

# GMM AM32

**grifo®** Mini Module AT mega 32L

## TECHNICAL MANUAL



**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 (a.r.) FAX: +39 051 893.661

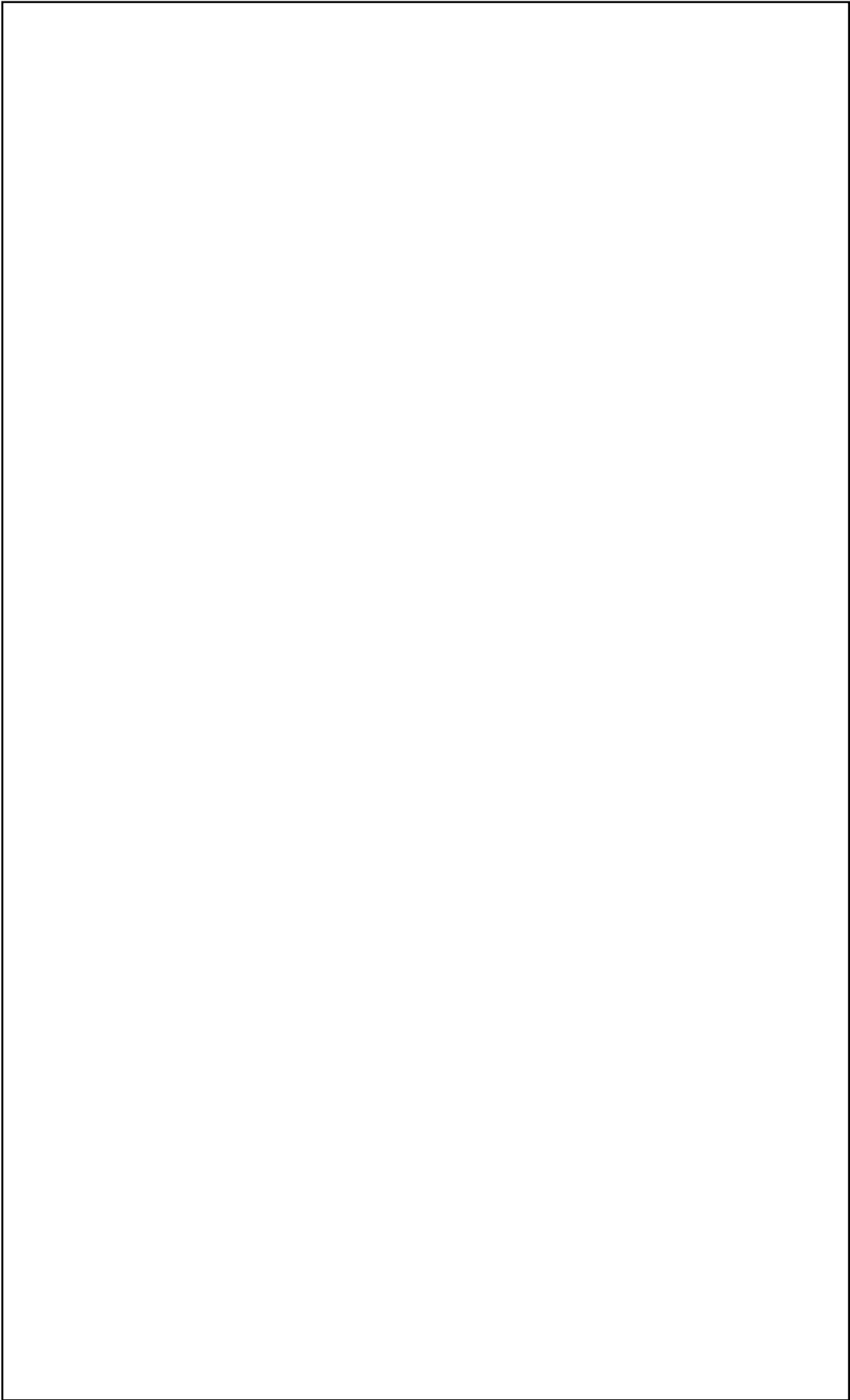


GMM AM32

Rel. 5.10

Edition 02 March 2006

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



# GMM AM32

**grifo®** Mini Module AT mega 32L

## TECHNICAL MANUAL

Standard container with **40** pins male socket, dual in line, **100** mils pitch, **600** mils width; very small dimension: **20.8 x 61.5 x 16.3 mm**; single power supply voltage required in the wide range from **3.0** to **5.0 Vdc** (the current consumption can change according to module connections); availability of power saving setting as **Idle Mode** and **Power Down Mode**; **Atmel ATmega32L** microcontroller with external **7.3728 MHz** crystal; **32KBytes FLASH** for code, up to **2KBytes FLASH** can be reserved for **Boot Loader**, **2KBytes SRAM** for data, **1KBytes EEPROM** for data; **2** analog comparators channels with different input and output configurations options that allow to create easily bipolar **A/D** conversion; **8** channels multiplexed **A/D** converter, resolution **10** bits and **20 µs** conversion time; **21 Interrupt** sources; **3 Timers Counters**, **8** or **16** bits, with functions of **PWM**, compare, capture, etc.; **32** digital **I/O** lines available on connector; One hardware serial communication line with programmable physical protocol (Baud Rate up to **115200 Baud**, **8** or **9** data bits, **1** or **2** stop bit), **RS 232** buffered or at **TTL** level; **JTAG** interface for in-circuit remote debugging; **I<sup>2</sup>C BUS** controller, completely software configurable; **SPI** interface programmable for synchronous, high speed communications; **Reset** circuitry; **Real Time Clock** for long and accurate timing, active even in low power modes, capable to manage date and time and to generate periodic interrupts; **240** bytes of **SRAM** for configuration parameters; **RTC** and **SRAM** backed with on-board **Lithium** battery; **Watch Dog** section programmable from **16 ms** up to **2 s**; **2** status **LED** managed by software through one digital **I/O** line; **8** ways dip switch for operating modes selection of the **Mini Module**; internal **FLASH** and **EEPROM** can be managed through **ISP** (In System Programming); **Boot Loader** preinstalled that allows to program microcontroller **FLASH** and **EEPROM** through a **RS 232** serial port; wide range of Software Development tools used to develop the user application program, as: Assembler; **C** compilers (ICC AVR, DDS Micro C); **BASIC** Compilers (BASCOM AVR); etc.; long list of Demo Programs and use examples supplied under source form, duly remarked, and executable form 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](mailto:grifo@grifo.it)

<http://www.grifo.it>

<http://www.grifo.com>

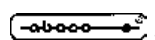
Tel. +39 051 892.052 (a.r.) FAX: +39 051 893.661



GMM AM32

Rel. 5.10

Edition 02 March 2006



, 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
CARD VERSION .....	3
GENERAL INFORMATION .....	4
DIGITAL I/O LINES .....	6
SERIAL COMMUNICATION .....	6
CLOCK .....	6
MEMORY DEVICES .....	8
I <sup>2</sup> C BUS INTERFACE .....	8
SPI INTERFACE .....	8
DIP SWITCH .....	9
A/D CONVERTER .....	10
ANALOG COMPARATOR .....	10
TIMER COUNTER AND PWM .....	10
WATCH DOG .....	10
REAL TIME CLOCK .....	10
TECHNICAL FEATURES .....	12
GENERAL FEATURES .....	12
PHYSICAL FEATURES .....	13
ELECTRIC FEATURES .....	13
INSTALLATION .....	14
VISUAL SIGNALATIONS .....	14
CONNECTIONS .....	14
CN1 - EXTERNAL POWER SUPPLY CONNECTOR .....	14
MINI MODULE CONFIGURATION .....	16
SERIAL COMMUNICATION SELECTION .....	18
CONNECTOR SIGNALS INTERFACEMENT .....	20
POWER SUPPLY .....	20
INTERRUPTS .....	21
JTAG INTERFACE .....	21
SUPPORT CARDS .....	22
USE WITH GMB HR168 MODULE .....	22
USE WITH GMM TST2 MODULE .....	24
HOW TO START .....	26
RICOMPILATION WITH BASCOM AVR .....	30
RICOMPILATION WITH ICC AVR .....	32
SOFTWARE DESCRIPTION .....	34

<b>PERIPHERAL DEVICES SOFTWARE DESCRIPTION .....</b>	<b>36</b>
<b>ACTIVITY LEDS .....</b>	<b>36</b>
<b>DIP SWITCH .....</b>	<b>36</b>
<b>BACKED SRAM + SERIAL RTC .....</b>	<b>37</b>
<b>CPU INTERNAL PERIPHERALS .....</b>	<b>37</b>
 <b>BIBLIOGRAPHY .....</b>	 <b>38</b>
 <b>APPENDIX A: DATA SHEET .....</b>	 <b>A-1</b>
<b>ATmega32L .....</b>	<b>A-1</b>
<b>PCF 8583 .....</b>	<b>A-2</b>
 <b>APPENDIX B: ISP PROGRAMMING WITH GMM TST 2 .....</b>	 <b>B-1</b>
<b>COMMON OPERATIONS FOR ALL PROGRAMMERS .....</b>	<b>B-1</b>
<b>PROGRAMMING USING grifo® MP AVR-51+; grifo® UEP 49 .....</b>	<b>B-3</b>
<b>PROGRAMMING USING EQUINOX EPSILON5 .....</b>	<b>B-5</b>
<b>PROGRAMMING USING ATMEL AVR ISP .....</b>	<b>B-7</b>
<b>PROGRAMMING USING PONY PROG .....</b>	<b>B-9</b>
 <b>APPENDIX C: GMM TST 2 ELECTRIC DIAGRAM .....</b>	 <b>C-1</b>
 <b>APPENDIX D: ALPHABETICAL INDEX .....</b>	 <b>D-1</b>

# FIGURES INDEX

FIGURE 1: LOCATION OF REVISION NUMBER .....	3
FIGURE 2: BLOCKS DIAGRAM .....	7
FIGURE 3: GMM AM32 PHOTO .....	9
FIGURE 4: COMPONENTS MAP (COMPONENTS SIDE) .....	11
FIGURE 5: LEDs TABLE .....	14
FIGURE 6: CN1 - SOCKET WITH MINI MODULE SIGNALS .....	15
FIGURE 7: LEDs, DIP SWITCH, ETC. LOCATION .....	16
FIGURE 8: DSW1 8 WAYS DIP SWITCH TABLE .....	17
FIGURE 9: EXAMPLE OF RS 232 SERIAL CONNECTION .....	19
FIGURE 10: EXAMPLE OF TTL SERIAL CONNECTION .....	19
FIGURE 11: IMAGE OF MODULE GMB HR 168 AND GMM AM32 .....	23
FIGURE 12: GMM TST 2 + GMM AM32 PHOTO .....	25
FIGURE 13: RS 232 SERIAL CONNECTION BETWEEN A GMM AM32 AND A PC .....	26
FIGURE 14: EXAMPLES TABLE .....	27
FIGURE 15: BOOT LOADER UTILITY .....	29
FIGURE 16: LOADING A SOURCE FILE WITH BASCOM AVR .....	30
FIGURE 17: CONFIGURATION OF COMPILER BASCOM AVR .....	31
FIGURE 18: COMPILATION WITH BASCOM AVR .....	31
FIGURE 19: LOADING PROJECT FILE WITH ICC AVR .....	32
FIGURE 20: COMPILATION WITH ICC AVR .....	32
FIGURE 21: TOP VIEW OF GMM AM32 .....	35
FIGURE 22: AVAILABLE CONNECTIONS DIAGRAM .....	39
FIGURE B-1: DIAGRAM OF INTERFACE BETWEEN GMM TST 2 AND MP AVR-51+ .....	B-2
FIGURE B-2: DIAGRAM OF INTERFACE BETWEEN GMM TST 2 AND UEP 49 .....	B-2
FIGURE B-3: DIRECT CONNECTION BETWEEN GMM AM32 AND SEP 40 .....	B-3
FIGURE B-4: DIRECT CONNECTION BETWEEN GMM AM32 AND UEP 49 .....	B-3
FIGURE B-5: COMPONENT SELECTION USING PG4UW .....	B-4
FIGURE B-6: DIRECT CONNECTION BETWEEN GMM AM32 AND EPSILON5 OR AVR ISP .....	B-5
FIGURE B-7: DIRECT CONNECTION BETWEEN GMM AM32 AND MP AVR-51+ .....	B-5
FIGURE B-8: COMPONENT CONFIGURATION USING PG4UW .....	B-6
FIGURE B-9: PROGRAMMER CONFIGURATION USING PG4UW .....	B-6
FIGURE B-10: PROJECT CONFIGURATION USING EQTOOLS .....	B-8
FIGURE B-11: CPU SELECTION USING AVR STUDIO .....	B-10
FIGURE B-12: CPU CONFIGURATION USING AVR STUDIO .....	B-10
FIGURE B-13: AVR ISP CONFIGURATION USING AVR STUDIO .....	B-10
FIGURE B-14: LOGO OF PONY PROG .....	B-11
FIGURE B-15: CPU SELECTION WITH PONY PROG .....	B-12
FIGURE B-16: CPU CONFIGURATION WITH PONY PROG .....	B-12
FIGURE B-17: PROGRAMMING USING PONY PROG .....	B-12
FIGURE C-1: GMM TST2 ELECTRIC DIAGRAM (1 OF 3) .....	C-1
FIGURE C-2: GMM TST2 ELECTRIC DIAGRAM (2 OF 3) .....	C-2
FIGURE C-3: GMM TST2 ELECTRIC DIAGRAM (3 OF 3) .....	C-3





## 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 Mini Module are not provided with any kind of ESD protection. They are connected directly to their respective pins of microcontroller. Mini Module is affected by electrostatic discharges. Personnel who handles Mini Modules is invited to take all necessary precautions to 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 etc. installation are destined - and then executable - always and in exclusive way from specialized warned and educated personnel, or directly from the TECHNICAL AUTHORIZED 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 informations of this manual. After this reading, the user can use the general index and the alphabetical index, respectly at the begining and at the end of the manual, to find information in a faster and more easy way.

## CARD VERSION

This handbook make reference to card **GMM AM32** with printed circuit version **221003**.

The validity of the information contained in this manual is subordinated to the printed circuit revision number and so the user must always check the correspondance.

The printed circuit revision number is always printed in several positions of the circuit and following figure shows the easiest to access.

