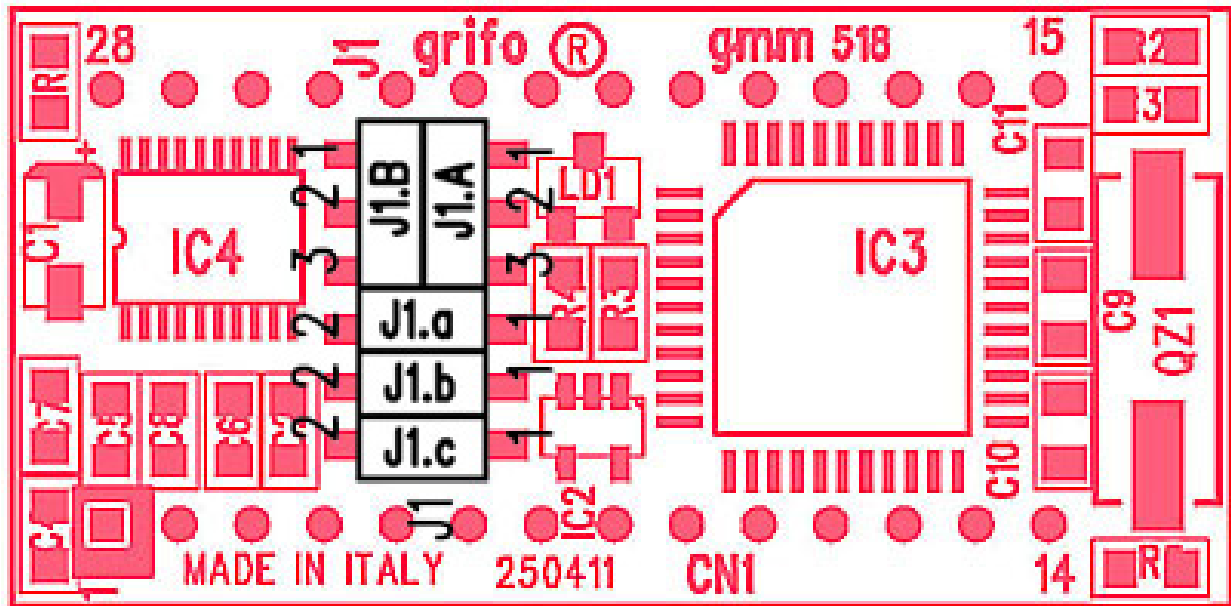


FIGURE 11: JUMPERS LOCATIONS

In the previous tables the * (asterisk) denotes the default connection, or on the other hand the connection set up at the end of testing phase, that is the configuration the user receives. The user can check the base configuration originally received, also in APPENDIX C of the manual.

In order to locate the Jumpers, please refer to figure 8 and 11.

INTERRUPTS

One of the most important **GMM 518 Zero** features is the powerful interrupts management. Below there is a short description of which devices can generate interrupts and their modalities; for further information about interrupts management please refer to the microcontroller data sheet or APPENDIX A of this manual.

- Pin 19 of CN1 -> Generates /INT0 = P3.2 on the microcontroller.
- Pin 18 of CN1 -> Generates /INT1 = P3.3 on the microcontroller.
- Peripheral devices -> Generate an internal interrupt. Possible sources of internal interrupts are the sections: Timer 0, Timer 1, Timer 2, PCA, UART, A/D converter, **CAN** controller, **CAN** timer, etc.

An interrupt management section, integrated in microcontroller, allows to enable, disable, mask and prioritize, so the user has the possibility to service promptly and efficiently any external event. The same interrupt section let the user manage the **14** interrupt sources with **4** priority levels; the addresses of the interrupt service routines can be software programmed by the user placing them on the proper code area; viceversa the interrupts priority level and activation are software programmable through specific internal registers. So the application program can always react promptly to every events, by defining also the priority of contemporaneous interrupts.

CONNECTOR SIGNALS INTERFACEMENT

In order to prevent possible connecting problems between **GMM 518 Zero** and the external systems, the user has to respect carefully all the indications and figures of the manual and he must follow these instructions:

- For **RS 232** signals the user must follow the standard specifications of this protocol, defined by **CCITT** normatives.
- All **TTL** signals must follow the rules of this electric standard. The connected digital signals must be always referenced to card ground (**GND**) and then the **0V** level corresponds to logic status **0**, while the **5V** level corresponds to logic status **1**. The connection of these lines to devices of the controlled system (encoders, switches, proximity, electric valves, relays, motors, etc.) must be performed through proper power interfaces; it is preferable to adopt opto coupled interfaces that ensure an electric insulation between **Mini Module** electronic and external noisy, typically generated by power electronic.
- The inputs for analog section must be connected to low impedance signals in the range **0÷3.0 V**, that assure greater stability and precision. This range is valid also for the reference voltage **Vref**, used by **A/D** section, that must be perfectly filtered and stabilized.
- The **PWM** signals generated by **Timer**, **Counter** and **PCA** sections are **TTL** type so they must be properly buffered in order to drive the power circuitry. Typical interfaces can be a simple current driver (when a **PWM** signal is still required) or an integrator circuit, when analog voltage is necessary.
- Also **I2C BUS** signals are at **TTL** level, as defined by the same standards; for completeness it is remarked that in a network with several devices and rather long, it is better to study the connection lay out and to set properly the output stage, the operational modes and the programmable bit rate: the best values of all these settings allow right communications in any condition. Each **I2C BUS** connection must have two pull up resistors at the extremes of the cable, that is near the two units at the greatest distance, as defined by standard specification of this interface.
- The **CAN** communication signals are at **TTL** level and they can't be directly connected to signals of the **CAN** lines, but it must be interposed a proper line driver, as described in paragraph **CAN INTERFACE**.

SERIAL COMMUNICATION SELECTION

Asynchronous serial line of **GMM 518 Zero** can be buffered in **RS 232** or **TTL**. In case it is buffered in **RS 232**, the line signals are protected against discharges up to **±15 KV**.

By software the serial line can be programmed to operate with all the standard physical protocols, in fact the bits per character, parity, stop bits and baud rates can be defined by setting proper microprocessor's internal registers.

By hardware can be selected which one of the electric protocols is used, through Jumpers configuration, as described in the previous tables; the user can select autonomously one or the other type by following the information listed below. Moreover the following figures shows how a generic external system can be connected to **GMM 518 Zero** serial line, with both the electric standards.

- SERIAL LINE IN TTL

J1.A = 1 - 2
 J1.B = 1 - 2
 J1.a = Open

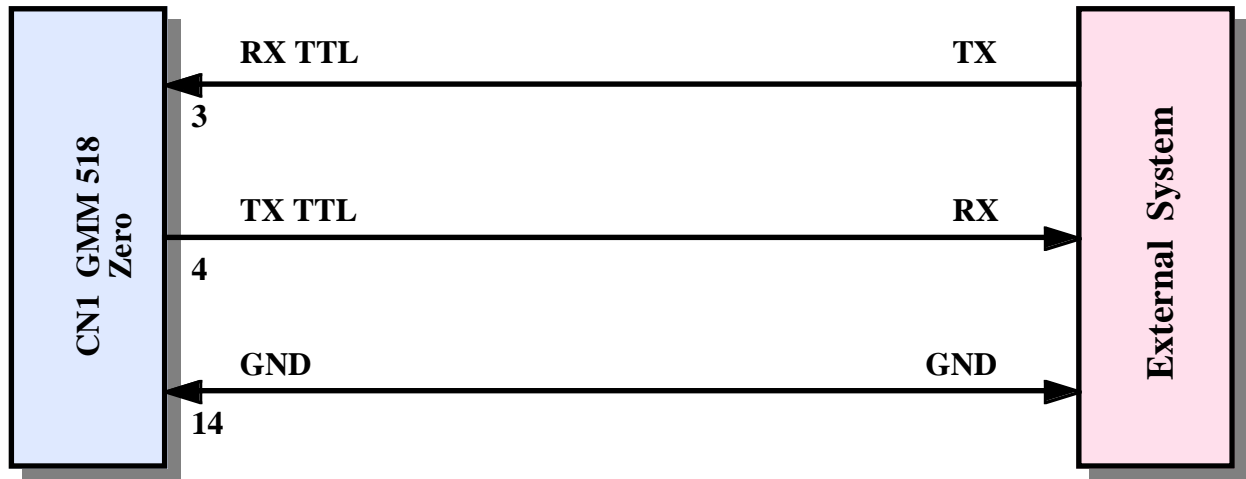


FIGURE 12: TTL SERIAL LINE CONNECTION EXAMPLE

- SERIAL LINE IN RS 232 (default configuration)

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

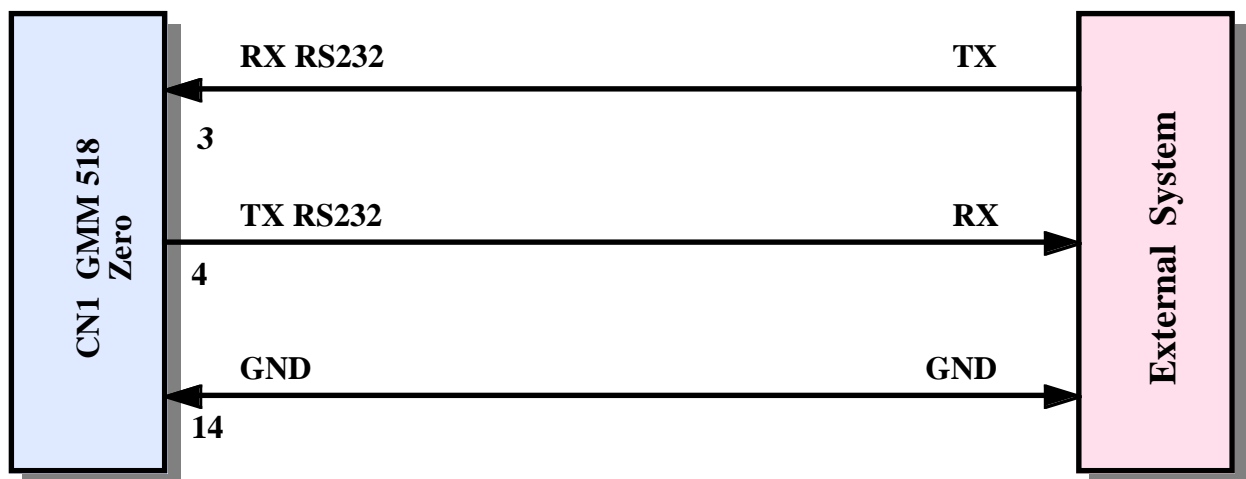


FIGURE 13: RS 232 SERIAL LINE CONNECTION EXAMPLE

OPERATING MODE SELECTION

As described on figure 10 and in next chapters, the Jumpers **J1,b** selects the operating mode of **GMM 518 Zero Mini Module**. In detail are available two modes obtained from the following configurations:

J1.b		Operating Mode
Open	->	RUN mode
Closed	->	DEBUG mode

In **RUN** mode after a reset or power on the application program saved in **FLASH** is always executed, independently by external conditions, while in **DEBUG** mode the **Boot Loader** of microcontroller is always executed.

Programs for **PC**, as the **FLIP**, communicates with Boot Loader through the serial line and allow to read, erase and write some on board memories through the **ISP** modality.

The **ISP** technique (**In System Programming**) reduces the cost and the time for development in fact it eliminates the use of external EPROM, programmer, eraser, etc. For further information about **ISP** please consult the specific technical documentation released by **ATMEL**.

CAN INTERFACE

About electric specifications of **CAN BUS**, according with **ISO-11898** standard, it is a differential line with a **60 Ω** impedance. This line is considerably different from the two **CAN** communication signals available on **CN1**, that are at **TTL** level. For this reason the connection of **GMM 518 Zero** to any **CAN** interface must be absolutely performed by inserting a proper **CAN** line driver. For detailed descriptions of this driver please read the data sheet of the selected components, that normally supplies also some application schematics (for example it can be examined the **PHILIPHS P82C250** component).

By developing a right connection of the driver both on **TTL** side (signals from controller of Mini Module) and differential side of **CAN BUS**, it can be reached the **1 Mbit/sec** speed and it can communicate with each device provided of the same interface. For example on the **CAN** line it is suggested to add two termination resistors (**120 Ω**), at the edges of the same, that ensure the right impedance; whenever the systems connected on the **CAN** network have high differences of potentials, the subsequent problems of communication and/or functionality can be solved by wiring also the grounds of the systems.

Among **grifo®** cards there is a rich list of cards provided of **CAN** interface that are ready to use, even in the **Mini Module** format: the user can examine these products before than he adds the driver to **GMM 518 Zero**.