### PRINTED CIRCUIT REVISION NUMBER

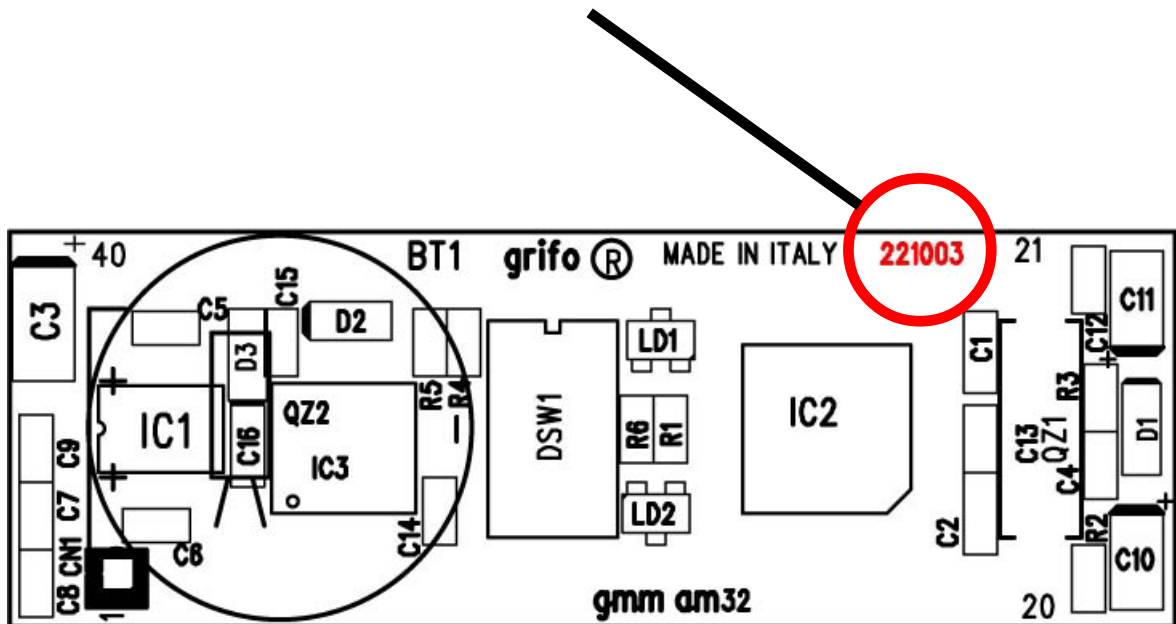


FIGURE 1: LOCATION OF REVISION NUMBER

## GENERAL INFORMATION

**GMM AM32** (**grifo**<sup>®</sup> Mini Module with ATmega32L) is a module based on microcontroller **Atmel ATmega32L**, a powerful and complete system on a chip provided with **CPU**, integrated memory (both for data and code), a **watch dog**, many digital I/O lines, multifunction **timers counters** with capture and compare capability, two asynchronous serial lines, one **I<sup>2</sup>C BUS** synchronous serial line, 2 comparators for analog signals, **A/D** converter, etc.

In module's very small area, are already mounted the components that exploit microcontrollers' performance and that allow each functionality mode like a reset generator circuitry.

Possible applications of **GMM AM32** Mini Modules are several.

For example, smart **Intelligent Nodes** with local functionalities as PID algorithms for controlling temperatures, motors, valves, etc. or **decentralized systems** as robots, automation of production line machines, big factory automations.

Moreover **Teleacquisition** and **Telecontrol** on medium and low distances, the **Home Automations** (lights turning ON/OFF, heating and cooling systems control, supervision of electric devices, security and access control systems), **car automations** (lights turning ON/OFF, heating and cooling systems control, supervision services for drivers, anti-theft and access control systems, functionality checks).

It is really important the **Didactics** use in fact **GMM AM32** offers a very low cost system suitable to learn a **RISC** microcontroller with AVR core and to develop the typical start applications for the students.

For this purpose it is likewise interesting the **GMM TST 2** support card that solves the problems for power supply, for serial connection to development PC and for module lines connection.

In the same support card there is a matrix keyboard and a LCD display that allow test and experience with low cost user interface solutions.

**GMM AM32** is delivered with a Boot Loader preinstalled. This program allows to reprogram Flash and EEPROM through a simple RS 232 serial port (for example, PC COM port).

On board microcontroller electronics also supports the well-known **JTAG** interface, that allows to control its working status directly on the application it is running.

In this case debugging is controlled through the PC and a specific interface to connect to the six signals of Mini Module directly available on its socket.

The PC program that controls JTAG interface is AVR Studio, free IDE developed by Atmel they distribute through their website.

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 are:

- Standard container with **40 pins male socket**, dual in line, **100 mils** pitch, **600 mils** width
- Very small dimension: **20.8 x 61.5 x 16.3 mm**
- Single power supply voltage required in the wide range from **3.0** to **5.0 Vdc** (the current consumption can change according to module connections)
- Availability of power saving setting as **Idle Mode** and **Power Down Mode**
- **Atmel ATmega32L** microcontroller with external **7.3728 Mhz** crystal
- **32 KBytes FLASH** for code, up to **2KBytes FLASH** can be reserved for **Boot Loader**, **2KBytes SRAM** for data, **1KBytes EEPROM** for data
- **2** analog comparators channels with different input and output configurations options that allow to create easily bipolar A/D conversion

- **8** channels multiplexed **A/D** converter, resolution **10** bits and **20**  $\mu$ s conversion time
- **21** **interrupt** sources
- **3** **Timers Counters**, **8** or **16** bits, with functions of **PWM**, compare, capture, etc.
- **32** digital **I/O** lines available on connector
- **One hardware serial communication line** with programmable physical protocol (Baud Rate up to 115200 Baud, 8 or 9 data bits, 1 or 2 stop bit), **RS 232** buffered or at **TTL** level
- **JTAG** interface for in-circuit remote debugging
- **I<sup>2</sup>C BUS** controller, completely software configurable
- **SPI** interface programmable for synchronous, high speed communications
- **Reset** circuitry
- **Real Time Clock** for long and accurate timing, active even in low power modes, capable to manage date and time and to generate periodic interrupts
- **240** bytes of **SRAM** for configuration parameters
- **RTC** and **SRAM** backed with on-board **Lithium** battery
- **Watch Dog** section programmable from 16 ms up to 2 s
- **2** status **LEDs** managed by software through one digital I/O line
- **8** ways dip switch for operating modes selection of the Mini Module
- Internal **FLASH** and **EEPROM** can be managed through **ISP** (In System Programming)
- **Boot Loader** preinstalled that allows to program microcontroller **FLASH** and **EEPROM** through a **RS 232** serial port
- Wide range of Software Development tools used to develop the user application program, as: Assembler; **C** compilers (ICC AVR, DDS Micro C); **BASIC** Compilers (BASCOM AVR); etc.
- Long list of Demo Programs and use examples supplied under source form, duly remarked, and executable form 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 section on verify their connections please refer to figure 2.

## DIGITAL I/O LINES

The Mini Module **GMM AM32** is provided with 32 TTL digital I/O lines, of the microprocessor Atmel ATmega32L, that are all the signals of Ports PA, PB, PC and PD.

These lines are connected directly to 40 pins connectors with standard **grifo**® Mini Module pin out, allowing to be connected directly to several interface cards.

By software it is possible to define and acquire the function and the status of these lines, and also to match them to peripheral devices (like Timer Counter, Interrupt, I<sup>2</sup>C BUS, SPI, etc.), simply programming some CPU internal registers.

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

## SERIAL COMMUNICATION

On **GMM AM32** there is always availability of one hardware serial line that is completely software configurable for physical protocol (baud rate, stop bits number, length of character, etc) by simply programming some microprocessor registers as described in the manufacturer documentation or in the appendix A of this manual.

The serial lines are connected to CN1 connector at TTL or RS 232 level, thanks to some on board dip switches configuration, so when the card must be connected in a network or at long distance or with other systems that use different electric protocol, the user must provide external drivers (RS 232, RS 422, RS 485, current loop, etc.).

Please remember that on CN1 connector more than standard receive and transmit signals are available 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 an RS 232 handshake. For example it can be used the **MSI 01** module that converts a TTL serial line in any other electric standards in a practical and inexpensive way.

Please read SERIAL COMMUNICATION SELECTION paragraph of this manual or contact directly **grifo**® technician for further explanation or any other necessary information.

## CLOCK

On **GMM AM32** module there are two separate and independent circuits based on crystals, that generate the clock signals for the microcontroller and the RTC module.

The first generates a 7,3728 MHz while the second generates a 32768 Hz.

The choice of using two circuits and two separated clock sources, has the advantage to reduce cost in the larger number of low, middle speed applications and to afford the high speed applications when necessary.

About speed and performances please remind that **GMM AM32** has a RISC microcontroller on board, capable to execute in average one instruction per clock cycle.

So, considering the frequency of quartz installed, execution speed may be greater than 7 MIPS.

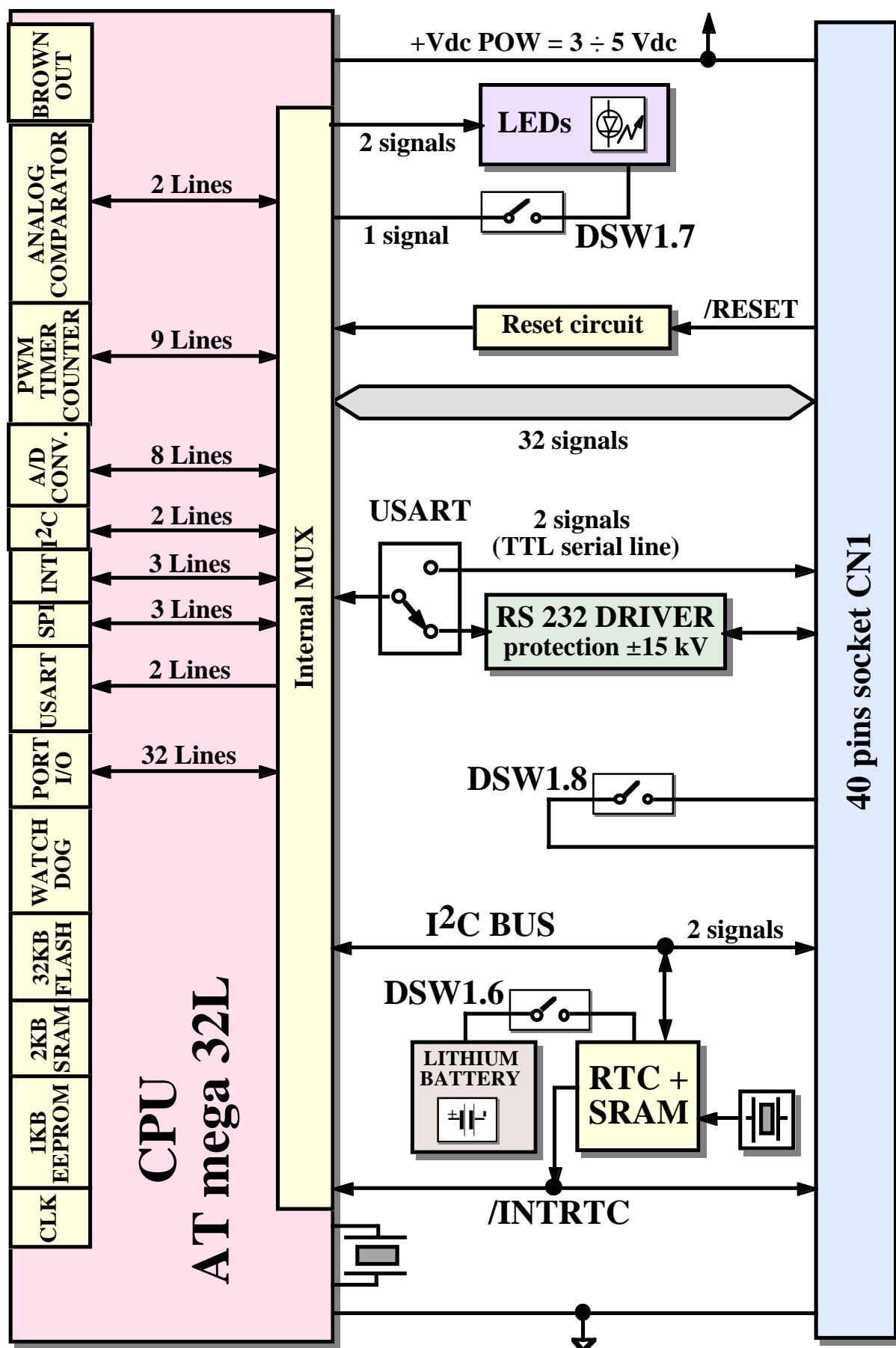


FIGURE 2: BLOCKS DIAGRAM

## MEMORY DEVICES

The card is provided of 35.25K of memory divided with a maximum of **32K** Bytes FLASH EPROM, **2K** Bytes of internal SRAM, **1K** Bytes EEPROM and 240 Bytes in RTC module, reachable only through I<sup>2</sup>C BUS.

The memory configuration must be chosen considering the application to realize or the specific requirements of the user.

Thanks to on board EEPROM (and SRAM of RTC when the back up battery is connected) 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 and configuration, etc. in each working conditions.

Whenever the amount of memory for data is not sufficient (i.e. for data login systems), it is always possible to connect external memory devices (with SRAM, EEPROM, FLASH technologies) through the comfortable and efficient SPI and I<sup>2</sup>C BUS interface of the card.

The addressing of memory devices is controlled by microcontroller as described in the component data sheet or in APPENDIX A of this manual.

## I<sup>2</sup>C BUS INTERFACE

Standard pin out of 40 pins **grifo**<sup>®</sup> Mini Module connector reserves two pins, 12 and 13, to I<sup>2</sup>C BUS interface. These signals are provided with a 4.7 k $\Omega$  pull-up on the Mini Module board.

**GMM AM32** features a hardware I<sup>2</sup>C BUS interface and is managed through microcontroller internal registers. It can work both as master or slave, in transmission or reception.

This interface allows to connect components featuring the same communication standard to expand locally the potentialities of Mini Module.

For this Mini Module, a rich serie of demo programs that explain how to use A/D and D/A converters, memories, temperature sensors, on-board RTC, etc. by a complete and well commented code is available.

Remarkable is the possibility to connect the **grifo**<sup>®</sup> **QTP** operator panels serie through I<sup>2</sup>C BUS. **QTP** are capable to manage alphanumeric and graphic display and several models of keyboards, according to the model selected.

Mini Modules support cards manufactured by **grifo**<sup>®</sup> (like **GMB HR168**) are provided also with a connector dedicated to I<sup>2</sup>C BUS, to easy the field connections.

Please remark that RTC with backed SRAM permanently takes the slave address **A0**, so user applications cannot use it.

For further information please refer to component data sheet or appendix A of this manual.

## SPI INTERFACE

Mini Module **grifo**<sup>®</sup> **GMM AM32** is provided with a SPI serial interface featured through a specific hardware section of the microcontroller.

Signals MOSI, MISO and SCK of SPI interface are available respectively on pins 14, 15 and 18 of connector CN1.

All interface parameters are managed through microcontroller internal registers.

For further information please refer to component data sheet or appendix A of this manual.



## DIP SWITCH

**GMM AM32** Mini Module is provided with one dip switch featuring eight ways whose purpose is to set up several electric parameters of module itself and the card operating mode.