SUPPORT CARDS

GMM 518 Zero Mini Module can be used as a macro components for some support cards either developed by the user or directly chosen from the **grifo®** boards. In the following paragraphs are described the coupling with the most interesting support cards.

USE WITH GMB HR84 MODULE

The **GMB HR84** is signalized because it is specifically designed to connect the **28 pins Mini Modules** to field signals, through proper power circuits, optocoupled, in order to be easily and fastly installed in the electric control box of the system that must be developed.

The complete description of this product is available in the relative data sheet and technical manual while in this paragraph are listed the advantages obtained by using this pair of cards and the configuration of each parts.

GMB HR84 allows easily to:

- supply power to **Mini Module** through the on board switching section that accept a wide range input voltage either **DC** or **AC**;
- connect **8 TTL** lines of the **I/O** ports to optocoupled inputs that can be indifferently **NPN** or **PNP** type. The status of all the **8** inputs is visualized through green **LED**. As the connected lines of **Mini Module** are multiplexed with internal peripheral devices it is possible to obtain high level functions as counters, combinations check, events manager, interrupts, etc.;
- generate a galvanic insulated voltage that supplies the **NPN** or **PNP** inputs, in order to connect simple and low cost external pure contacts;
- connect four **TTL** lines of the **I/O** ports to buffered relays outputs, visualized through red **LED**;
- connect the **I2C BUS**, complete of power supply, on a dedicated connector;
- connect the communication serial line through a comfortable **8 pins standard AMP MODU II 8 pins** connector;
- buffer the signals of the **TTL** serial line from **Mini Module** in **RS 422**, **RS 485** or passive **Current Loop**;
- connect the **PWM** signals through a comfortable standard **AMP MODU II 8 pins** connector;
- generate a reference voltage for **A/D** section and connect one analog input;
- provide comfortable wirings by quick release screw terminal connectors and other standard connectors;
- perform a fast mechanical mounting on **Omega Rails**.



FIGURE 14: GMB HR84

The following configurations allow to use the couple **GMB HR84 + GMM 518 Zero** in **RUN** mode, with **RS 232** serial line:

GMM 518 Zero Configuration

J1.A = 2 - 3
J1.B = 2 - 3
J1.a = Closed
J1.b = Open
J1.c = Don't Care

GMB HR84 Configuration

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

Serial connection cable with development PC = **AMP8 Cable** with female **D9** connector.

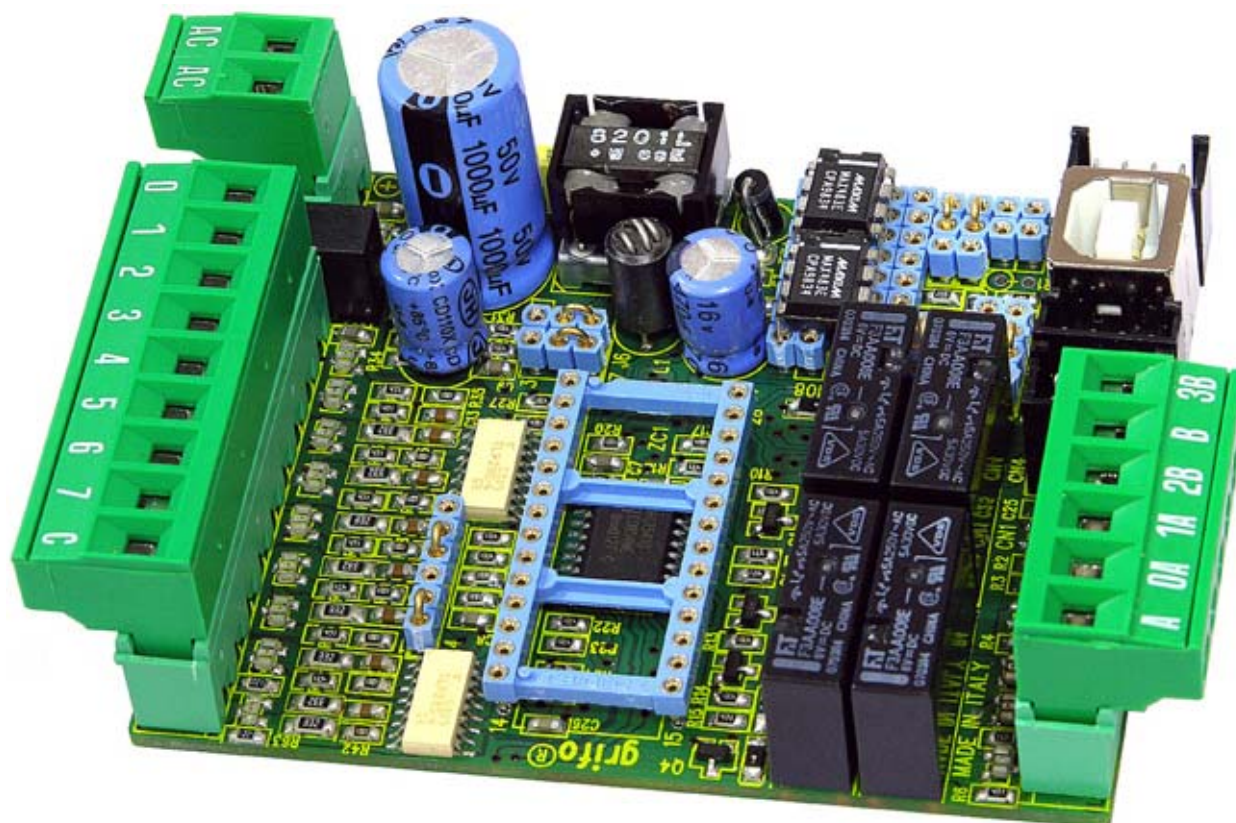


FIGURE 15A: GMB HR84 OPEN FRAME. AT SIGHT THERE IS THE 28 PIN SOCKET WHERE THE DIFFERENT MINI MODULES CAN BE MOUNTED.

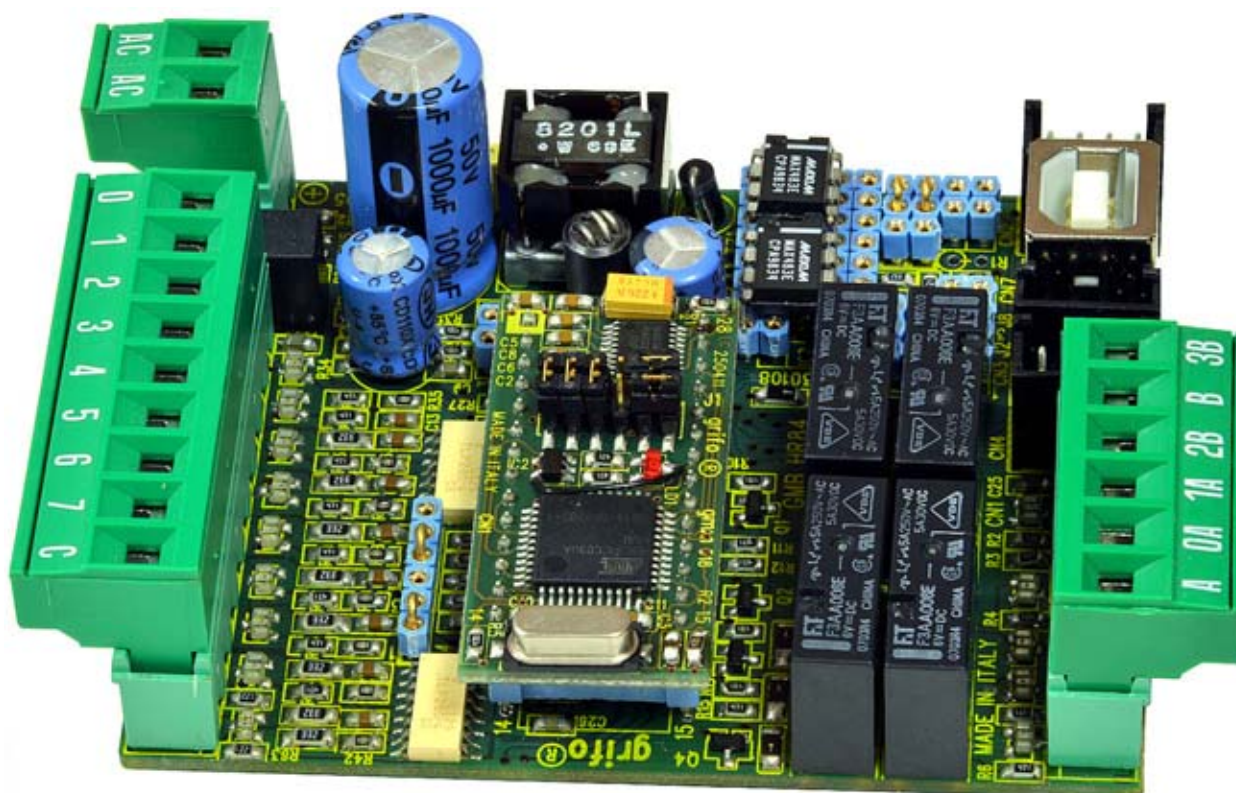


FIGURE 15B: GMB HR84 OPEN FRAME WITH 28 PIN GMM 518 ZERO MINI MODULE MOUNTED.

USE WITH GMM TST 3 BOARDS

Amongst **grifo**® cards, **GMM TST 3** es the ones designed specifically to be the evaluation boards for **GMM xxx Mini Modules**, with **28** and **40** pins.

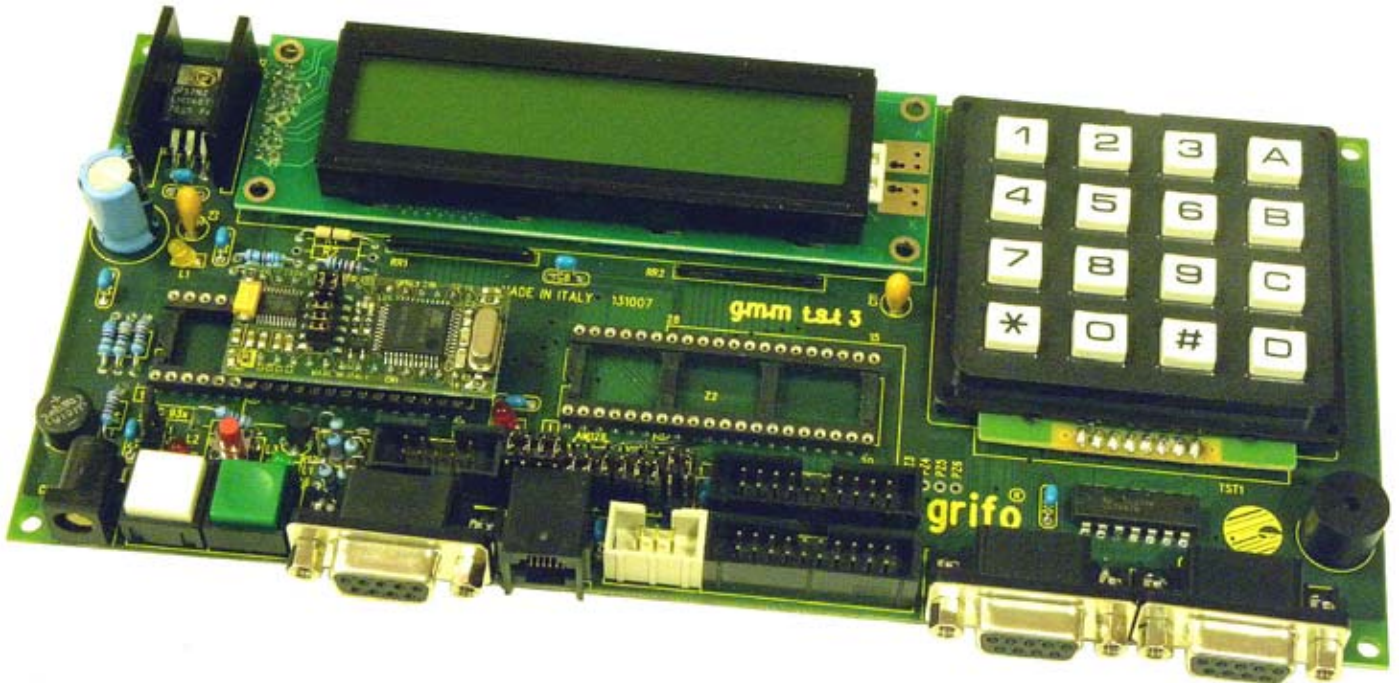


FIGURE 16: GMM TST3 + GMM 518 ZERO COUPLE

The **GMM TST 3** is an evolution of **GMM TST**, but for **GMM 518 Zero Mini Module** all them can be used indifferently. This manual will describe the latest **GMM TST 3** but everyone can easily use the other model by following the instructions of relative technical manual.