In fact DSW1 allows to decide whether serial interface **USART** must be buffered as **RS 232** or **TTL**, in this latter case allowing to use an external buffer like **RS 422**, **RS 485** or **current loop** drivers. DSW1 also allows to connect or not the back up battery of RTC + SRAM module.

In addition, status of switch DSW1.7 can be acquired by software through reading a specific microcontroller signal (PD4).

For further information please see also the paragraph **DIP SWITCH**.

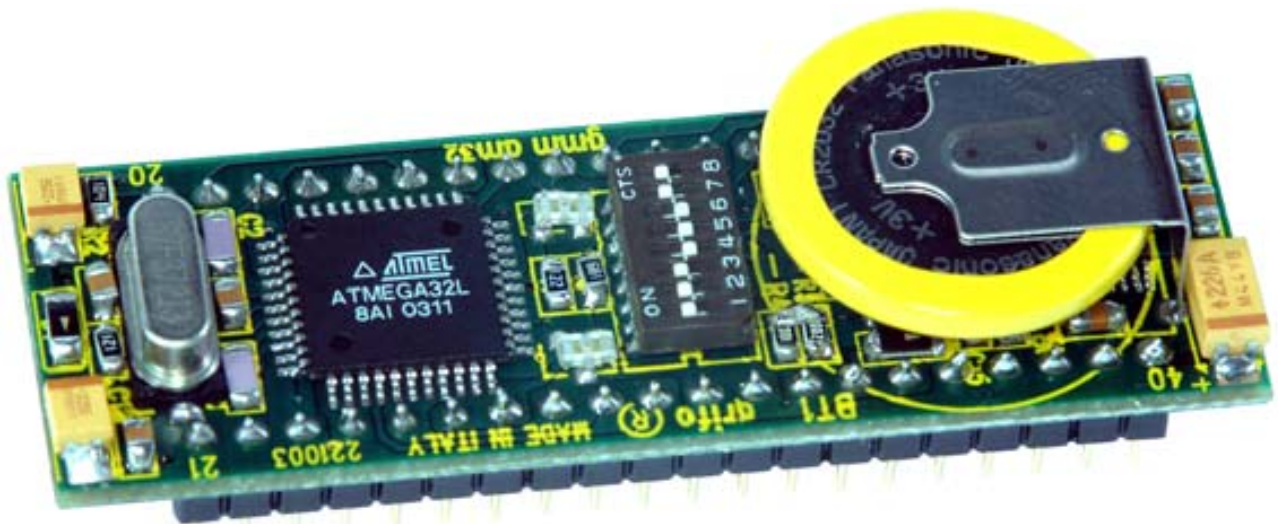


FIGURE 3: GMM AM32 PHOTO

## **A/D CONVERTERS**

Mini Module **GMM AM32** is provided with an eight channels A/D converter, with 10 bit of resolution.

Reference voltage can be fetched externally through a pin or internal reference generator can be used.

Range of signals is from 0 to +Vdc POW, analog signal generator must have a low output impedance.

Conversion end can be used to trigger an interrupt, if enabled.

Peripheral management is performed by manipulating specific microcontroller internal registers.

For further information please refer to Appendix A or to the comments in high-level languages examples.

## **ANALOG COMPARATOR**

Microcontroller ATmega32L features an analog comparator that can select as input both an internal reference voltage and several pins (pin 26 and 27 of CN1, but also A/D converter inputs) through an internal multiplexer.

For further information, please refer to application notes on Atmel web site.

## **TIMER COUNTER AND PWM**

Microcontroller is provided with three Timer/Counter (two featuring eight bits, one featuring sixteen bits) capable to count clock pulses (through a programmable prescaler), level transitions on specific pins and to generate interrupts.

They can also be used in PWM mode, to generate signals of variable duty cycle and frequency set by software with eight or sixteen bits of resolution.

Typical applications of these signals are motor velocity control, in fact several motor control cards are provided with compatible inputs.

Another application is analog signals generation by simply adding an integrator.

## **WATCH DOG**

Microcontroller ATmega32L features a hardware programmable watch dog capable to reset the CPU if it is enabled and the application program does not retrigger it in the activation time.

This latter time can be programmed between 16 ms and 2 s.

## **REAL TIME CLOCK**

Mini Module features an on-board Real Time Clock (in I<sup>2</sup>C BUS with slave address **A0**), capable to manage hours, minutes, seconds, year, month, day and weekday. It also features 240 bytes of SRAM. This component can be completely managed by software and is backed through a back up battery that, if enabled, can warrant data validity in any operating condition.

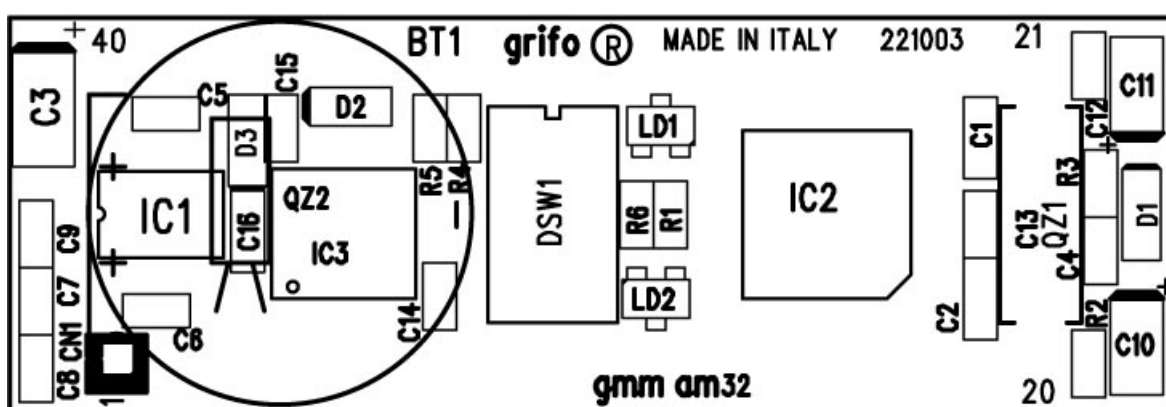


FIGURE 4: COMPONENTS MAP (COMPONENTS SIDE)

## TECHNICAL FEATURES

### GENERAL FEATURES

<b>Resources:</b>	32 TTL digital I/O lines 2 analog inputs on comparator 8 channels A/D converter 3 Timer/Counter for compare, capture, PWM 1 Watch dog section 1 SPI interface 1 I <sup>2</sup> C BUS interface 1 Real Time Clock section 21 interrupt sources 1 hardware serial line RS 232 or TTL 1 Dip switch 8 ways 2 status LEDs
<b>Memories:</b>	32 KBytes FLASH EPROM for code 2 KBytes SRAM for data 1 KBytes EEPROM for data 240 Bytes external SRAM for data (on I <sup>2</sup> C BUS)
<b>CPU:</b>	Atmel ATmega32L
<b>Clock frequency:</b>	7.3728 MHz
<b>Counter maximum frequency:</b>	Clock I/O frequency
<b>Power on time:</b>	from 36 m to 92 ms, settable through configuration bits
<b>Watch dog intervent time:</b>	programmable from about 16 ms up to 2 s
<b>A/D converter resolution:</b>	10 bits
<b>A/D conversion time:</b>	min. 20 µsec (for 10 bits)

**PHYSICAL FEATURES**

<b>Size: (W x H x D):</b>	20.8 x 61.5 x 16.3 mm
<b>Weight:</b>	14 g
<b>Connectors:</b>	40 pins male socket DIL, 100 mils pitch, 600 mils width
<b>Temperature range:</b>	0÷50 °C
<b>Relative humidity:</b>	20%÷90% (without condense)

**ELECTRIC FEATURES**

**Power supply voltage:** +Vdc POW = +3.0 ÷ +5.0 Vdc

<b>Current consumption +Vdc POW = 5 Vdc:</b>	minimum	376 µA
	normal	13 mA
	maximum	21 mA

<b>+Vdc POW = 3 Vdc:</b>	minimum	325 µA
	normal	5 mA
	maximum	8 mA

**Back up battery voltage:** 3.0 Vdc

**Back up battery consumption:** 2.9 µA

**Impedance analog signals generators:** <10 kΩ

**RS 232 protection:** ±15 kVdc

**I<sup>2</sup>C BUS pull-up resistor:** 4.7 kΩ

**Brown out threshold:** 2.7 or 4.0 Vdc with hysteresis

## INSTALLATION

In this chapter there are the information for a right installation and correct use of the **GMM AM32** card. In detail there are the locations and functions of each connector, of the user settable dip switches, LEDs, and so on.

### VISUAL SIGNALATIONS

**GMM AM32** features the LEDs described in the following table:

LED	COLOUR	PURPOSE
LD1	Green	Driven by signal PD4 , OC1B of Mini Module and dip switch DSW1.7. When this latter is ON, LED must be ON and signal must be configured as input.
LD2	Red	Driven by signal PB7 , SCK of Mini Module, it can be used as activity LED and can be managed by software.

**FIGURE 5: LEDs TABLE**

The main function of LEDs is to inform the user about card status, with a simple visual indication and in addition to this, LEDs make easier the debug and test operations of the complete system. To recognize the LEDs location on the card, please refer to figure 7. while for further information please refer to paragraph **ACTIVITY LEDS**.

### CONNECTIONS

The **GMM AM32** 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 7) that simplify and speed the installation phase. Some additional figures shows the pins functionalities and some of the most frequently used connections.

#### **CN1 - EXTERNAL POWER SUPPLY CONNECTOR**

CN1 is a 40 pins, male, dual in line, socket connector with 100 mils pitch and 600 mils width.

On CN1 are available all the interfacement signals of the Mini Module as the power supply, the I/O lines, the synchronous and asynchronous communication line, the on board peripheral devices signals, the operating mode selection lines, etc.

Some pins of this connector have multiple purposes, in fact they can be multiplexed by programming some software registers with several CPU internal devices and the following figure lists all these possible functionalities. So the signals available on CN1 have different types as described in the following **CONNECTOR SIGNALS INTERFACEMENT** paragraph and they follow grifo® Mini Module standard pin out.

To avoid problems in pin counting and numbers the figure 6 shows the signals directly on the top view of the **GMM AM32**; moreover the serigraph reports the pins number on the four corner of the card both on bottom (solder) and top (component) side.

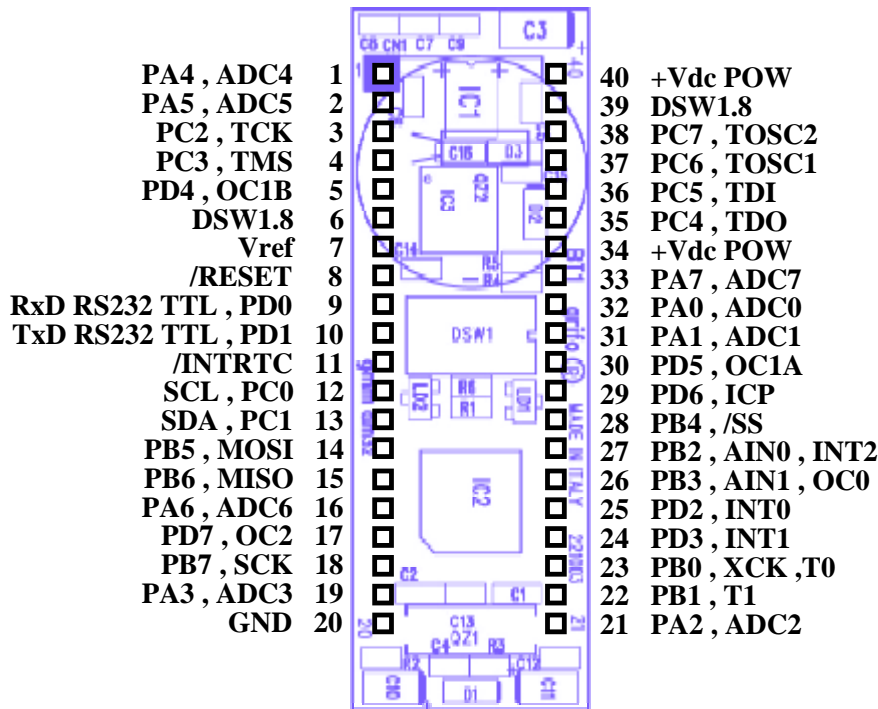


FIGURE 6: CN1 - SOCKET WITH MINI MODULE SIGNALS

Signals description:

<b>RxD RS232 TTL</b>	= I - Receive Data of USARTn it can be buffered as RS 232 or TTL
<b>TxD RS232 TTL</b>	= O - Transmit Data of USART; it can be buffered as RS 232 or TTL
<b>PDI</b>	= I - ISP programming data output line
<b>PDO</b>	= O - ISP programming data input line
<b>PAX, PBx, ... ,PDx</b>	= I/O - CPU I/O TTL signals
<b>SCL</b>	= I/O - I <sup>2</sup> C Bus clock signal
<b>SDA</b>	= I/O - I <sup>2</sup> C Bus reception and transmission signal
<b>Tn</b>	= I - External inputs for counters of timer n-th
<b>INT n</b>	= I - CPU n-th interrupt signal
<b>ADCn</b>	= I - N-th A/D converter analog input
<b>AINn</b>	= I - N-th comparator analog inputs
<b>AREF</b>	= I - A/D converter reference voltage
<b>OCn(A/B/C)</b>	= O - Compare output of n-th section
<b>TDO, TDI</b>	= I/O - Data signals of JTAG interface
<b>TCK, TMS</b>	= I - Clock and control signals of JTAG interface
<b>XCKn</b>	= I - Exeranl clock signals for USARTn (in synchronous mode)
<b>/INTRTC</b>	= O - Periodic interrupt signal generated by RTC PCF 8583
<b>/RESET</b>	= I - Reset signal
<b>MOSI</b>	= O - SPI interface data output
<b>MISO</b>	= I - SPI interface data input
<b>/SS</b>	= I - SPI interface slave select
<b>SCK</b>	= I/O - SPI interface clock
<b>+Vdc POW</b>	= I - Power supply
<b>GND</b>	= - Ground

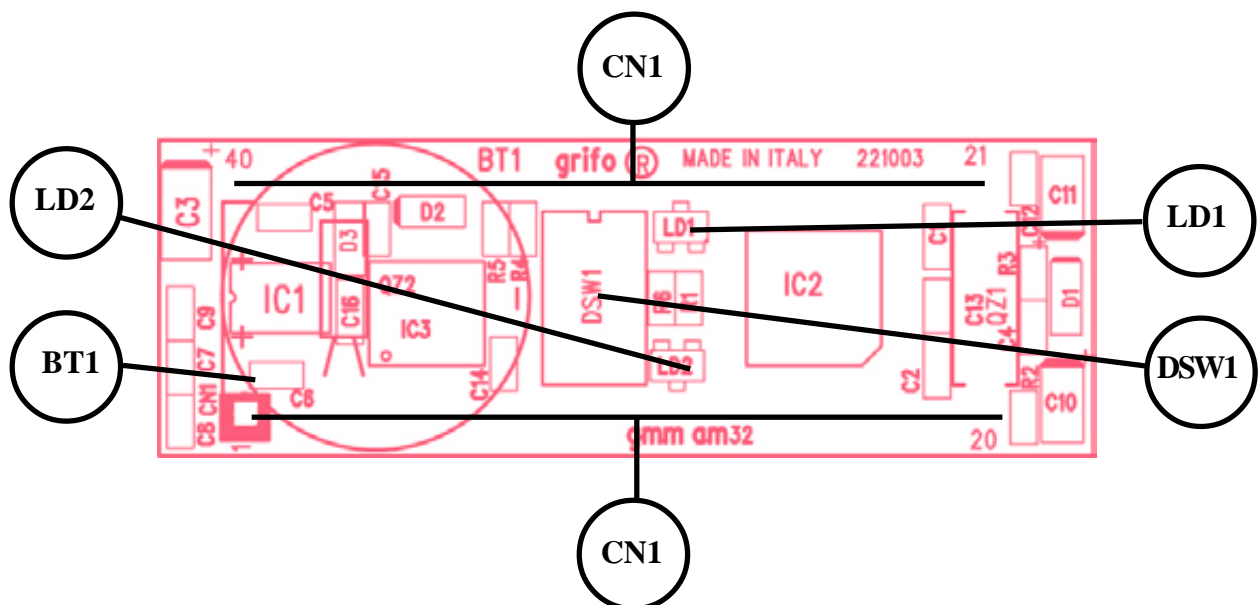
## MINI MODULE CONFIGURATION

On **GMM AM32** module there is an 8 ways dip switch that defines some configurations of the card. In the following figure is reported their list, their position and their functions in all the available connection modes.