In APPENDIX B it is available also the complete electric diagram of **GMM TST 3**.

GMM TST 3 allows easily to:

- supply power to **Mini Module** through the on board linear section that accept a wide range input voltage either **DC** or **AC**;
- connect the lines of the **I/O** ports and **A/D** converter signals to comfortable low profile connectors compliant to Standard **I/O ABACO**®; this standard pin out is available on some cards that buffer the digital **I/Os** in different modes: Relays, Transistors, Optocoupled inputs, **LED**, Push Buttons, etc.;
- connect **RS 232** signals through a comfortable **D9** female type connector;
- set and visualize the status of **Mini Module I/O** signals through coloured push buttons and **LED**, disconnectable by jumpers;
- generate sound feedback using the on board **Buzzer**;

- generate a reference voltage for **A/D** section;
- restart the mounted **Mini Module** through a comfortable **Reset** push button;
- develop quickly and comfortably user interface application by taking advantage of on board **LCD** Display (backlit **2** rows x **20** characters) and matrix keyboard (**4x4=16** keys);
- develop easily a support card that satisfy customer requirements starting from the supplied electric diagram;
- support the programming of **Mini Module** through proper external **ISP** programmers (features not used with **GMM 518 Zero**);

The following configurations allow to use the couple **GMM TST 3** + **GMM 518 Zero** in **RUN** mode, with **RS 232** serial line:

GMM518 Zero Configuration

J1.A = 2 - 3
J1.B = 2 - 3
J1.a = Closed
J1.b = Open
J1.c = Don't Care

GMM TST 3 Configuration

J1 = 2 - 3
J2 = Open
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

Serial connection cable with development **PC = CCR 9+9E** (that is an extension cable provided of **D9 Female** and **D9 Male** connectors).

In corrispondence of the first purchase included in the received materials (**CD**) there are numerous demo programs that allow to use immediately, both the resources of **Mini Module** plus support boards.

HOW TO START

In this chapter are listed the operations that must be performed to start using the **GMM 518 Zero** in a practical and fast way, solving the typical beginners problems.

The chapter includes some common sections and other parts that are different according with used development environment; all these paragraphs contains interesting information even for the users that already know the product, in fact there is the description of a fast and complete functional test. Moreover it is supposed that the user already have the accessories (power supplier and serial connection cable) and a **PC** provided of the necessary features in order to execute the programs described in the following points. This **PC** is identified with the name development PC and its minimum requirements are listed in the documentation of the used programs.

A) CONNECTIONS ARRANGEMENT

- A1) Perform the serial connection between Mini Module **GMM 518 Zero** and development PC or on the other hand connect the two communication signals (TX RS232, RX RS232) and the reference ground signal (**GND**), to a free **COMx** serial port of **PC**. Obviously this connection changes according with possible support card used but generally it is described by following figure.

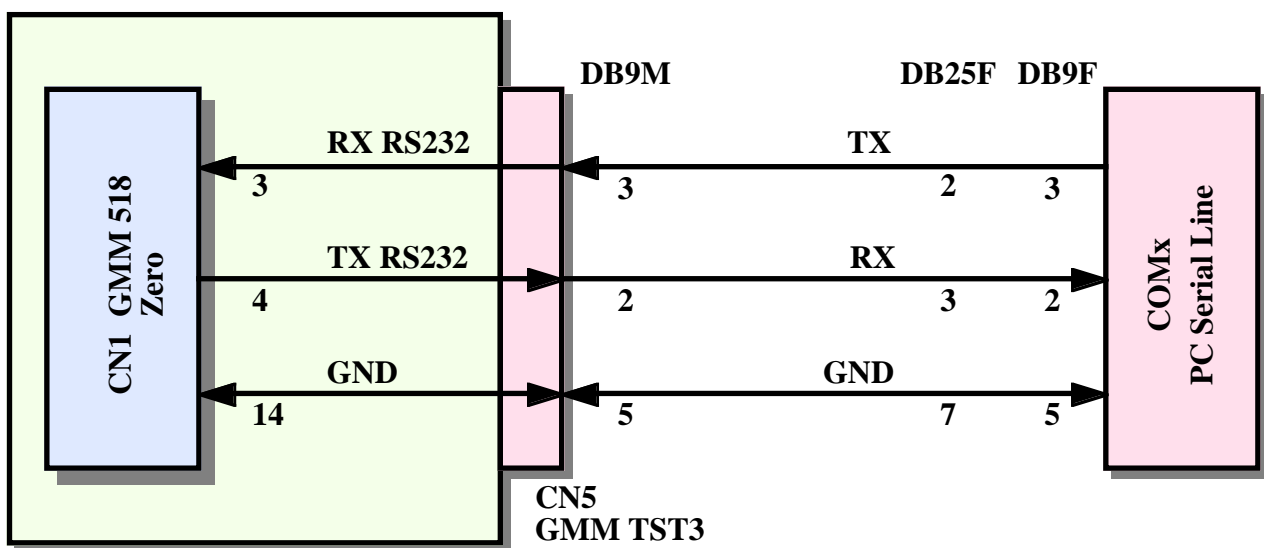


FIGURE 17: RS 232 SERIAL CONNECTION BETWEEN GMM 518 ZERO AND PC

On figure 17 it can be easily discovered that the connection cable with **GMM TST3 + GMM 518 Zero** is a normal serial extension cable (as those used for example for **RS 232** modems) and it can be also ordered to **grifo®** with the code **CCR 9+9E**.

- A2) Provide a suitable power supply: when the **Mini Module** is used alone this voltage must be applied as described in **POWER SUPPLY** paragraph, for example by using a laboratory bench model. Viceversa when a couple is used it can be used many different power sources, as described in technical manual of the support card where **Mini Module** is installed.

B) TEST OF DEMO PROGRAM SAVED ON MINI MODULE

- B1) Found the **HYPERTERMINAL** communication program on the development **PC**, that normally is located on Windows menu: *Start / Program / Accessories / Communication*, execute it and through the properties windows, setup the communication parameters to:

<i>Connect</i>	<i>directly to COM x (those used at point A1)</i>
<i>Bit rate</i>	<i>19200</i>
<i>Data Bits</i>	<i>8</i>
<i>Parity</i>	<i>No</i>
<i>Stop Bit</i>	<i>1</i>
<i>Flow control</i>	<i>None</i>

As an alternative it can be used any other serial communication program and also the terminal emulation integrated in **BASCOM 8051**.

- B2) Select **RUN** mode, that is Jumper **J1.b** of **GMM 518 Zero** in Open position..
- B3) Supply power voltage to **Mini Module**, or reset it, and then check that its **LED** continuously blink; contemporaneously on development **PC** monitor appears the demo program presentation message. Each **GMM 518 Zero**, received for the first time, is delivered with its **Demo** program already saved in **FLASH** and arranged to start automatically at power on, with the described functionalities.
If these results are not obtained, please check again the power supply, the serial connection and the **J1.b** configurations.
- B4) Follow the instructions of the demo, test all the resources of **Mini Module** and verify the obtained effects directly on hardware: the user can interact with demo by using the development **PC** as a console.
- B5) When demo execution is completed turn off **GMM 518 Zero** power supply.
- B6) Exit from **HYPERTERMINAL** program on development **PC**.

C) FLASH REPROGRAMMING WITH DEMO PROGRAM

- C1) On the received **grifo®** disk find and then install the utility program **FLIP** on a comfortable folder of development **PC** hard disk. **FLIP** manages the **ISP** programming of **FLASH EPROM** on **GMM 518 Zero** and it interacts with the **Boot Loader** executed by the same module.
You can freely download and install the latest version of **FLIP** directly from **Atmel** web site: www.atmel.com.
- C2) Create a new folder on hard disk of development **PC** that will be the place where the user saves all his files; up to now this folder is named working folder.

- C3) On the received **grifo®** CD find the demo program of **GMM518 Zero**: this file has the name visualized in the start up phase, at point B3, and it can be reached from the start page of **CD** by following the path: *English / Examples Tables / Mini Block and Mini Modules programs*.
- C4) Copy all the files found at previous point to working folder created at point C2. At the end, please ensure that all the files copied on hard disk have the *Read only* attribute disabled.



FIGURE 18: EXAMPLE PROGRAMS CD FOR 8051

- C5) Select **DEBUG** mode, that is Jumper **J1.b** of **GMM 518 Zero** in Close position.
- C6) Turn off and then turn on the **Mini Module** or reset it.

- C7) Run the **FLIP** program, installed at point C1. It can be used the *Start / Programs / ATMEL / FLIP x.y.z / FLIP* link, when the default setting have been used during installation of the same program.
- C8) Select the device to program by clicking the first button on the top left, picking the *AT89C51CC03* in the *Device Selection* window and then press *OK*:

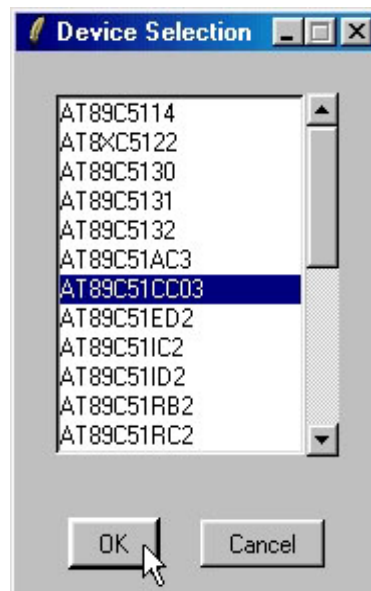


FIGURE 19: FLIP SETTINGS WINDOW (1 OF 3)

- C9) Select the communication mode for **ISP** programming by clicking the second button on the top left, picking in sequence: *RS 232*, the serial *Port* of development PC, *115.200 Baud* and then press *Connect*. At this point the **FLIP** starts communication with microcontroller Boot Loader and fill in a list of data in its main window. If communication fails and after about **20** seconds the window *Timeout error* appears, try to reduce communication speed, from **115.200** to **19.200** Baud, and repeat previous points.

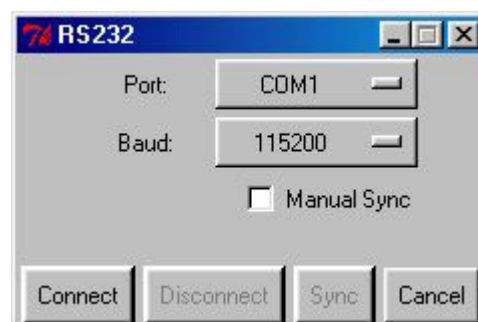


FIGURE 20: FLIP SETTINGS WINDOW (2 OF 3)

- C10) Load the file to write in **FLASH** (that is the one with **.HEX** extension copied at point C4) by clicking the third button on top right side and then select the file by using the displayed dialog box. In the frame *FLASH Buffer Information* several information about the file just loaded appear; in detail the box *HEX File:* must report the file name.

- C11) Select all the check boxes in the frame *Operations Flow* as in figure 21, in order to let **FLIP** execute in sequence the four operations: erase, blank check, program and verify.
- C12) At this point make sure that main windows of **FLIP** looks like figure 21; in details the data in the *Size*, *BLJB*, *X2*, *Device SSB* and *BSB / EB / SBV* boxes must exactly match.

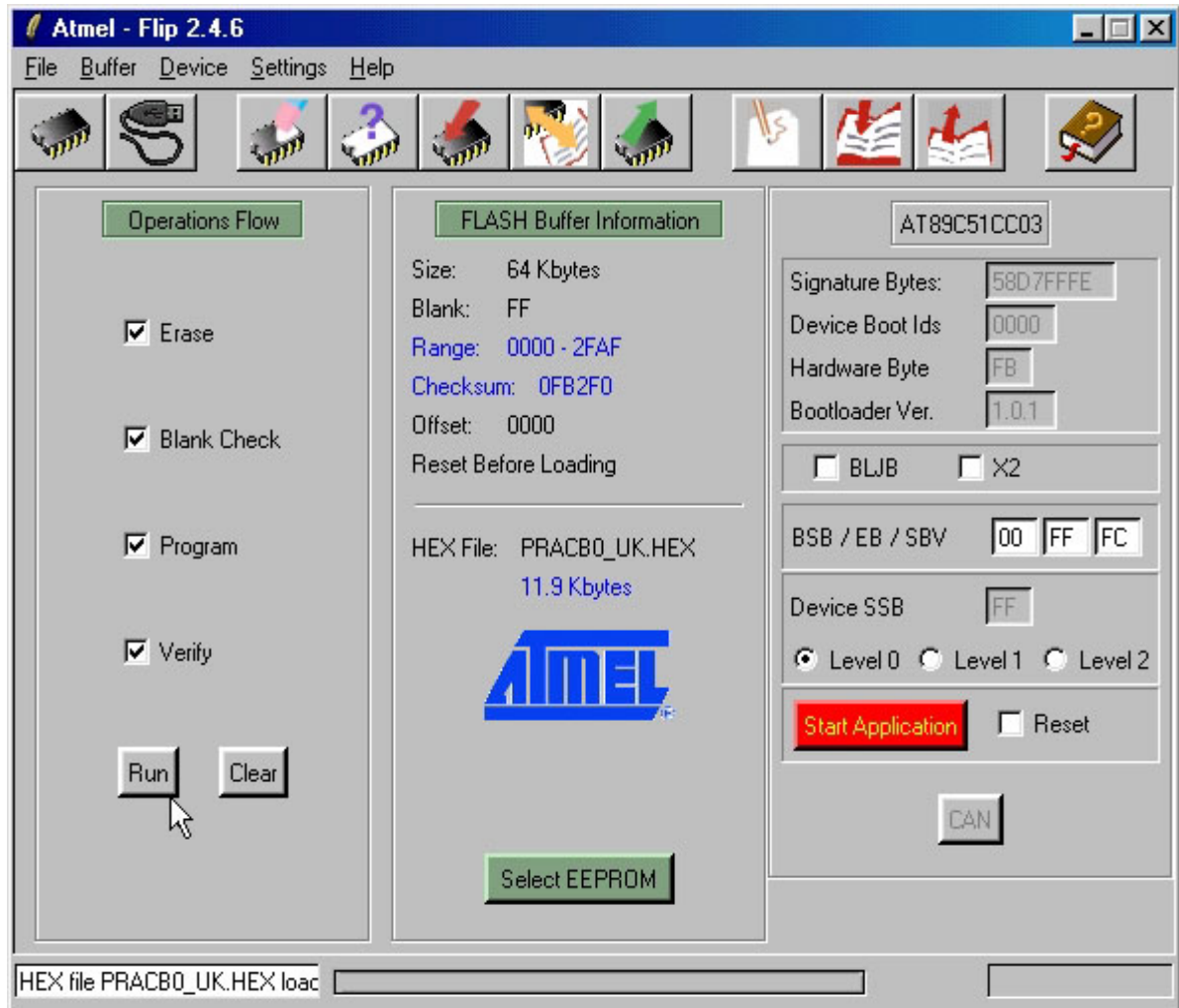


FIGURE 21: FLIP SETTINGS WINDOWS (3 OF 3)

- C13) Press button *Run* in the main window to start the preselected ISP operations. All this operations will program the on board **FLASH** with the loaded code.
- C14) Wait the execution end of **ISP** operation . The status bar on the bottom reports operation progress and near text box, on the left, reports operation status; the check boxes become red and then green when the respective operation is successfully completed. Thus wait for *Verify* check box to become green.
- C15) At this point the **FLASH** is programmed and **FLIP** can be closed.
- C16) Retest the program just saved by repeating the steps B1÷B6.

The steps just described must be repeated for any test of the application program and for this reason it could be preferable to speed up their execution, by using the **BatchISP** modality of **FLIP**. This automatically performs the instructions of a proper command file, as the following one:

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

For further information on **ISP** programming and **FLIP** use, please consult the specific technical documentation released by **ATMEL**.

D) GENERATE EXECUTABLE CODE OF DEMO PROGRAM

- D1) Install the software environment selected to develop the application program on the hard disk of the development **PC**. As described in the chapter **DEVELOPMENT TOOLS** there are many different software packages that satisfy any customers requirements but here we remind only the most diffused as the **BASCOM 8051**, **µC/51** and **LADDER WORK**. For detailed information on this products please refer to relative user manuals or the proper on line helps.
- D2) Check that at point C4 from **grifo® CD** had been copied all the files of the demo program, not only the one with executable code. In detail in the working folder must be available the source file, the project file, the declaration files, provided of the following extensions according with the used development tools
- | Development tools | Language | Source | Project | Declaration | Environment |
|--------------------|----------|--------|---------|-------------|-------------|
| BASCOM 8051 | BASIC | .BAS | - | .DAT | - |
| µC/51 | C | .C | .MAK | .H | .WSP |
| LADDER WORK | LADDER | .PJN | .PJN | - | - |
- D3) Compile the source file by using the selected software tools in order to obtain the **.HEX** file identic to those received and already used, at point C. This operation is really different according with the selected development tools, so here follows the detailed steps properly divided and organized:

D3A) RECOMPILE WITH BASCOM 8051

- D3A1) Copy the declaration file **.DAT** described at point D2 inside the installation folder of **BASCOM 8051**: `\MCS Electronics\BASCOM8051\`.

- D3A2) Execute the **BASCOM8051** and when his IDE is opened, load the source file (with **.BAS** extension), through the menu *File / Open*:

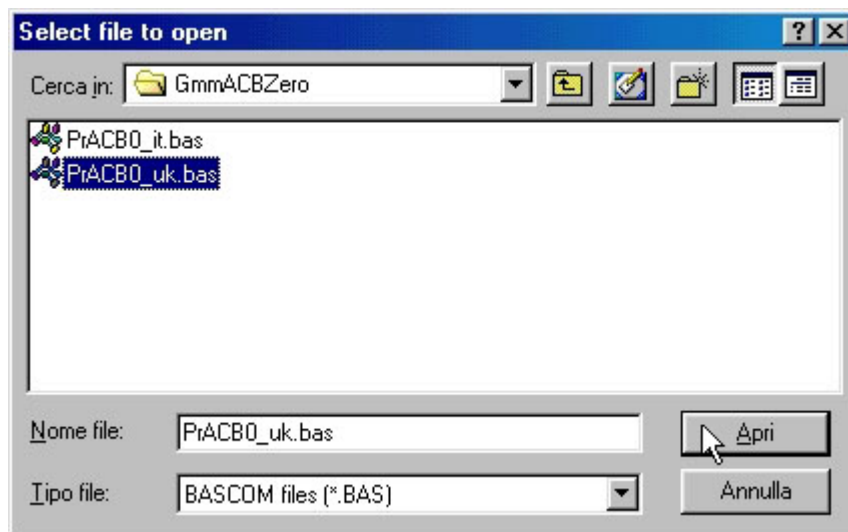


FIGURE 22: LOAD SOURCE FILE WITH **BASCOM 8051**

- D3A3) Open the configuration window of **BASCOM8051** compiler, by selecting the command *Option / Compiler / Misc*, then define the settings described in the following figure and finally confirm with *Ok* button.

The setting for *Register file* field must match the used microcontroller and it can be selected only when the declaration files have been correctly copied, as described at point D3A1.

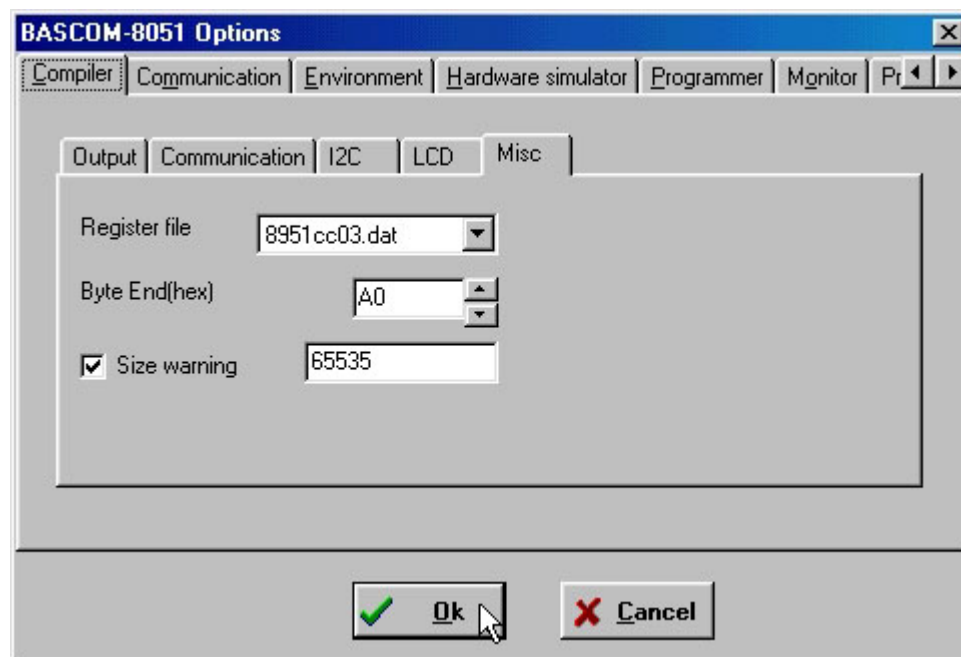


FIGURE 23: CONFIGURE COMPILER WITH **BASCOM 8051**

- D3A4) Compile the source file of demo program with the simple pressure of the hot key *F7*, or by selecting the command *Program / Compile*, and verify that no errors happens. An *.HEX* file must be obtained equal to those available on **grifo®** disk and already used at points C.

The compiling time change according to used development PC; anyway the user must wait that both the passes are completed, through a specific status window displayed during compilation, and then check that the bottom side of **IDE** doesn't show errors. In other words at the end of compilation it must be displayed a window similar to those reported in following figure.

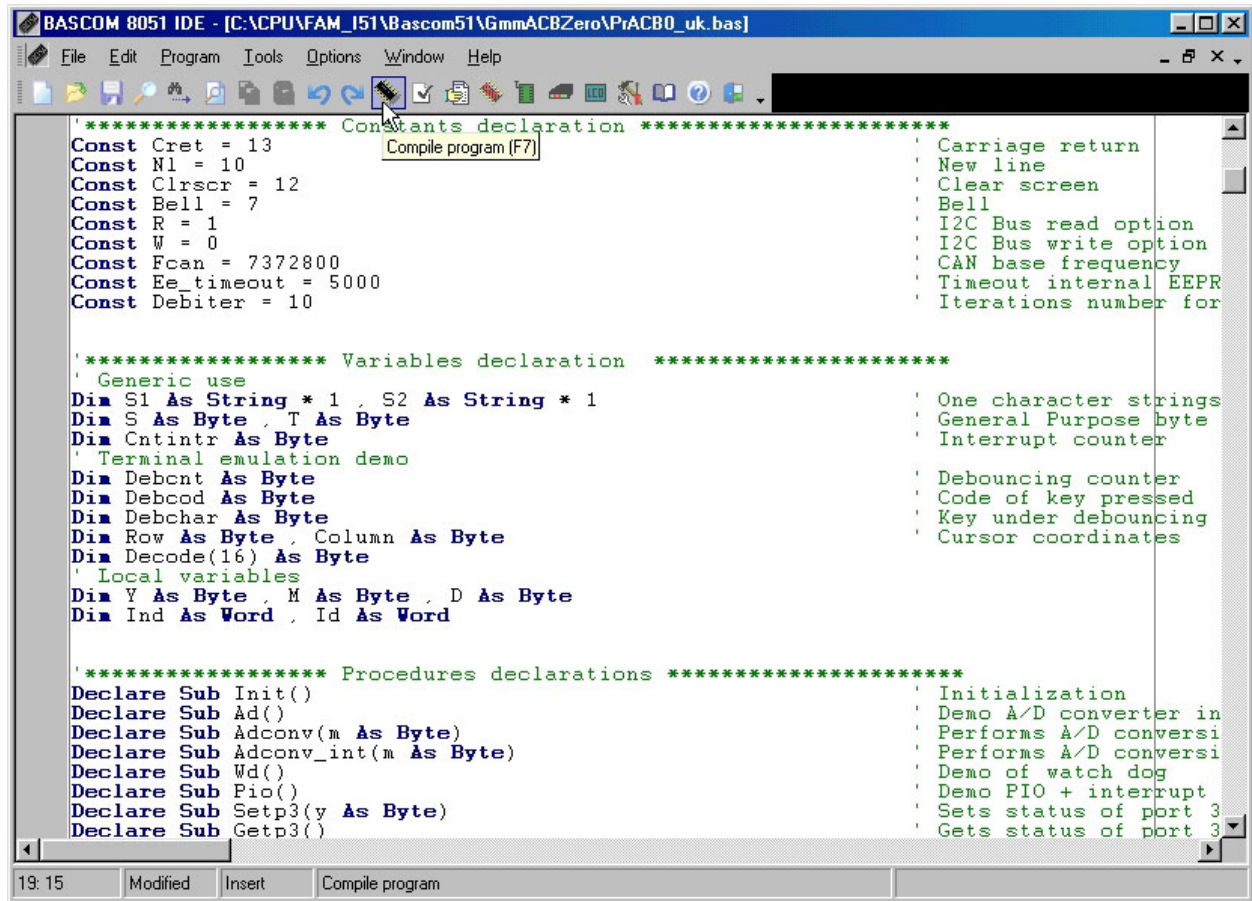


FIGURE 24: COMPILE WITH BASCOM 8051

D3B) RECOMPILE WITH μ C/51

D3B1) The μ C/51 software package includes two separated development tools. The first is the **JFE** editor that is a complete **IDE** provided of multifiles editor, tree garbages manager and an executer of external programs. This can be configured by proper programs in order to obtain a complete environment capable to perform all the operations required by application program development.