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.

To recognize the configuration elements location, please refer to figure 7.

For further information about serial communication lines, please refer to paragraph **SERIAL COMMUNICATION SELECTION**.



**FIGURE 7: LEDs, DIP SWITCH, ETC. LOCATION**



SWITCH	POSITION	PURPOSE	DEF.
DSW1.1	ON	It connects the serial receive signal RxD RS232 TTL , PD0 of the microcontroller to RS232 driver. Used in conjunction with DSW1.3,5.	*
	OFF	It does not connect the serial receive signal of the microcontroller to RS232 driver, thus allowing the direct connection to RxD RS232 TTL , PD0 signal on CN1. Used in conjunction with DSW1.3,5.	
DSW1.2	ON	It connects TxD RS232 TTL , PD1 signal on CN1 to RS232 serial driver. Used in conjunction with DSW1.4.	*
	OFF	It does not connect TxD RS232 TTL , PD1 signal on CN1 to RS232 serial driver thus allowing the direct connection to microcontroller. Used in conjunction with DSW1.4.	
DSW1.3	ON	It connects RxD RS232 TTL , PD0 signal on CN1 to RS232 serial driver. Used in conjunction with DSW1.1,5.	*
	OFF	It does not connect RxD RS232 TTL , PD0 su CN1 signal on CN1 to RS232 serial driver thus allowing the direct connection to microcontroller. Used in conjunction with DSW1.1,5.	
DSW1.4	ON	It connects TxD RS232 TTL , PD1 signal on CN1 directly to microcontroller, with no use of RS232 serial driver. Used in conjunction with DSW1.2.	*
	OFF	It does not connect TxD RS232 TTL , PD1 signal on CN1 to microcontroller, thus allowing the use of RS232 serial driver. Used in conjunction with DSW1.2.	
DSW1.5	ON	It connects RxD RS232 TTL , PD0 signal on CN1 directly to microcontroller, with no use of RS232 serial driver. Used in conjunction with DSW1.1,3.	*
	OFF	It does not connect RxD RS232 TTL , PD0 signal on CN1 to microcontroller, thus allowing the use of RS232 serial driver. Used in conjunction with DSW1.1,3.	
DSW1.6	ON	It connects on-board battery to RTC PCF 8583, allowing to keep date, time and SRAM content even when power supply is turned off.	*
	OFF	It does not connect on-board battery to RTC PCF 8583, so date, time and SRAM content are lost when power supply is turned off.	
DSW1.7	ON	It connects signal PD4 of microcontroller to GND, turning ON LD1. PD4 must be configured as input.	*
	OFF	It does not connectsignal PD4 of microcontroller to GND, allowing to drive LED LD1.	
DSW1.8	ON	It connects pins 6 and 39 of CN1, the 40-pins socket.	*
	OFF	It does not connect pins 6 and 39 of CN1, the 40-pins socket.	

FIGURE 8: DSW1 8 WAYS DIP SWITCH TABLE

## SERIAL COMMUNICATION SELECTION

Serial communication line of **GMM AM32** can be buffered as RS 232 or TTL.

By software, it is possible to define physical communication protocol by setting some microcontroller internal registers.

Serial interface has its own group of registers for configuration.

Electric protocol is selected by hardware and requires the dip switches to be configured correctly, as described in previous tables; the user can set any configuration in autonomy following the below reported information:

### - SERIAL LINE CONFIGURED AS RS 232 (default configuration)

DSW1.1	=	ON
DSW1.2	=	ON
DSW1.3	=	ON
DSW1.4	=	OFF
DSW1.5	=	OFF

### - SERIAL LINE CONFIGURED AS TTL

DSW1.1	=	OFF
DSW1.2	=	OFF
DSW1.3	=	OFF
DSW1.4	=	ON
DSW1.5	=	ON

Figures 9 and 10 show how to connect a generic external system to both serial lines of **GMM AM32**.

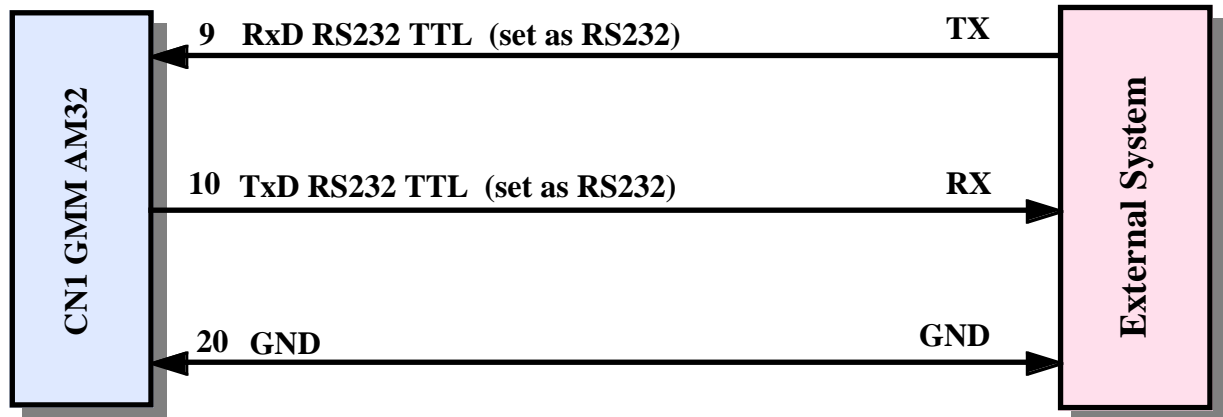


FIGURE 9: EXAMPLE OF RS 232 SERIAL CONNECTION

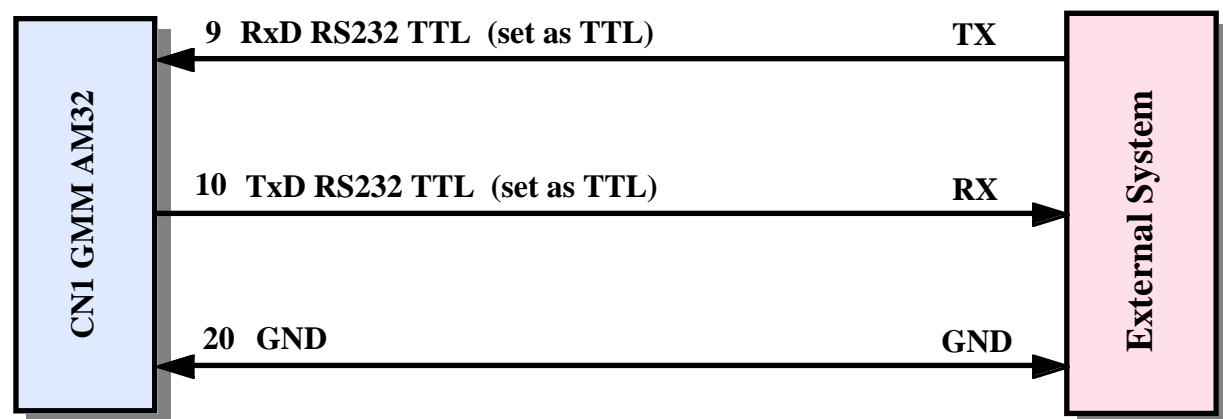


FIGURE 10: EXAMPLE OF TTL SERIAL CONNECTION

## CONNECTOR SIGNALS INTERFACEMENT

To prevent possible connecting problems between **GMM AM32** and the external systems, the user has to read carefully the previous paragraph information and he must follow these instructions:

- For RS 232 signals the user must follow the standard specifications of this protocol, defined by CCITT normative.
- All TTL signals must follow the rules of this electric standard. The connected digital signals must be always referred to card ground (GND) and then the 0V level corresponds to logic state 0, while the +Vdc POW level corresponds to logic state 1. The connection of these lines to devices of the controlled system (encoders, switches, proximity, electric valves, power relays, 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 comparators must be connected to signals generators featuring a low impedance in the range from 0 to selected reference voltage, to assure greater stability and precision.
- The inputs for A/D converter must be connected to signals generators featuring a low impedance in the range from 0 to selected reference voltage, to assure greater stability and precision.
- PWM signals generated by Timer Counter and OCM sections are TTL type so they must be buffered to interface the power circuitry. Typical interfaces can be current driver (if PWM signal is still required) or an integrator circuit if analog voltage is required.
- Also I<sup>2</sup>C BUS and SPI 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 best operational modes and the programmable bit rate: all these conditions allow communications in any condition. On Mini Module, signals SDA and SCL are pulled-up to +Vdc POW through 4.7 kΩ resistors.

## POWER SUPPLY

Mini Module can be supplied by a tension in the range 3 ÷ 5 Vdc called +Vdc POW in this manual. **GMM AM32** design adopted all the circuital and componentistic options that reduce sensibility to noise and reduce consumption, including the possibility to switch the microcontroller to low consumption modes.

In optimal situation, the minimum consumption (in power down mode) is lower than 1 mA, so it can, for example, increase battery life in case of portable applications.

For further information please refer to paragraph ELECTRIC FEATURES.

## INTERRUPTS

A remarkable feature of **GMM AM32** card is the powerful interrupt management.

Here follows a short description of which devices can generate interrupts and their modalities; for further information about interrupts management please refer to the microprocessor data sheet or APPENDIX A of this manual.

- Pin 19 of CN1 -> Generates an interrupt INT0 of microprocessor.
- Pin 18 of CN1 -> Generates an interrupt INT1 of microprocessor.
- CPU peripherals -> Generate an internal interrupt. In detail the possible microcontroller interrupt sources are: Timer Counter, OCM, USART, analog comparator, A/D converter, I<sup>2</sup>C BUS, SPI, EEPROM, ecc.

An interrupt management section, integrated in microcontroller, allows to enable, disable and mask, so the user has the possibility to respond promptly and efficiently to any external event.

The microcontroller has an interrupt section that let the user manage the 21 interrupt sources. So the application program has always the possibility to react promptly to every event.

## JTAG INTERFACE

The microcontroller on board of **GMM AM32** is provided with JTAG interface.

Signals of this interface (TDI, TDO, TMS and TCK) are available respectively on pins 36, 35, 4 and 3 of CN1. Configuration registers of microcontroller must be carefully programmed to enable or disable JTAG functions.

## SUPPORT CARDS

**GMM AM32** Mini Module can be used as a macro components for some support cards either developed by the user or directly chosen from the **grifo**<sup>®</sup> boards. In the following paragraphs are reported the suggested configuration of the most interesting support cards.

### USE WITH GMB HR168 MODULE

Amongst **grifo**<sup>®</sup> cards, **GMB HR168** module is the one designed specifically to provide 16 optocoupled inputs; relay outputs and in addition to the comfortable cabling by quick release connectors and the possibility to install on omega rails.

The complete description of the 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.

**GMB HR168** allows easily to:

- to supply the Mini Module through on board power supply;
- to have sixteen TTL I/O signals of microprocessor ports optocoupled NPN and PNP at the same time and visualized through LEDs (green for the first byte and yellow for the second byte); I/O signals are multiplexed with timer inputs, so developed functions like counters are immediatly available;
- to have eight TTL I/O signals of microprocessor ports on bufferd relays driving and visualized through red LEDs;
- to connect on **PC BUS** and power supply on a dedicated connector;
- to connect immediatly communication serial line through a comfortable 8 pins standard AMP MOD II 8 pins connector;
- to buffer easily USART TTL signals from microprocessor in RS 422, RS 485 or current loop;
- to connect PWM signal through a comfortable standard AMP connector;



FIGURE 11: IMAGE OF MODULE GMB HR 168 AND GMM AM32



## USE WITH GMM TST2 MODULE

Amongst **grifo**® cards, **GMM TST2** is the one designed specifically to provide a first entry point to 28 and 40 pins Mini Modules, with suitable evaluation purposes. The complete description of the product is available in the relative data sheet and technical manual, its electric diagram is in appendix C of this manual. In this paragraph are listed the advantages obtained by using this pair of cards. The **GMM TST2** allows easily to:

- supply the Mini Module through on board AC, DC power supply;
- connect all the I/O signals of microcontroller ports on comfortable connectors compatible with **I/O ABACO**® standard pin out;
- connect immediately USART serial line through a comfortable 9 pins D type connector;
- set and show the status of 2 microcontroller I/O lines through push button and LEDs with different colours;
- generate audible feed back thanks to active buzzer mounted on board;
- develop in a short time user interface applications by using the on board matrix keyboard with 4x4=16 keys and the backlite LCD display with 2 rows of 20 characters;
- develop easily a support card that satisfy customer requirements starting from the supplied electric diagrams;
- program FLASH and EEPROM using the Boot Loader and in ISP modality.

The following configuration is suggested to use the couple **GMM TST 2 + GMM AM32** in their base version, that is RUN mode with serial line buffered in RS 232:

### ***GMM AM32 configuration***

DSW1.1	=	ON
DSW1.2	=	ON
DSW1.3	=	ON
DSW1.4	=	OFF
DSW1.5	=	OFF
DSW1.6	=	OFF
DSW1.7	=	OFF
DSW1.8	=	OFF

### ***GMM TST 2 configuration***

J1	=	2-3
J2	=	2-3
J3	=	not connected
J4	=	not connected
J5	=	not connected
J6	=	not connected
J7	=	not connected

The serial connection cable with development PC is the CCR 9+9 E (or in other words an extension cable provided of D9 Female and D9 Male connectors).