The second includes an editor named **uEdit** useful to examine and change the source and a project manager, named **UmShell**, that can compile the source and obtain the executable code for **GMM 518 Zero**.

The following steps list the basic operations required by each tools of μ C/51, while for detailed information about all the described programs, please read the complete documentation included in the software package.

- D3B2) By using **JFE** editor, first of all run the homonymous program from Windows start menu and then open the environment file of the demo program, with **.WSP** extension, through the menu *File / Open Workspace*. Whenever this file is not available, the user can generate it or alternatively he can use the uEdit and UmShell programs described later.

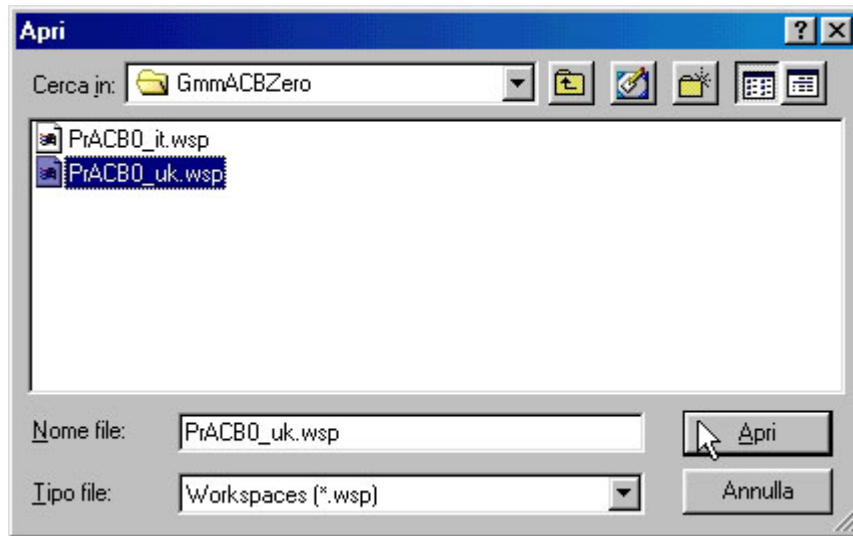


FIGURE 25: LOAD ENVIRONMENT FILE WITH μ C/51 AND JFE

At this point everything is ready to use and in order to create the executable code of the program it is sufficient press the **RE-MAKE** button on the command bar of **JFE**. Then, when compilation stops, verify that no errors have been found, as visualized in the following figure.

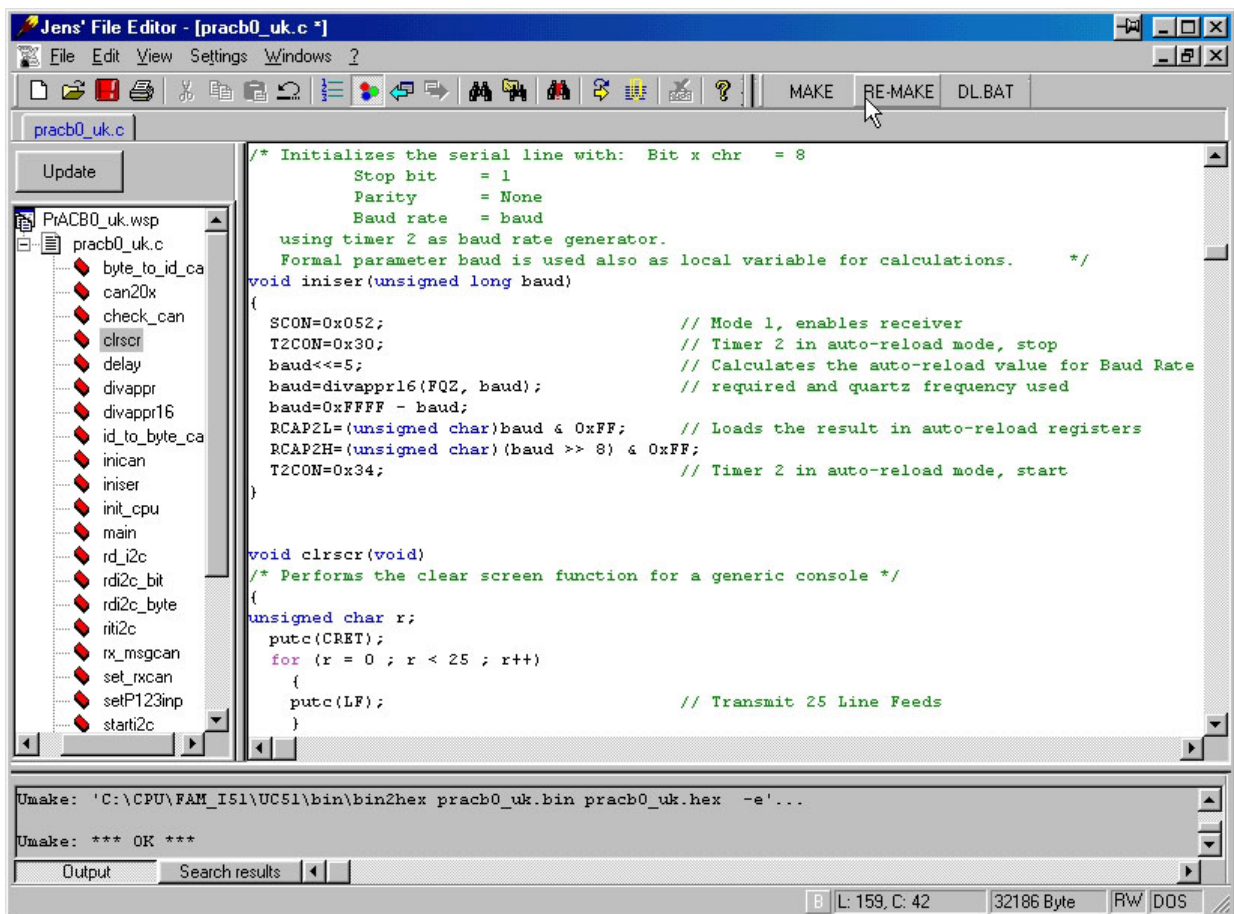


FIGURE 26: COMPILE WITH μ C/51 AND JFE

At this point an *.HEX* file must be obtained equal to those available on **grifo®** disk and already used at points C.

- D3B3) By using **uEdit** and **UmShell**, first of all run the first program from Windows start menu and then open the sourcefile of the demo program, with *.C* extension, through the menu *File / Open*

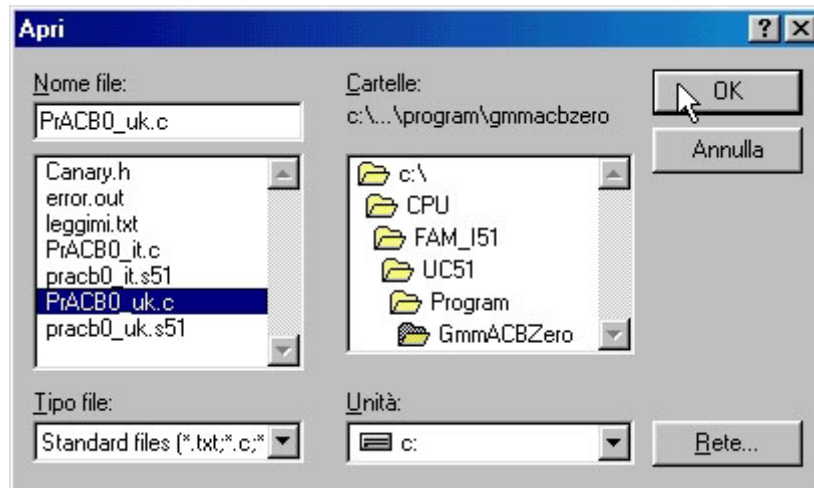


FIGURE 27: LOAD SOURCE FILE WITH μ C/51 AND uEDIT

- D3B3) Going on with **uEdit** and **UmShell**, then run the second program from Windows start menu and then open the project file of the demo program, with *.MAK* extension, through the menu *File / Open*

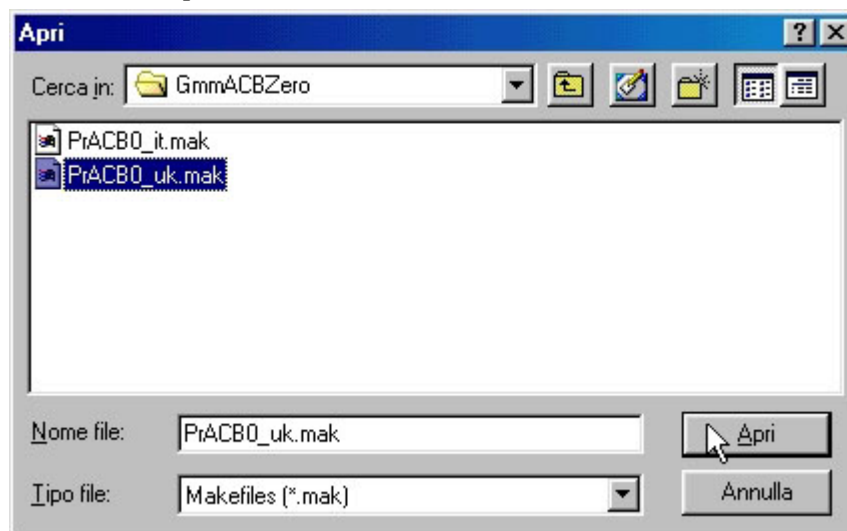


FIGURE 28: LOAD PROJECT FILE WITH μ C/51 AND uMSHELL

The project file defines all the compiler modalities of the source and so it decides the features of the generated executable code, especially for data and code areas allocation, optimizations, console management, etc.

- D3B5) Still using **uEdit** and **UmShell**, now compile the source by pressing the second button from the right of UmShell, or alternatively through its menu *Make / Remake Target*. During the compile phase it must be checked that no errors are found and at the end an *.HEX* file must be obtained equal to those available on **grifo®** disk and already used at points C

The compiling time change according to used development **PC**; anyway the user must wait that all the phases are completed, as properly visualized by **UmShell** window, by obtaining a condition similar to following figure.