**FIGURE 12: GMM TST 2 + GMM AM32 PHOTO**

## HOW TO START

Across this chapter we presume that you have a **GMM TST 2** or a **GMB HR168** where to install **GMM AM32**.

For further information please refer the specific manual of **GMB HR168 + GMM AM32**.

### A) SERIAL CONNECTION BETWEEN GMM AM32 AND PC

A1) To make the serial connection between **GMM AM32** and a PC, the structure described on figure 13 should be built.

The program delivered to the customer in the Mini Module performs the alternative blink of the on board LEDs, without using the serial interface. Demo program uses widely such interface, it also provides a section dedicated to it. So it is a good idea to make the serial connection as first thing.

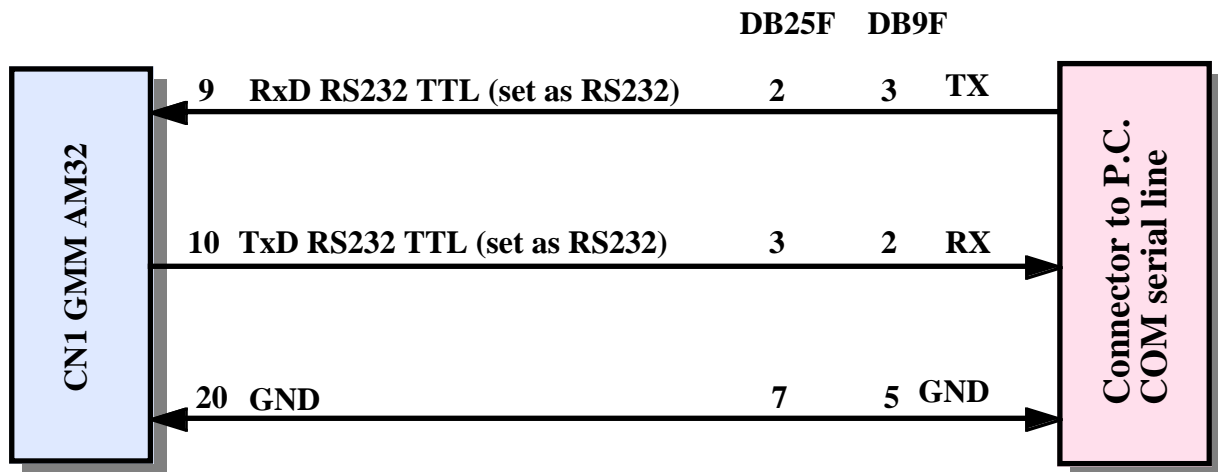


FIGURE 13: RS 232 SERIAL CONNECTION BETWEEN A GMM AM32 AND A PC

A2) Keep ready for running a terminal emulator on PC, configure it to use the serial port where Mini Module is connected with 19200 baud, 8 data bits, 1 stop bit, no parity. If you are using BASCOM AVR, you may simply open the terminal emulator in its IDE.

- A3) Supply **GMM TST 2** or **GMB HR168**. LEDs of Mini Module should start blinking alternatively about twice per second.



BOARD TYPE	GET	ASM	Ladder	Abaco® Link BUS	BASIC CB280	BASIC BASCOM 8051	BASIC BASCOM AVR	PIC BASIC	BASIC VARI	MCS® Basic 52	C	PASCAL	CPU / BLOCK TYPE
MISCELL.	-	-	-	-	-	●	-	-	-	-	-	-	-
CAN GM0	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C51cc03 - 8051 Code
CAN GM1	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C51cc01 - 8051 Code
CAN GM2	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C51cc02 - 8051 Code
GMM 5115	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C5115 - 8051 Code
GMM 876	●	-	-	-	-	-	-	●	-	-	-	-	Microchip PIC16F876A - PIC 14 Code
GMM 932	●	-	-	-	-	●	-	-	-	-	-	-	PHILIPS P89LPC932 - 8051 Code
GMM AC2	●	-	-	-	-	●	-	-	-	-	●	-	Atmel T89C51AC2 - 8051 Code
GMM AM08	●	-	-	-	-	-	●	-	-	-	-	-	Atmel ATmega08 - AVR Code
GMM AM32	●	-	-	-	-	-	●	-	-	-	-	-	Atmel ATmega32 - AVR Code
GMB HR84	●	-	-	-	-	●	●	●	-	-	●	-	Mini Block 8 input opto 4 output relè
GMB HR168	●	-	-	-	-	●	-	-	-	-	●	-	Mini Block 16 input opto 8 output relè




FIGURE 14: EXAMPLES TABLE

## B) FLASH REPROGRAMMING

Mini Module programming is performed using the Boot Loader preprogrammed in the Flash of Mini Module itself.

Such program allows to create a communication between Mini Module and the PC, and to use it to send a .hex file to code memory and/or EEPROM memory area of microcontroller.

Boot Loader does not allow to program configuration bits and security fuses of microcontroller, this can be done using ISP programming (please refer to appendix B of this manual).

The portion of Flash memory taken by the Boot Loader is 2 KBytes, this means that the last block of 1 KWord at the end of the memory is reserved to Boot Loader and is not available for user application any more.

Boot Loader communicates through PC serial port installing a specific utility (that can be downloaded for free from our website [www.grifo.com](http://www.grifo.com) or can be found in our CD) called AVRBootloaderGrifo.exe. This utility allows to select the serial port to use and to select the files to program memories of Mini Module.

For some applications, use of an ISP programmer may be the only chance. Please refer to appendix B for more information about ISP programming.

Combo box called "Com Port" allows to select the serial port to use for communication with Mini Module.

Checking the checkbox "Application Code" the file whose name is written in the text box on the right will be stored in Flash of Mini Module. To choose the file press the "Browse" button.

Checkbox "EEPROM Code" performs the same operation described above, but writes to EEPROM.

There is also the possibility to use AVRBootloaderGrifo from DOS window, command prompt, a command line or as an external tool of an IDE.

Option for command line are:

/com1 , /com2 , ... , /com9	Number of serial port used for communication. Serial ports from COM1 to COM9 are supported
/f <FLASHProgram>	Indicates the complete pathname of the .HEX file to write in microcontroller FLASH memory
/e <EEPROMProgram>	Indicates the complete pathname of the .HEX file to write in microcontroller EEPROM memory

For example:

AVRBootloaderGrifo.exe /com2 /f C:\Projects\MotorControl\Main.hex

opens a connection on serial port COM2 to the Boot Loader and sends the file Main.hex located in folder C:\Projects\MotorControl\.

To integrate AVRBootloaderGrifo.exe in a IDE, for example the one of BASCOM AVR (described at point C) the user must:

- 1) Open the window of menu Options | Programmer
  - 2) In the text box Programmer choose "External Programmer"
  - 3) Click the tab Other
  - 4) In text box Program insert the complete pathname of AVRBootloaderGrifo.exe pressing Browse
  - 5) In text box Parameters insert command line parameter for AVRBootloaderGrifo
- Please remaind that the string {file} in the above mentioned text box is automatically replaces by BASCOM AVR with the name of .hex file just generated.
- Please refer to BASCOM AVR documentation for further information.

- B1) Find on CD **grifo®** and save to a comfortable position on your hard drive the demo program "d\_AM32u.hex". It can be found starting from main page following the path: English | Examples tables | Mini Modules and Mini Block examples | GMM AM32 (please refer to figure 14). You may want to remove the read-only attribute.
- B2) Connect the Mini Module to PC serial port making the connection described at point A. Close the terminal emulator.
- B3) Indicate in combo box "Com Port" the serial port connected at the previous point.
- B4) Check the check box "Application Code" and press the button "Browse" on the right, then select the file previously saved at point B1.
- B5) Uncheck the check box "EEPROM Code", if checked.
- B6) Press the button "Synch to Bootloader..." or the key combination Alt+S on the PC, then reset the Mini Module or turn off and then on its supply. The file is downloaded into the Mini Module. If this does not happen, and the program should indicate a "No response from target bootloader", try to repeat the operation decreasing the time between pressure of button on the PC and reset of Mini Module. If the problem persists, check cable and connection.
- B7) When operation is completed the program reports its status. In case of problems, check cable and connection.
- B8) Start the terminal emulator configured like in point A2 and verify that the application program just downloaded is executed in internal Flash.

**FIGURE 15: BOOT LOADER UTILITY**

## C) GENERATING DEMO EXECUTABLE CODE

- C1) Install on the hard disk of the development P.C. the software environment selected to develop the application program. As described in the chapter SOFTWARE DESCRIPTION there are many different software tools that satisfy any customers requirements but here we remind only the most diffused as the BASCOM AVR, ICC AVR, etc.
- C2) On **grifo®** CD in addition to file with the executable code of the demo program, described at point B2, there are also the source files of the same. These have an extension that identifies the used software development tools (for example d\_AM32u.bas for BASCOM AVR or d\_AM32u.c for ICC AVR) and they are properly organized inside demo programs tables available on CD, together with possible definition file (for example: d\_AM32u.prj for ICC AVR). Once these files have been located they must be copied in a comfortable folder on the hard disk of development PC.
- C3) Compile the source file by using the selected software tools: the file d\_AM32u.hex must be obtained equal to those available on **grifo®** CD and already used at points B. This operation is very different according to the programming environment selected, so here follow the details:

### C3 Bascom AVR) Ricompilation using BASCOM AVR.

C3 Bascom AVR a) In BASCOM IDE, load the program source with menu File | Open:

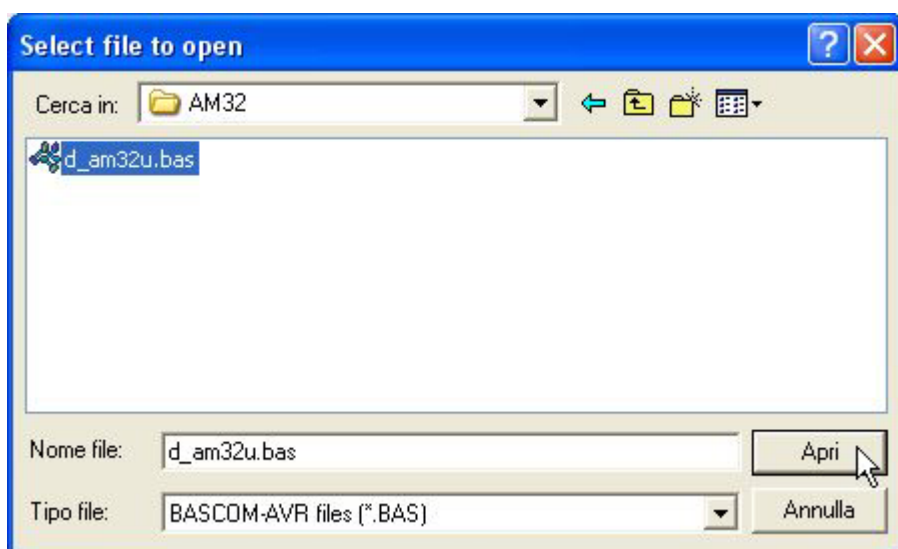


FIGURE 16: LOADING A SOURCE FILE WITH BASCOM AVR



C3 Bascom AVR b) From menu Options | Compiler | Chip set the value 64 for HW Stack, 32 for Soft Stack, 64 for Framesize, as suggested also in the source code, and press OK. Such values must be considered minimal and must be increased if required:

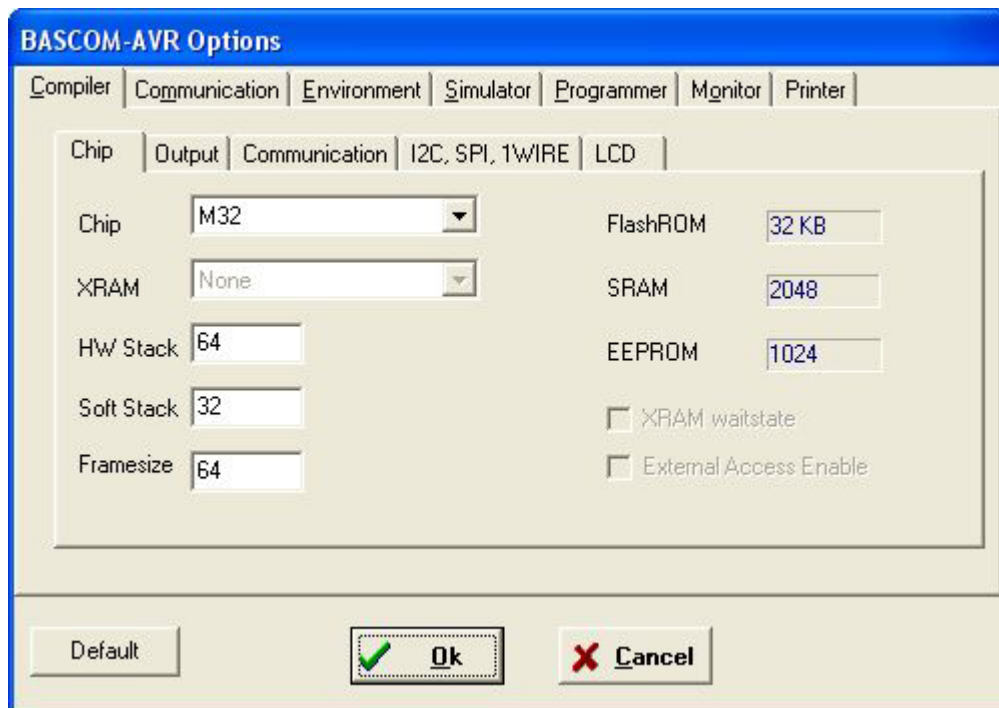


FIGURE 17: CONFIGURATION OF COMPILER BASCOM AVR

C3 Bascom AVR c) Compile the source file by pressing the button with the icon of an integrated circuit.