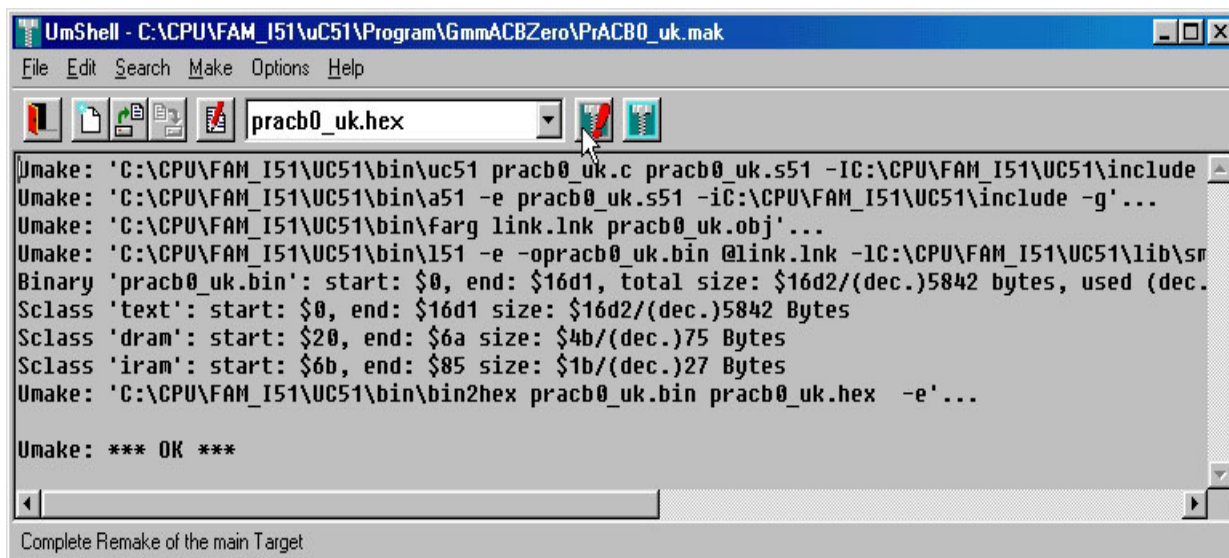


FIGURE 29: COMPILE WITH μ C/51 AND UMSHELL

D3C) RECOMPILE WITH LADDER WORK

- D3C1) First of all execute the **LADDER WORK** from Windows start menu and, when his **IDE** is opened, load the schematic of demo program (with **.PJN** extension) through the menu *File / Open*:

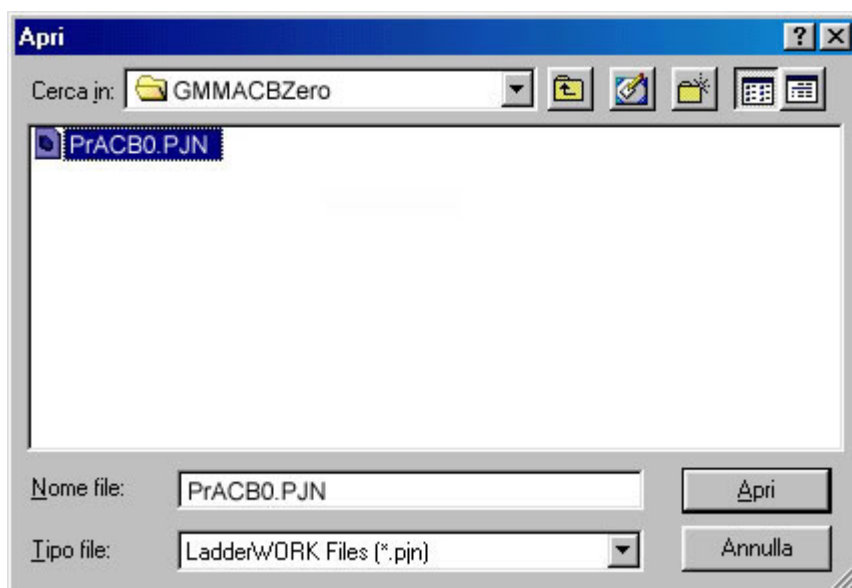


FIGURE 30: LOAD SCHEMATIC FILE WITH LADDER WORK

- D3C2) Ensure that the selected profile is the right one specific for **Mini Module GMM 518 Zero**:



FIGURE 31: CONFIGURE COMPILER WITH LADDER WORK

- D3C3) Compile the source schematic pressing the first button from the right:

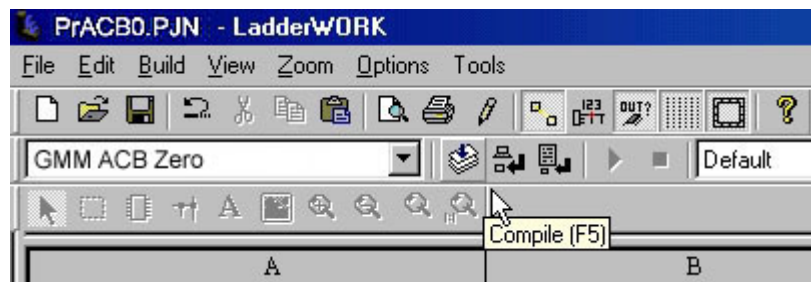


FIGURE 32: COMPILE WITH LADDER WORK

- D3C4) During the compile phase no errors must be visualized as in the following figure:

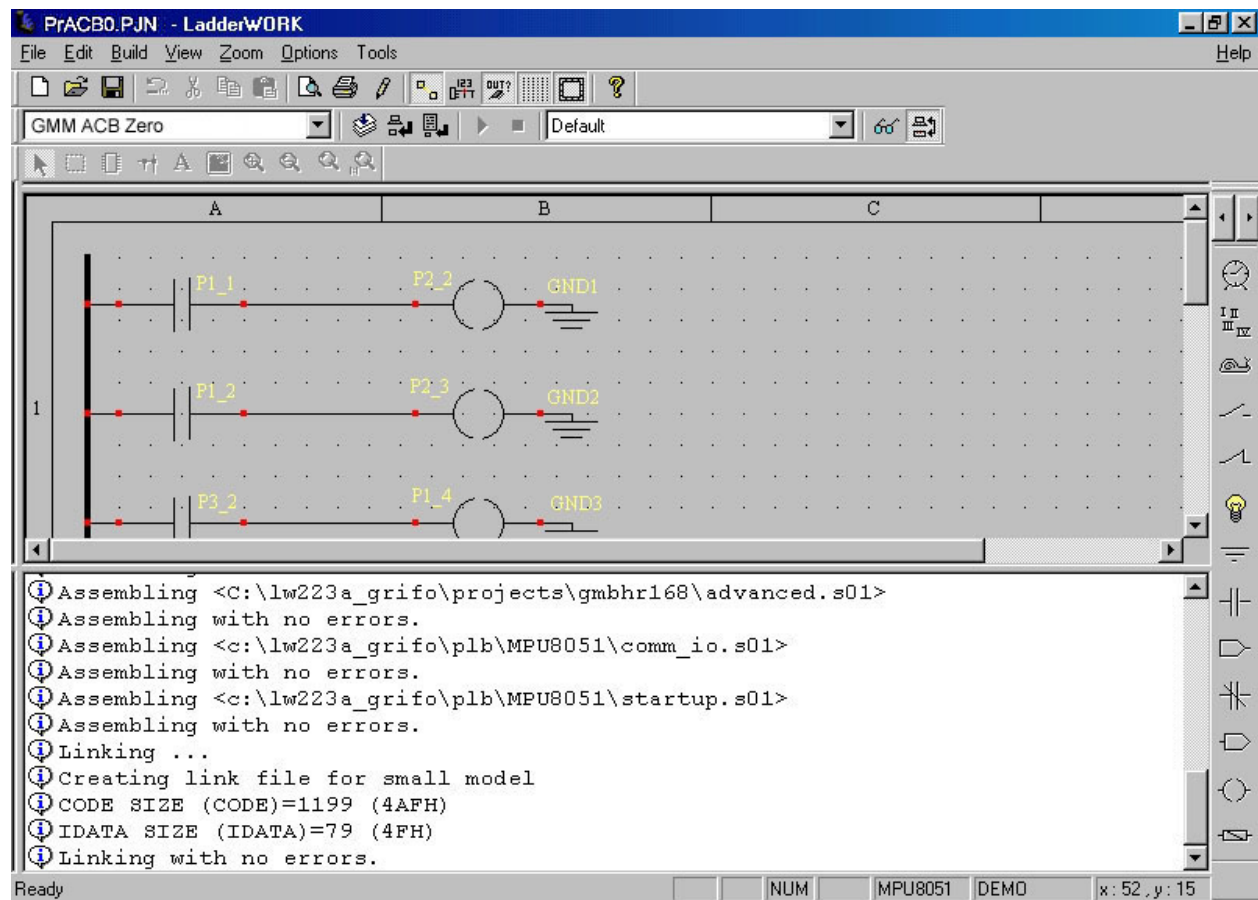


FIGURE 33: COMPILING RESULT WITH LADDER WORK

The compiling time change according to used development **PC**; anyway the user must wait that all the phases are completed, as described in the low window of **IDE**. At this point an **.HEX** file must be obtained equal to those available on **grifo®** disk and already used at points C.

- D4) Reperform the programming and the test of the **.HEX** file obtained by compiler, by executing again the steps C5÷C16.
- D5) When during execution of the steps above described a problem or a malfunction is found, we suggest to read and repeat again all the steps carefully and if malfunction persists please contact directly **grifo®** technicians.

Instead when execution of all the steps is right, the user has realized his first application program that coincides with demo of **GMM 518 Zero**.

At this point it is possible to modify the source of the demo/s program according to application requirements and test the obtained program with the steps above listed (from B1 to D4) in cyclic mode, until the developed application program is completely well running.

About the **FLIP** settings, please remind that they could be inserted only the first time in fact the same program maintains the last settings sucesfully used. Alternatively it could be used the batch modality, that can be often integrated in the development tool **IDE**, in order to speed up the programming and debug phase of the program.

When this focus is reached the developmnet **PC** can be eliminated, by obtaining a self running card, as below described:

E) FINAL PREPARATION OF APPLICATION

- E1) Set the **RUN** mode, that is **J1.b** in Open position, and disconnect the development **PC** if it is not required by same application.

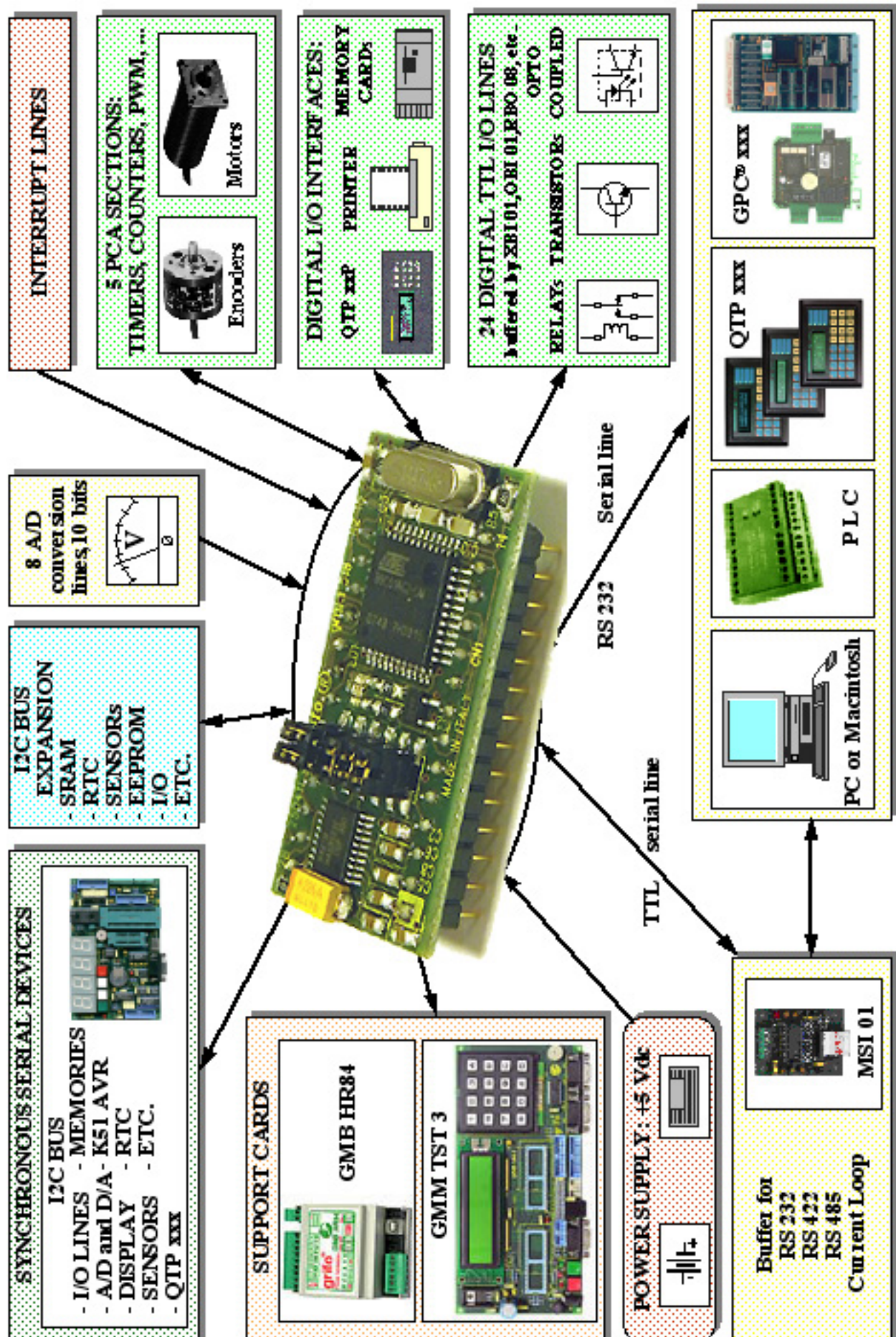


FIGURE 34: CONNECTION EXAMPLES

SOFTWARE DEVELOPMENT TOOLS

A wide selection of software development tools can be obtained, allowing the best use of the module features and to easily complete the necessary applications in a very short time. Generally all software packages available for the mounted microprocessor (or in other words the numerous tools for the **8051** family) can be used, either at high and low level.