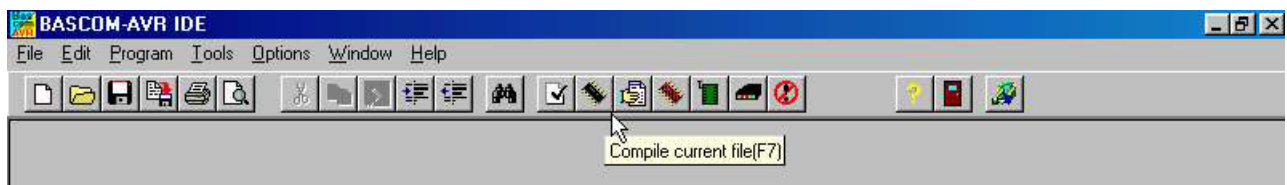


FIGURE 18: COMPILATION WITH BASCOM AVR

## C3 ICC AVR) Ricompilation with ICC AVR.

C3 ICC AVR a) In standard editor, load the project file using the menu Project | Open...:

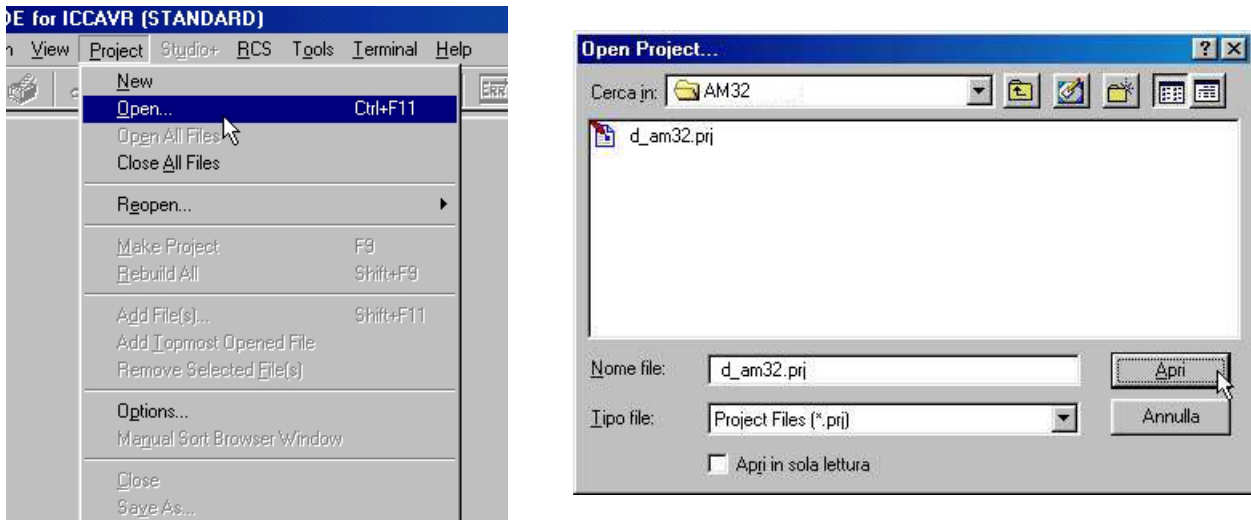


FIGURE 19: LOADING PROJECT FILE WITH ICC AVR

C3 ICC AVR b) Compile the project using the menu Project | Make project:

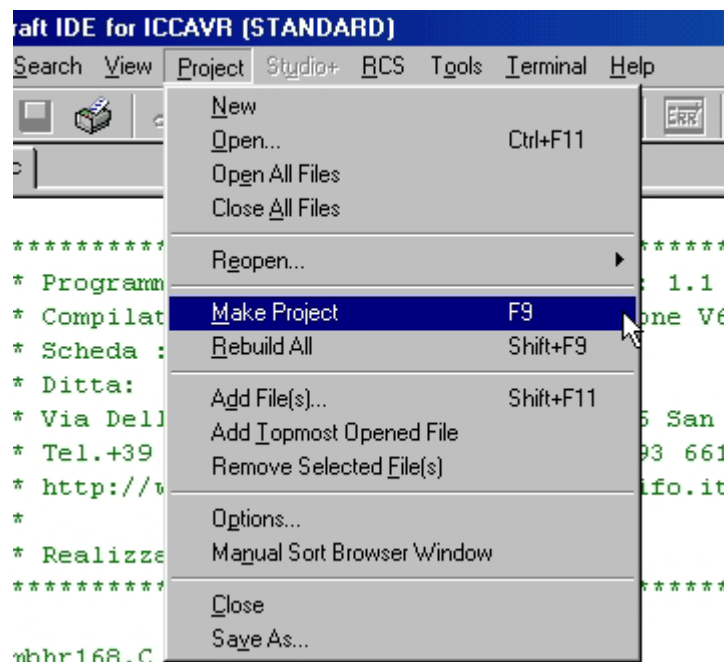


FIGURE 20: COMPILATION WITH ICC AVR



C4) Program the compiled file into FLASH memory of **GMM AM32** repeating the steps of point B.

## D) *FINAL APPLICATION*

D1) Close the Boot Loader PC utility.

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®** technician.

Instead when execution of all the steps above described is right, the user has realized his first application program that coincides with demo of **GMM AM32**.

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 (successive to B and C) in cyclic mode, until the developed application program is completely well running.

When this focus is reached the development PC can be eliminated.

Remember to reconfigure USART of Mini Module, if required.



## SOFTWARE DESCRIPTION

A wide selection of software development tools can be obtained, allowing use of the module as a system for its own development, both in assembler and in other high level languages; in this way the user can easily develop all the requested application programs in a very short time.

Generally all software packages available for the mounted microprocessor, or for the AVR family, can be used.

Software packages purchased from **grifo**<sup>®</sup> are always provided with example programs that show how to use each section of the board and a complete use documentation.

Remarkable are:

### **BASCOM AVR**

It is a powerfull new integrated development environment for AVR microcontroller. The toolset incorporates an editor, optimising BASIC compiler, assembler and HEX creator. The BASIC compiler produces very tight AVR machine code by virtue of the fact it translates the BASIC source into actually run timeassembly code wich is optimised to run as fast as possible. The target AVR microcontroller therefore runs true assembly code rather than tokenised code wich is found in many other BASIC compilers. It is also provided with integrated simulator for source level debugging and optional external libraries to drive or simulate several external devices (likebadge readers, PS/2 keyboards, graphic and alphanumeric displays, etc.).

### **ICC AVR PRO**

Cross compiler for C source program. It is a powerfull software tool that includes editor, ANSI C compiler, assembler, linker, library management program and project manager included in an easy to use integrated development environment for Windows and other P.C. operating systems. Library sources, floating point, integration with AVR studio, on line help and ANSI terminal emulator for target communication are provided too.

### **DDS MICRO C AVR**

Low cost ross compiler for C source program. It is a powerful software tool that includes editor, C compiler (integer), assembler, optimizer, source linker and library in one easy to use integrated development environment. There are also included the library sources and many utilities programs. The default IDE can be replaced by a new one named **Micro IDE**, that is more powerfull, for Windows operating system and provided of many utility functions.

### **AVR Studio**

It is a development tool for AVR family of microcontroller that fully control execution of program on AVR in circuit emulator or on the built in AVR instruction set simulator. AVR Studio supports source level execution of assembly and C programs generated by external compilers and assemblers. The tolls is based on a set of windows for source, watch, registers, memory, peripherals, message and processor that enable the user to have full control of the status of every element in the execution target. It also features an "application builder" to easy the generation of code to initialise all hardware peripherals (USART, SPI, Port, ADC, ect.) starting from a graphic interface.

There is also the remarkable possibility to drive the JTAG interface called "JTAG ICE" manufactured by Atmel.

A JTAG interface allows to enter the core of microcontroller to examine its status during execution of application program directly on the application hardware.

The user can insert both hardware and software breakpoints, and when execution is stopped the values contained in memory and internal registers can be examined.

JTAG interface also allows to reprogram the microcontroller memories.

Using JTAG interface several debugging problems are solved, increasing the possibility to eliminate bugs and drastically reducing the time required to obtain the final application completely debugged.



**FIGURE 21: TOP VIEW OF GMM AM32**

## 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 microprocessor internal peripheral devices 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.

### ACTIVITY LEDS

The **GMM AM32** allows software management of activity or status LEDs LD1 and LD2, through an I/O line of the microcontroller, with the following correspondence:

<b>PD4 = 0</b>	->	<b>LD1 ON</b>
<b>PD4 = 1</b>	->	<b>LD1 OFF</b>
<b>PB7 = 0</b>	->	<b>LD2 ON</b>
<b>PB7 = 1</b>	->	<b>LD2 OFF</b>

Please remind that PD4 and PB7 are available on CN1 respectively at pins 5 and 18.

Also, pin 5 of CN1 can be connected to GND by setting DSW1.7 ON, turning ON LED LD1.

In this case microcontroller digital I/O signal PD4 must be configured as input, to avoid conflicts. The signals PD4 and PB7 are set high after reset or power on, so during these phases LEDs are OFF or disabled.

### DIP SWITCH

**GMM AM32** allows to read by software the status of dip switch, DSW1.7, through two I/O signals of microcontroller:

<b>DSW1.7 ON</b>	->	<b>PD4 is 0</b>
<b>DSW1.7 OFF</b>	->	<b>PD4 is 1</b>

Please remark that microcontroller digital I/O signal PD4 controls also LED LD1, so closing DSW1.7 will turn LD1 ON.

Signal PD4 is available on CN1 respectively at pins 5.

## **BACKED SRAM + SERIAL RTC**

Mini Module **GMM AM32** is provided with a complete Real Time Clock capable to manage hours, minutes, seconds, day, month, year and weekday in complete autonomy.

This module can be backed up through the back up circuitry to warrant data validity in any working condition and is completely managed by software.

RTC section can also generate periodic interrupts at software programmable time intervals, to switch the CPU out of normal operations or to awaken it from low consumption modes.

For software management of serial SRAM + RTC backed module, please refer to specific manufacturer documentation. This manual reports no software information because management of this component is complex and requires a deep knowledge, anyway the user can use the demo programs supplied with the card. The board control logic allows to realize a serial communication with I<sup>2</sup>C bus standard protocol, through two I/O microprocessor pins. The only necessary information is the electric connection:

SDA , PC1 (input/output)	->	DATA line (SDA)
SCL , PC0 (input/output)	->	CLOCK line (SCL)

Please remark that **A0** of this component's slave address is bound to logic 0.

This means that RTC takes permanently the slave address **A0**, which is not available for user application.

The above mentioned module is capable to generate a programmable periodic signal, which is connected on-board to a 4.7 k $\Omega$  pull-up, and is available on pin 11 of Mini Module connector CN1, and is connected to signal PD4 of microcontroller.

## **CPU INTERNAL PERIPHERALS**

Registers description and purpose for all internal peripherals (Analog COMPARATORS, A/D CONVERTER, Timer Counters, USART, I<sup>2</sup>C BUS, SPI, OCM, etc.) is available in the proper data sheet and user manual of the manufacturer.

Please refer to chapter BIBLIOGRAPHY and to appendix A of this manual to easily locate such documentation.

## BIBLIOGRAPHY

In this chapter there is a complete list of technical books and notes, where the user can find all the necessary documentations on the components mounted on **GMM AM32** Mini Module.

Technical paper MAXIM: *True RS 232 Transceivers*

Manual PHILIPS: *I2C-bus compatible ICs*

Technical paper ATMEL: *Data Sheet Atmel ATmega32L*

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



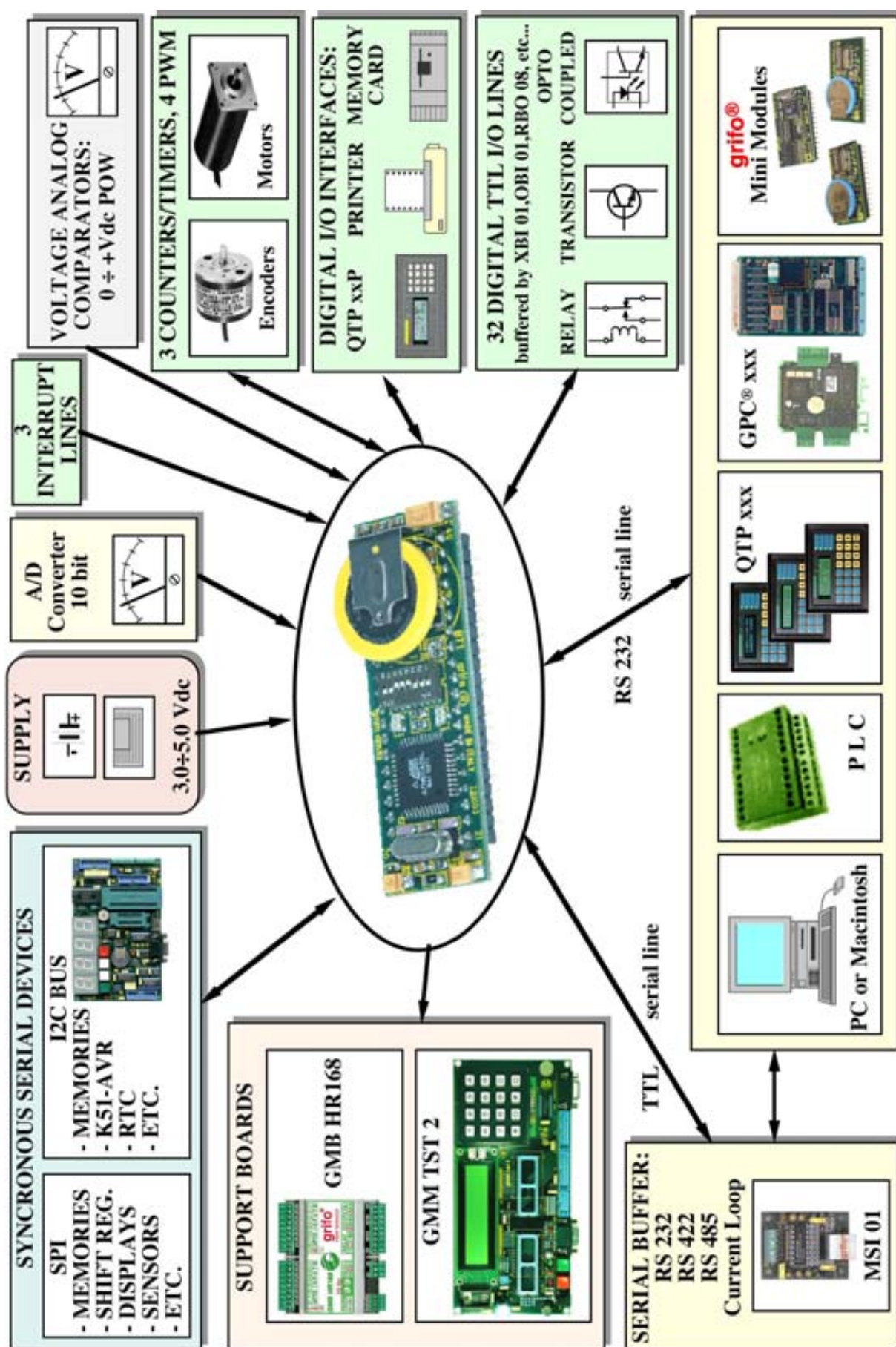


FIGURE 22: AVAILABLE CONNECTIONS DIAGRAM





## APPENDIX A: DATA SHEETS

**grifo®** provides a completely free technical documentation service to make available data sheets of on board components, through its web site. In this chapter the user found the complete and ready to use links and URLs to these information, together with the first pages of the same documents. To use our technical documentation service just connect to our site [www.grifo.com](http://www.grifo.com) and click its icon.