All the software development tools supplied by **grifo®** always include many example programs, in source and executable format, fully remarked, that shows how to manage each section of the card. As described in previous chapter these demo program are ready to use and so they speed up the evaluation of the module and the development phase of final application.

Among these we remind:

SYS51CW: Cross Compiler for **C** source programs. It is a powerful software tool that includes Editor, C Compiler, Assembler, Optimizer, Linker, Library, Simulator and Remote Symbolic Ddebugger, included in an easy to use **Integrated Development Environment** for Windows.

SYS51PW: Cross Compiler for **PASCAL** source programs. It is a powerful software tool that includes editor, **PASCAL** Compiler, Assembler, Optimizer, Linker, Library, Simulator and Remote Symbolic Debugger, included in an easy to use **Integrated Development Environment** for Windows.

DDS MICRO C 51: Low cost Cross Compiler for **C** source programs. It is a powerful software tool that includes Editor, C Compiler (Integer), Assembler, Optimizer, Linker, Library, and Remote Debugger, in one easy to use **IDE**. Includes the Library Sources and many Utilities Programs.

BASCOM 8051: Cross Compiler for **BASIC** source programs. It is a powerful software tool that includes Editor, BASIC Compiler and Simulator included in an easy to use integrated development environment for Windows. Many memory models, data types and direct use of hardware resource instructions are available.

µC/51: It is a comfortable, low cost, software package with a complete **IDE** that allows to use an editor, and **ANSI C** Compiler, and Assembler, a Linker and a Remote Source Level Debugger user configurable. Sources of main Libraries and Remote Debugger are included, and so several Utility and demo programs.

LADDER WORK: it is an easy to use system capable to generate automation application using the very famous and diffused contacts logic. It includes a graphic editor to place and connect hardware components of the card (digital **I/O**, counters, **A/D**, etc.) like on an electric diagram and define their properties, an efficient compiler to create the executable code and an utility to download it to card memories. Integrated **IDE** makes comfortable use of all these tools. Delivered on a **CD** for Windows with user manual and hardware key.

PERIPHERAL DEVICES SOFTWARE DESCRIPTION

Below there is a specific description of the software managements of the on board peripheral devices. Whenever the reported documentation is not sufficient, please search a more detailed description of the devices in manufacturing company data sheets. Furthermore in this chapter the microcontroller internal peripheral devices (Timer, Counter, PCA, I/O Ports, A/D, UART, Interrupt, Watch Dog, etc.) are not described; so if their programming is necessary, please refer to APPENDIX A of this manual.

In the following paragraphs the **D7÷D0** and **.0÷7** indications denote the eight bits of the combination involved in **I/O** operations.

CONFIGURATION INPUTS

The status of configuration jumper **J1.c** can be obtained by software, through a simple read operation of the relative bits on **Port 0**:

J1.c -> P0.0

The acquisition is in inverted logic, in fact jumper **Closed** position corresponds to logic level **0 (L)** and jumper **Open** position corresponds to logic level **1 (H)**.

Moreover the **J1.b** acts as a **RUN** or **DEBUG** selector: after a **Reset** or a power on the jumpers is in Close position the **Boot Loader** is Run, otherwise when the Jumpers is Open the user program saved in internal **FLASH** is executed, starting from address **0** of code area.

J1.b -> Closed = DEBUG

J1.b -> Open = RUN

VISUALIZATION LED

The **LED LD1** can be software driven and its status can be defined by simple operations on relative bits of **Port 2**:

LD1 Off (Red) -> P2.7 = 1

LD1 Lit (Red) -> P2.7 = 0

Driving is in inverted logic, in fact **LED** is **ON** when bit is **0 (L)** and **LED** is **OFF** when the corresponding bit is **1 (H)**.

All signals of **Port 0** are kept at logic level **1** during the **Reset** or the power on, so when one of these phases happen, the **LED** is disabled.

MULTIFUNCTIONS SIGNALS

Numerous signals connected to **CN1** can be used with different functionalities that are software selectable by developed application program. The specializations of the multiplexed signals is performed through proper settings of internal registers of microcontroller, as described in the component data sheet. For example the pins **20÷27** can operate as analog inputs (**ANn**) or as digital **I/Os (P1.n)** according with setting of **ADCF** register; when they are configured as **I/Os** then they can operates as **Timer 2** or **PCA** signals, by enabling properly these two peripherals.

MEMORY ACCESS

On **GMM ACB Zero Mini Module** are available different memory types that can be easily managed by user application program, as below described:

<i>Memory</i>	<i>Allocation</i>	<i>Access</i>	<i>Management</i>
64K bytes of FLASH	Code area	R W	Instructions for code area reading Calls to Boot Loader functions
256 bytes of IRAM	Internal data area	R/W	Instructions for internal RAM either with direct and indirect access
2K bytes of ERAM	External data area	R/W	Instructions for external RAM access
2K bytes of EEPROM	Dedicated area	R/W	Through dedicated Special Function Registers

The addressing modalities and the instructions for memory access of the microcontroller are explained in the data sheets of the device, so please consult them or the APPENDIX A at the end of this manual. This know how become necessary when the user program is coded in assembly, viceversa when an high level development tools is used the memories are located and managed automatically and confortably by using specific format specifiers in the variables declarations.

It is is important remind that the last **8** bytes of on board **EEPROM** are reserved for calibration data and they can't be modified. Both the user application program and the **ISP** management of **EEPROM** must preserve these bytes in order to avoid malfunctions of Mini Module.

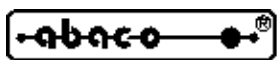
BIBLIOGRAPHY

In this chapter there is a complete list of technical books, where the user can find all the necessary documentations on the components mounted on **GMM 518 Zero**.

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

The described manuals can be requested directly to manufacturer or local dealers. Alternatively this information and/or their upgrades can be found in specific internet web pages, of the listed companies.

Many manuals in electronic format are available also in our technical documentation service, as described in APPENDIX A.



grifo®

ITALIAN TECHNOLOGY



APPENDIX A: ON BOARD DEVICES DESCRIPTION

grifo® provides a completely free technical documentation service to make available the data sheets of on board components, through its web site. This chapter shows only the first pages of the data sheets, but the user can download the complete documents from the "Technical documentation Service" link, on the home page.

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, VFP64 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 ⁽¹⁾ 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

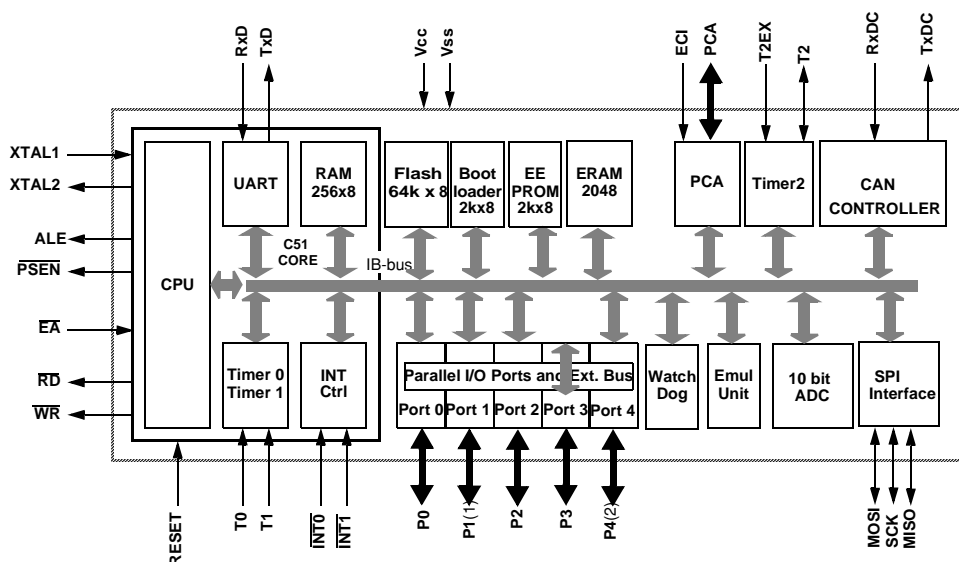
The AT89C51CC03 is the third member of the CANary™ family of 8-bit microcontrollers dedicated to CAN network applications.

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.

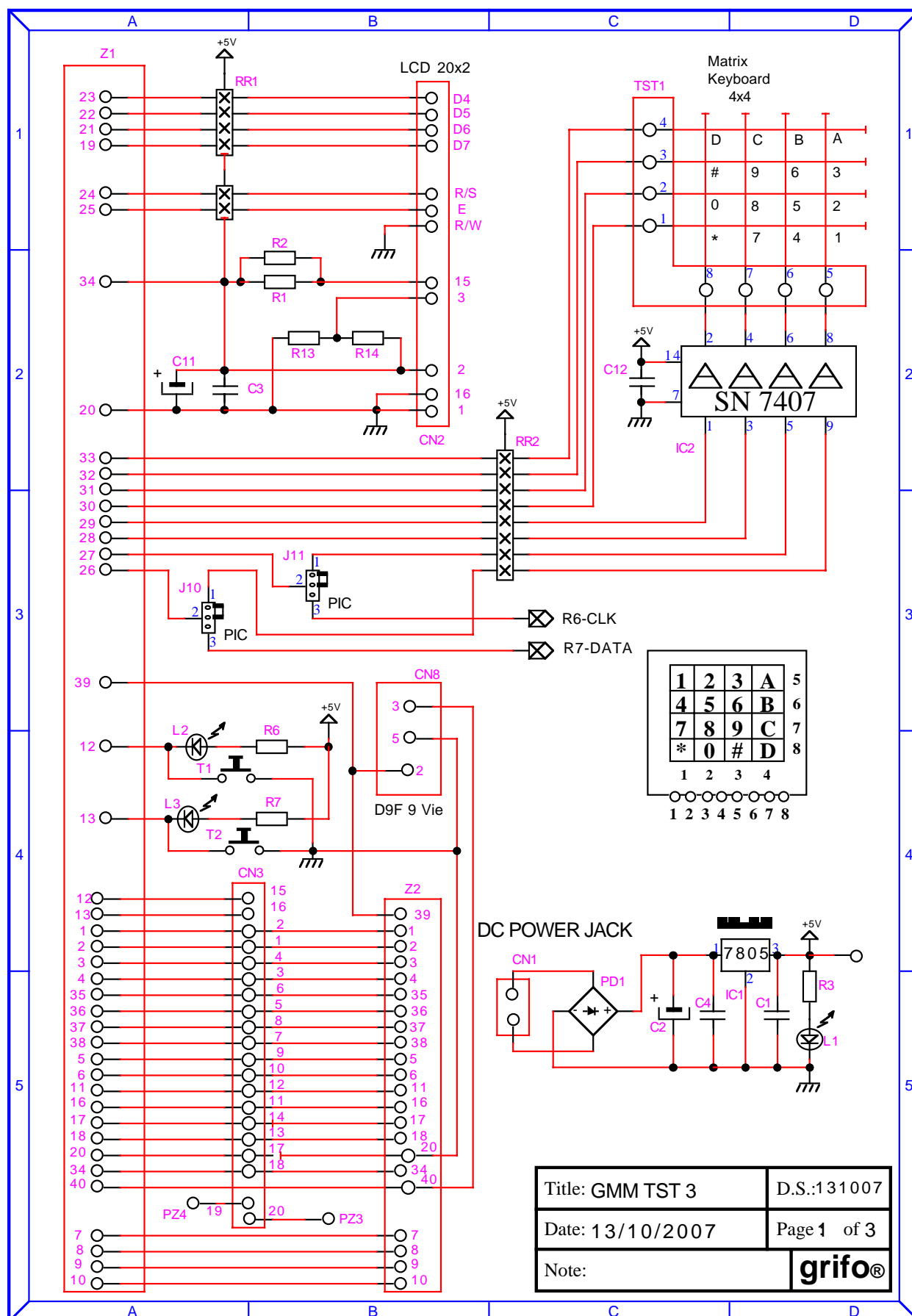
Primary attention is paid to the reduction of the electro-magnetic emission of AT89C51CC03.

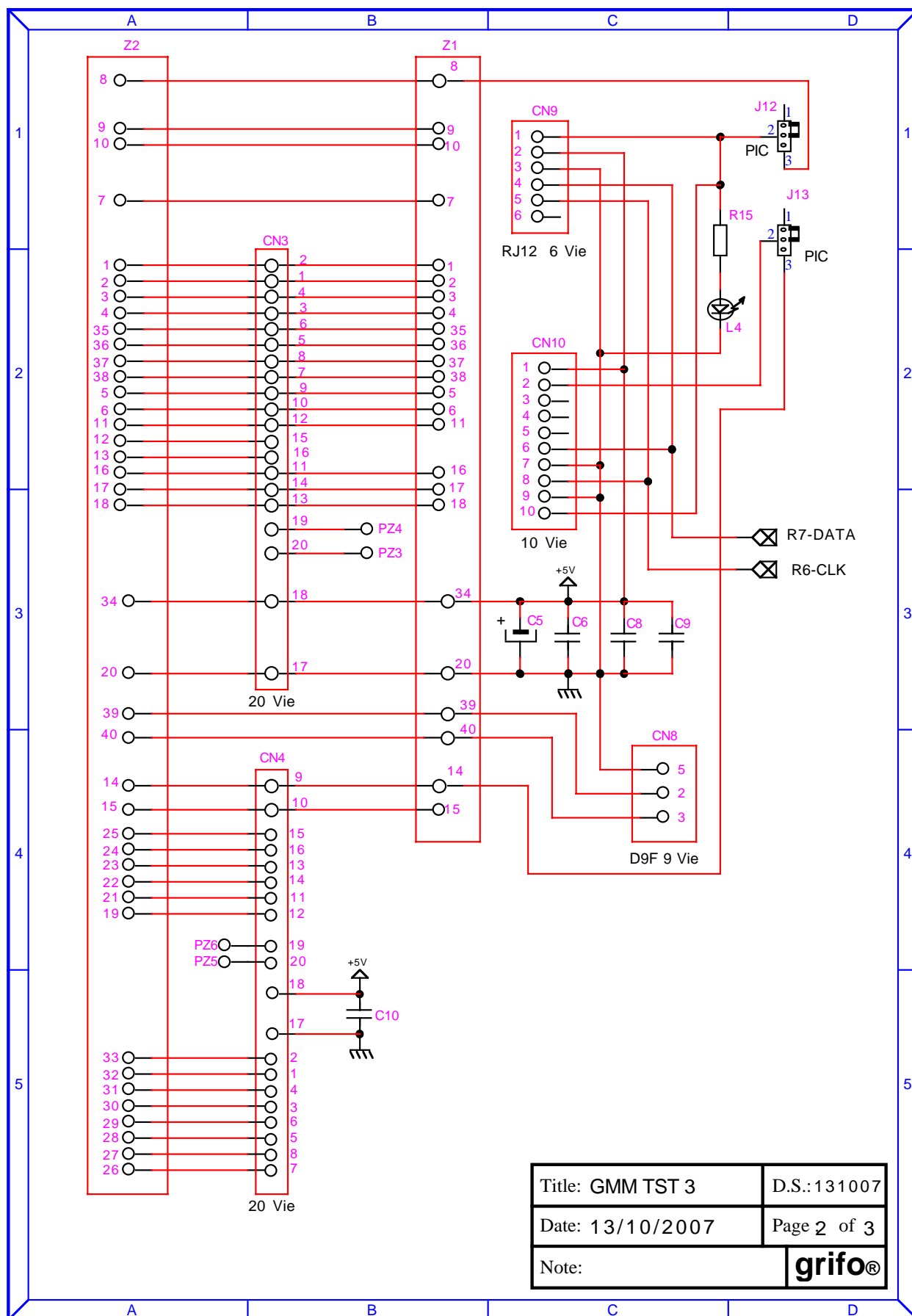
Block Diagram

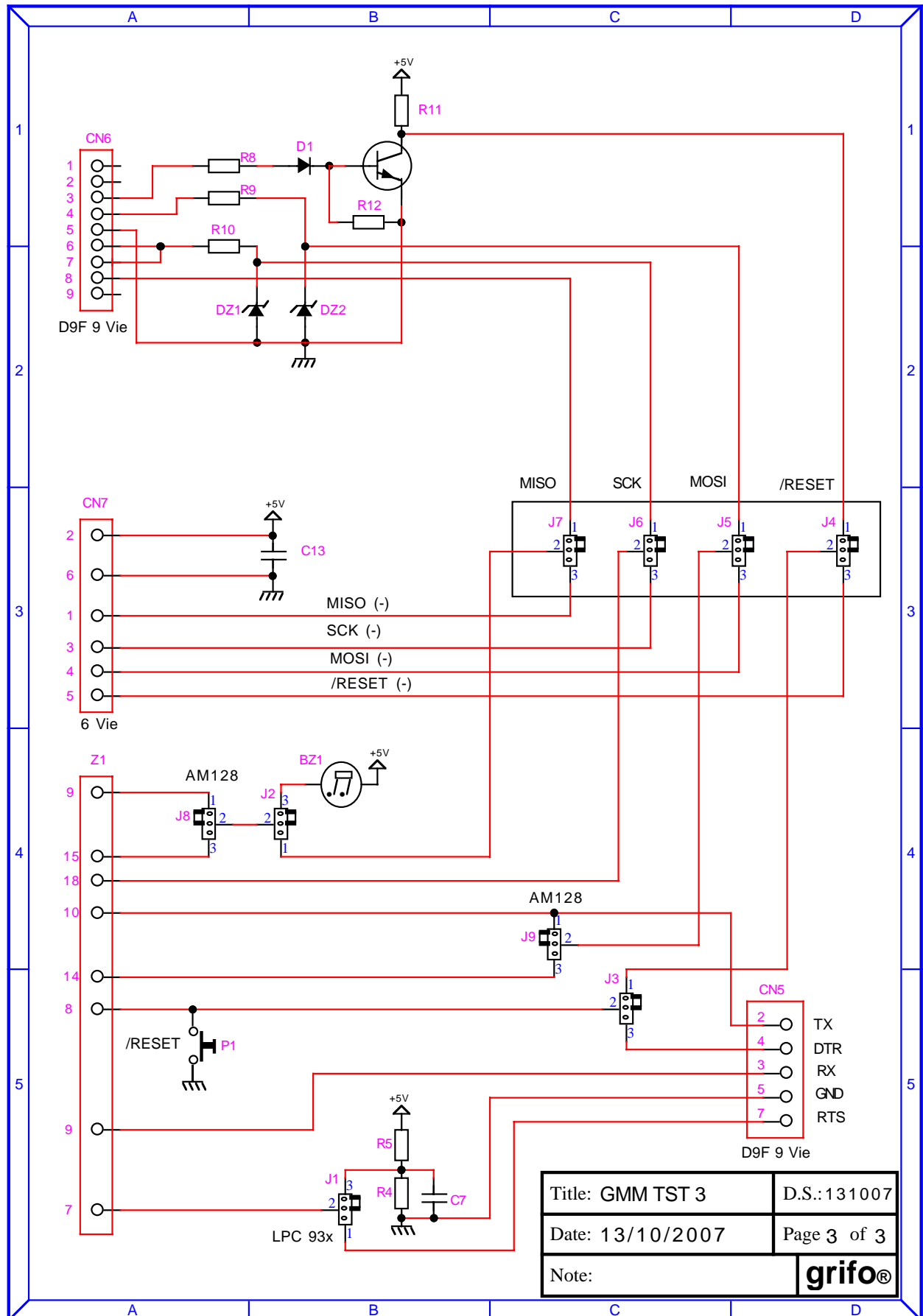


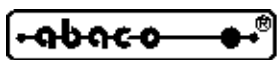
- Notes:
1. 8 analog Inputs/8 Digital I/O
 2. 5-Bit I/O Port

APPENDIX B: GMM TST 3 ELECTRIC DIAGARAM









grifo®

ITALIAN TECHNOLOGY



APPENDIX C: BASE CONFIG., OPTIONS, ACCESSORIES

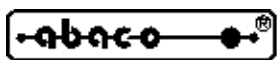
In corrispondenza della prima acquisto, o dopo una riparazione, il **GMM 518 Zero** è fornito nella sua configurazione base. Le caratteristiche di questa configurazione sono state descritte molte volte nel manuale (by using also the name default configuration) e in questo appendice sono riassunte, opportunamente divise nella seguente tabella.

DISPOSITIVO	CONFIG.	FUNZIONE
J1.A J1.B J1.a	2-3 2-3 Connected	<i>Configuration</i> of the asincronous serial line on to CN1 for RS 232 standard communication.
J1.b	NOT Connected	NOT CONNECTED It doesn't connect the microcontroller /PSN line by selecting the RUN mode. After each ON or RESET of the Mini Module the program written on to the FLASH built on the microcontroller starts.
J1.c	NOT Connected	NOT CONNECTED It leaves free the configuration input acquirable via software so it will become logic level 1 (H) .
FLASH for Boot Loader	Boot Loader	When DEBUG Mode (J1.b=Connected) is selected it allows the PC communication that executes the FLIP for managing the ISP programming of the on board FLASH end EEPROM memories.
Internal FLASH for User	Demo Program	It makes signal LED blinking and working with a console on RS232 serial line fixed at 19.200 Baud, 8 bit x char, No parity, 1 Stop Bit, No Handshake.
Internal EEPROM for User	Calibration Data	The last 8 byte of this memory are reserved to the calibration data and can't be modified. Both user program and ISP managing must save these bytes for avoiding Mini Module malfunction.

FIGURE C1: DEFAULT CONFIGURATION

Previous table can be reduced anymore by asserting that the product is supplied calibrated, with all the I/O lines free for user and ready for an RS 232 serial communication, with a physic protocol of 19.200, 8, No, 1.

The **GMM 518 Zero** hasn't options that can be added in the order phase; as described in the SUPPORT CARDS chapter there are other boards suitable for direct **Mini Module** mounting, that allows an immediate use of all the available resources.



grifo®

ITALIAN TECHNOLOGY



APPENDIX D: ALPHABETICAL INDEX

Simboli

.HEX 31, 34, 37, 40

/RES 17

µC/51 33, 35, 42

A

A/D converter 6, 12, 13, 20

Access 44

Accessories 28, C-1

Analog inputs 13, 17

Assistance 1

AT89C51CC03 12, 31, 33, A-1

Attribute 30

B

BASCOM 8051 33, 35, 42

BatchISP 33

Baud rate 29, 31

Bibliography 45

Bits x chr 29

Boot Loader 22, 31, 43

C

CAN 10, 17, 23

Clock 12

CN1 16

Code area 22

Communication 5, 29

COMx 28

Connector 16

Container 1

Control signals 17

Conversion time 12

Counter 12, 16

CPU 12

Current consumption 13

D

Data area 44

Data bits 29

Data sheets A-1

DDS MICRO C 51 42

DEBUG 18, 19, 22, 43
Declaration file 34
Default configuration 19, C-1
Demo programs 29, 30, C-1
Development PC 28
Directive 1
Directives 1
Display LCD 127
Documentation 1, A-1

E

EEPROM 22, 44
Electrostatic noises 1
ERAM 44
ESD 1

F

First use 28
FLASH 22, 44
FLIP 29, 32
Flow control 29

G

General information 4
GMB HR84 23, 41
GMM TST 26
GMM TST 3 26, 41, B-1
Ground 17

H

Handshake 29
HYPERTERMINAL 29

I

I2C BUS 16
Impedance 13
Interrupt 12, 19
Introduction 1
IRAM 44
ISP 22, 29, 32

J**JFE 36**

Jumpers 18, 19

L**LADDER WORK 33, 38, 42**

Latch 22

LEDs 14, 16, 18, 19, 43

Locations 3

M

Matrix keyboard 26

Memory 8, 12, 22, 44

Memory allocation 44

MSI 01 5, 41**N**

Normative 1

O

Options C-1

optocoupled inputs 24

P

Parity 29

PC 28, 41

PCA 17

Peripherals 22, 43

Physic protocol 29

Pitch 16

Port 6, 16, 43

Power down 14

Power failure threshold 13

Power on time 12

Power supply 13

Priority 19

Protection 1

R

Reference voltage 17

Relative humidity 12

Relay outputs 24

Reset 17

Resolution 12

RS 232 5, 13, 16, 21, 28, C-1

Rules 1

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

S

Safety 1

Security 8

Serial communication selection 20

Serial line 19

Size 12

Socket 16

Stop bits 29

Support cards 24

SYS51CW 42

SYS51PW 42

T

Temperature 13

Temperature range 12

Timeout error 31

Timer 12, 16

Trademarks 2

TTL 5, 20, 21

V

Version 3

Vref 17

W

Warranty 1

Watch Dog 8, 12, 17

Weight 12

X

X2 mode 8