### ATmega32L

Link: Home | Technical documentation Service | ATMEL | Data-Sheet ATmega32

URL: <http://www.grifo.com/PRESS/DOC/Atmel/ATmega32.pdf>

#### Features

- High-performance, Low-power AVR® 8-bit Microcontroller
- Advanced RISC Architecture
  - 131 Powerful Instructions – Most Single-clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 16 MIPS Throughput at 16 MHz
  - On-chip 2-cycle Multiplier
- Nonvolatile Program and Data Memories
  - 32K Bytes of In-System Self-Programmable Flash
    - Endurance: 10,000 Write/Erase Cycles
  - Optional Boot Code Section with Independent Lock Bits
    - In-System Programming by On-chip Boot Program
    - True Read-While-Write Operation
  - 1024 Bytes EEPROM
    - Endurance: 100,000 Write/Erase Cycles
  - 2K Byte Internal SRAM
  - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Extensive On-chip Debug Support
  - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Four PWM Channels
  - 8-channel, 10-bit ADC
    - 8 Single-ended Channels
    - 7 Differential Channels in TQFP Package Only
    - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
  - Byte-oriented Two-wire Serial Interface
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
  - 32 Programmable I/O Lines
  - 40-pin PDIP, 44-lead TQFP, and 44-pad MLF
- Operating Voltages
  - 2.7 - 5.5V for ATmega32L
  - 4.5 - 5.5V for ATmega32
- Speed Grades
  - 0 - 8 MHz for ATmega32L
  - 0 - 16 MHz for ATmega32
- Power Consumption at 1 MHz, 3V, 25°C for ATmega32L
  - Active: 1.1 mA
  - Idle Mode: 0.35 mA
  - Power-down Mode: < 1 µA



**8-bit AVR®  
Microcontroller  
with 32K Bytes  
In-System  
Programmable  
Flash**

**ATmega32  
ATmega32L**

**Preliminary**

Rev. 2503D-AVR-02/03



## PCF 8583

Link: Home | Technical documentation Service | I<sup>2</sup>C BUS | Data-Sheet PCF8583  
 URL: <http://www.grifo.com/PRESS/DOC/PHILIPS/PCF8583.pdf>

Product specification

Product specification

### PCF8583

#### Block/Calendar with 240 × 8-bit RAM

#### BLOCK DIAGRAM

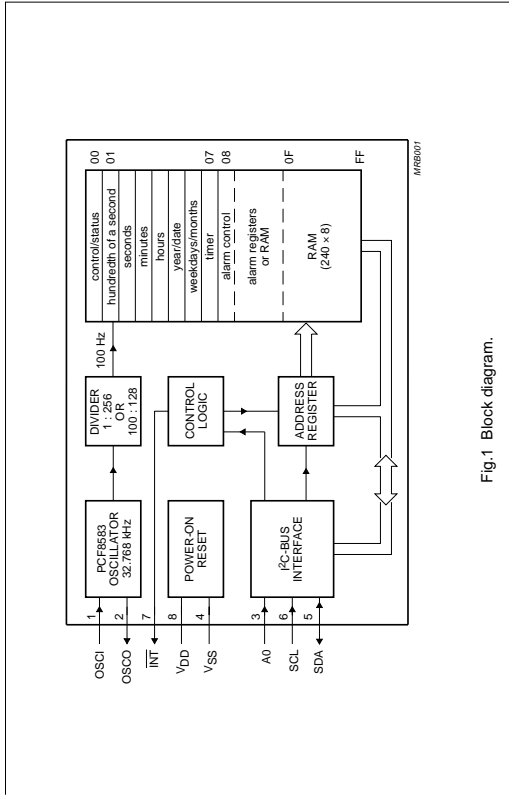


Fig.1 Block diagram.

#### PINNING

SYMBOL	PIN	DESCRIPTION
SCL	1	oscillator input, 50 Hz or event-pulse input
SC0	2	oscillator output
INT	3	address input
VSS	4	negative supply
VDD	5	serial data line
SCL	6	serial clock line
INT	7	open drain interrupt output (active LOW)
VDD	8	positive supply

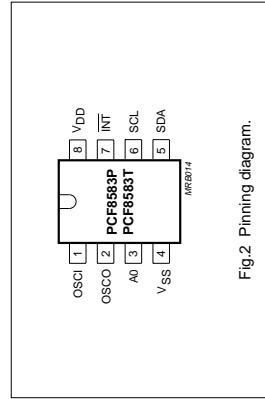


Fig.2 Pinning diagram.

#### 2 GENERAL DESCRIPTION

The PCF8583 is a clock/calendar circuit based on a 2048-bit static CMOS RAM organized as 256 words by 8 bits. Addresses and data are transferred serially via the two-line bidirectional I<sup>2</sup>C-bus. The built-in word address register is incremented automatically after each written or read data byte. Address pin A0 is used for programming the hardware address, allowing the connection of two devices to the bus without additional hardware.

The built-in 32.768 kHz oscillator circuit and the first 8 bytes of the RAM are used for the clock/calendar and counter functions. The next 8 bytes may be programmed as alarm registers or used as free RAM space. The remaining 240 bytes are free RAM locations.

#### FEATURES

- I<sup>2</sup>C-bus interface operating supply voltage: 2.5 V to 6 V
- clock operating supply voltage (0 to +70 °C): 2.0 V to 6.0 V
- 40 × 8-bit low-voltage RAM
- data retention voltage: 1.0 V to 6 V
- operating current (at f<sub>SCL</sub> = 0 Hz): max. 50 µA
- clock function with four year calendar
- universal timer with alarm and overflow indication 4 or 12 hour format
- 2.768 kHz or 50 Hz time base
- serial input/output bus (I<sup>2</sup>C)
- automatic word address incrementing
- programmable alarm, timer and interrupt function
- slave address:
  - READ: A1 or A3
  - WRITE: A0 or A2.

#### QUICK REFERENCE DATA

SYMBOL	PARAMETER	CONDITION	MIN.	TYP.	MAX.	UNIT
V <sub>DD</sub>	supply voltage operating mode	I <sup>2</sup> C-bus active	2.5	—	6.0	V
V <sub>DD</sub>	supply voltage operating mode	I <sup>2</sup> C-bus inactive	1.0	—	6.0	V
I <sub>DD</sub>	supply current operating mode	f <sub>SCL</sub> = 100 kHz	—	—	200	µA
I <sub>DD</sub>	supply current clock mode	f <sub>SCL</sub> = 0 Hz; V <sub>DD</sub> = 5 V	—	10	50	µA
I <sub>DD</sub>	supply current clock mode	f <sub>SCL</sub> = 0 Hz; V <sub>DD</sub> = 1 V	—	2	10	µA
T <sub>OP</sub>	operating ambient temperature range		-40	—	+85	°C
T <sub>STG</sub>	storage temperature range		-65	—	+150	°C

#### ORDERING INFORMATION

TYPE	NAME	DESCRIPTION	VERSION
PCF8583P	DIP8	plastic dual in-line package; 8 leads (300 mil)	SOT197-1
PCF8583T	SO8	plastic small outline package; 8 leads; body width 7.5 mm	SOT176-1

## APPENDIX B: ISP PROGRAMMING WITH GMM TST 2

In this appendix the user can find specific instructions to program Mini Module **GMM AM32** installed on a **GMM TST 2** using an external ISP programmer.

This method to program Mini Module on board Flash and EEPROM can be considered an alternative to the one explained in section "B" of chapter "How to start" of manual.

Content of other sections in above mentioned chapter are still valid anyway.

ISP programming allows also to change the value of configuration bits and security fuses, in addition to the programming of whole Flash and EEPROM content.

The Boot Loader does not allow to program the whole content of Flash, because the Boot Loader itself resides in a Flash area which becomes protected.

Also, configuration bits, that enable the Boot Loader itself, and security fuses are out of the range programmable by the Boot Loader and must be programmed in ISP mode.

If you think it is more convenient to use a Boot Loader on RS 232 serial line, refer to above mentioned chapter.

For more information on how to connect a PC serial port to **GMM TST 2** (or even to Mini Module as stand-alone) to test the program, refer to section "A" of chapter "How to start" of manual.

For further information on how to develop and debug the firmware, refer to section "C" of chapter "How to start" of manual.

Programmers that can be used with **GMM TST 2** are:

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

Information here reported are completed with direct connection diagrams (at page B-3 and B-5) that allow the user to build an ISP support hardware or to embed ISP support in user application.

### A) COMMON OPERATIONS FOR ALL PROGRAMMERS

A1) Programming of Mini Module on a **GMM TST 2** board is performed through a specific connector of the card and, if required, an interface adapter between the programmer and the connector itself.

The two **grifo®** programmers that can be interfaced with **GMM TST 2**, that is **UEP 49** and **MP AVR-51+**, use a specific interface for each one of them, whose diagram is shown in figures B-1 and B-2. On the left side of the figures there is the list of **GMM TST 2** CN7 connector's ways that must be connected to corresponding pins on programmer's connector indicated by the arrow.

Programmers AVR ISP and Equinox EPSILON5 do not require any specific interface.

A2) Insert **GMM AM32** in one of the 40 ways socket of **GMM TST 2** aligned to the bottom, as indicated by serigraph and shown in figure 12 of manual.

A3) Locate on **grifo®** CD and save to a comfortable position on the hard disk of the PC the file called "d\_am32u.hex" following the path: English | Example Tables | Mini Module and Mini Block examples | GMM AM32 (please refer to figure 14 of manual).

Next sections described a detailed configuration specific for each programmer.

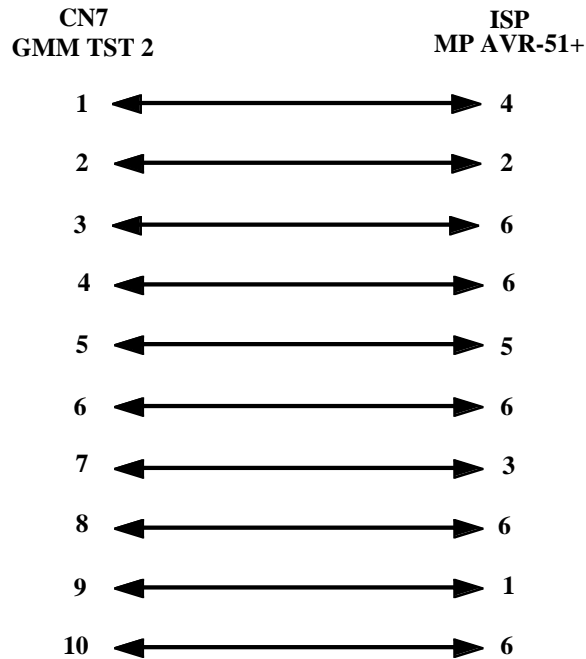


FIGURE B-1: DIAGRAM OF INTERFACE BETWEEN GMM TST 2 AND MP AVR-51+

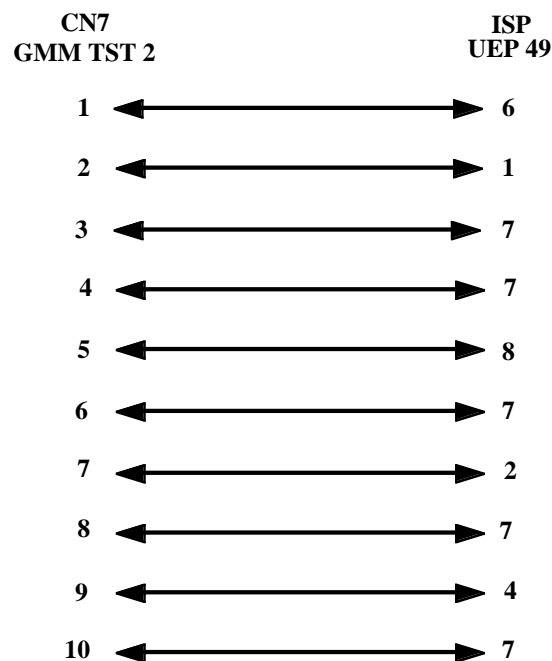


FIGURE B-2: DIAGRAM OF INTERFACE BETWEEN GMM TST 2 AND UEP 49

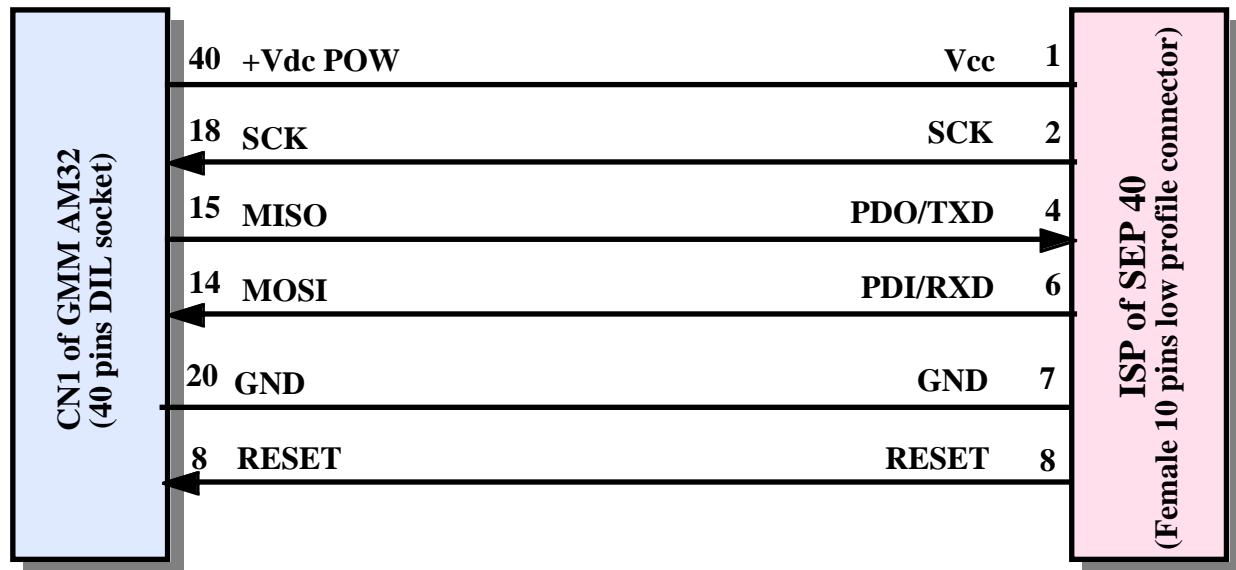


FIGURE B-3: DIRECT CONNECTION BETWEEN GMM AM32 AND SEP 40

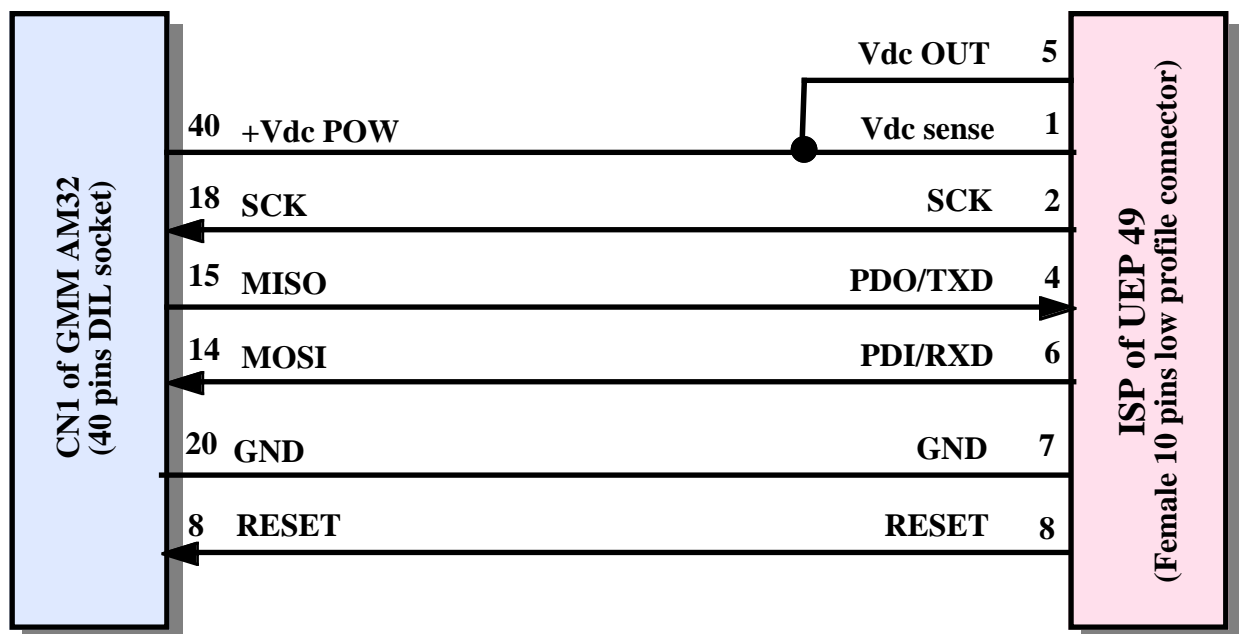


FIGURE B-4: DIRECT CONNECTION BETWEEN GMM AM32 AND UEP 49

## B) PROGRAMMING USING grifo® MP AVR-51+ ; grifo® UEP 49

B1) Instructions for installing, connecting to PC and using the programmer are delivered with the programmer itself. The control program for all grifo® programmers is called PG4UW and will be indicated by that name from now on.

B2) Connect the programmer to **GMM TST 2** using the interface described at point A1.

B3) Insert module in socket Z1 or Z2 of **GMM TST 2** using the interface and the instructions described at point A2, then turn on power supply of **GMM TST 2** and configure it as if a AVR ISP programmer should be used (see the manual of **GMM TST 2**).

B4) Select ATmega32L ISP as the component to program in specific menu of PG4UW, as indicated in figure B-5.

B5) Load the file previously saved on point A3 by pressing the button "Load".

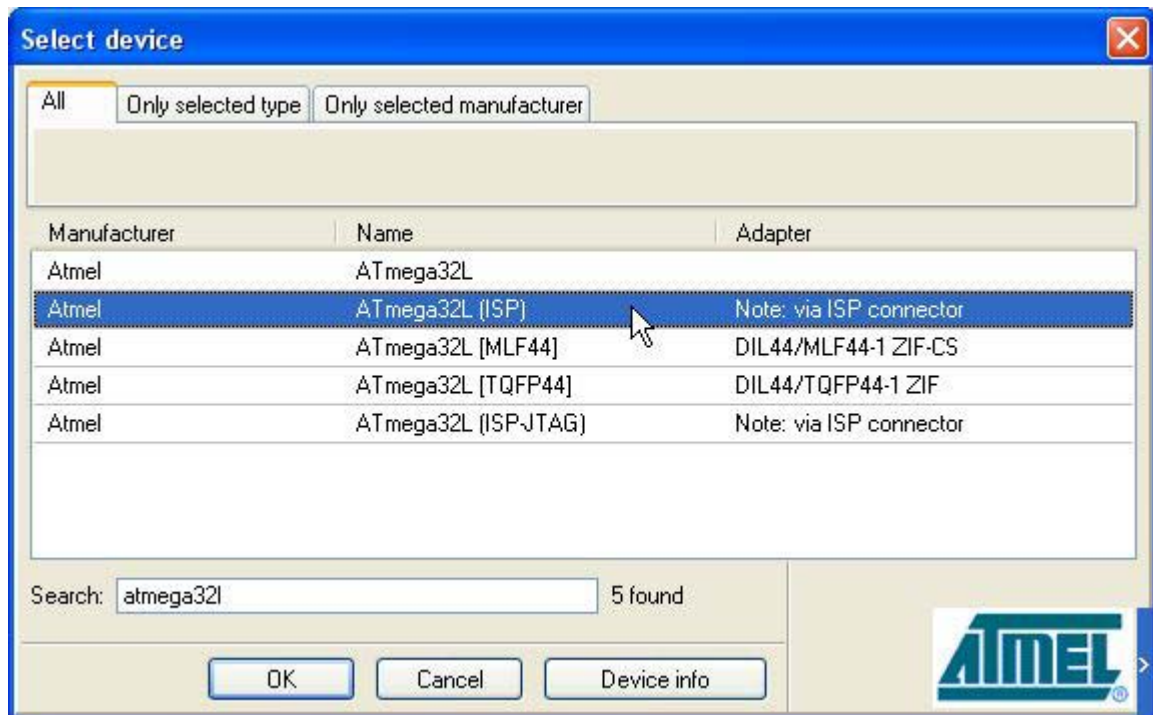
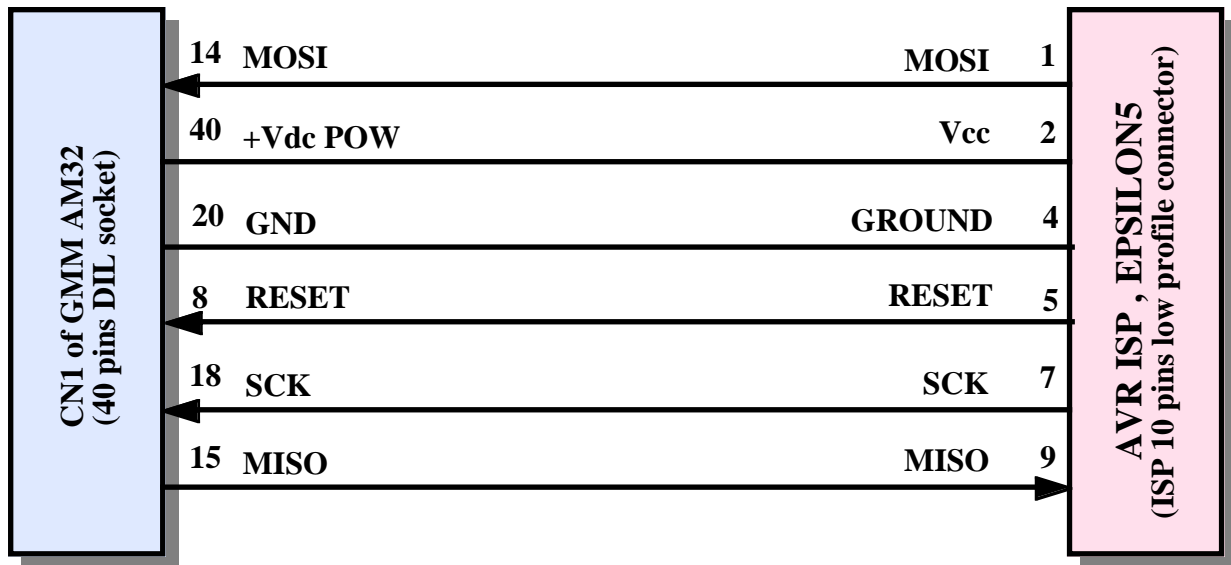
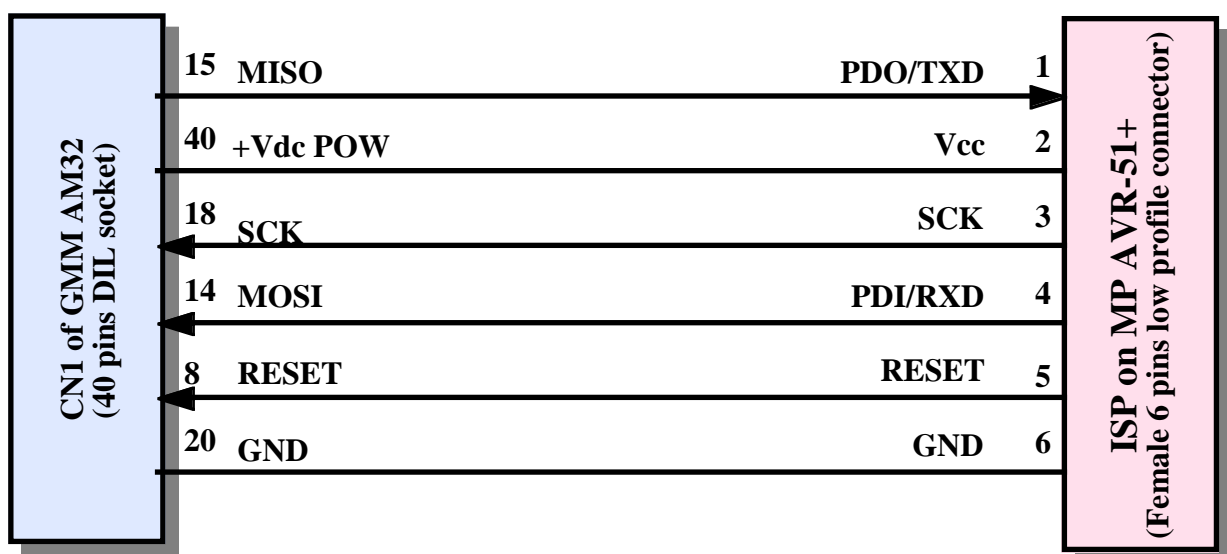


FIGURE B-5: COMPONENT SELECTION USING PG4UW

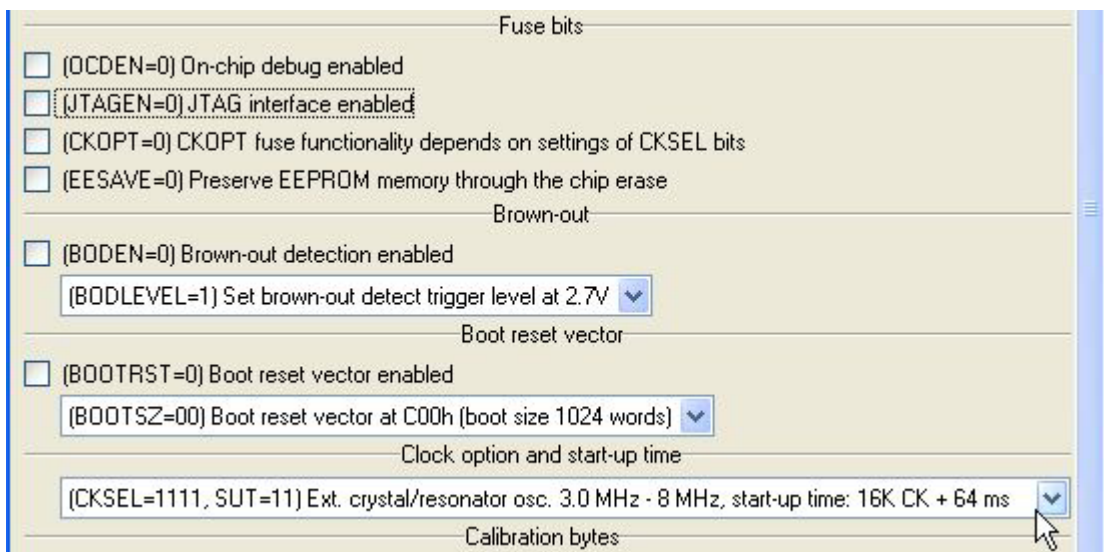


**FIGURE B-6: DIRECT CONNECTION BETWEEN GMM AM32 AND EPSILON5 OR AVR ISP**

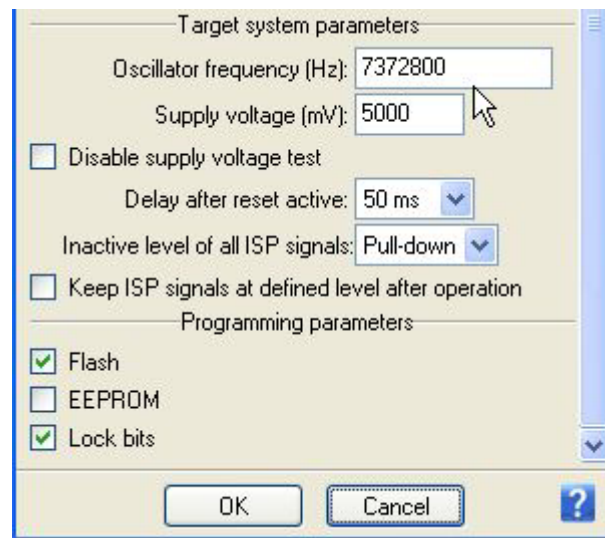


**FIGURE B-7: DIRECT CONNECTION BETWEEN GMM AM32 AND MP AVR-51+**

- B6) Configure the component using the menu Device options | View/Edit options and security. The component must be configured as indicated in figure B-8, that is the configuration required to make the demo program d\_am32u.hex work. Quartz and start-up time options must be configured like in the figure. Of course, the user can perform any other configuration required by his/her application.
- B7) Configure the programmer using the menu Device options | Operating options. Programmer must be configured as indicated in figure B-9.
- B8) Start the programming operation, pressing the "Program" button.
- B9) During the programming operation, a progress bar advances until the operation is completed. Eventual errors are promptly indicated.



**FIGURE B-8: COMPONENT CONFIGURATION USING PG4UW**



**FIGURE B-9: PROGRAMMER CONFIGURATION USING PG4UW**



## C) PROGRAMMING USING EQUINOX EPSILON5

C1) Programming of Mini Module on a **GMM TST 2** board is performed through a specific connector of the card. First of all, programmer Equinox EPSILON5 must be physically configured to connect to **GMM TST 2**. To perform such configuration:

- Connect the 10 ways flat cable to connector "J7-ATMEL10" of EPSILON5
- Connect jumper J9 of EPSILON5

for further information please refer to manual of EPSILON5.

C2) After configuring and closing the EPSILON5, its control program must be installed. This program's name is EQTools, and will be called this way from now on. For further information about installing the program and connecting EPSILON5 to the PC, please refer to manual of EPSILON5.

C3) Connect the programmer to connector CN7 of **GMM TST 2**.

C4) Insert module in socket Z1 or Z2 of **GMM TST 2** using the interface and the instructions described at points A2 and A3, then turn on power supply of **GMM TST 2** as described in the manual of **GMM TST 2**.

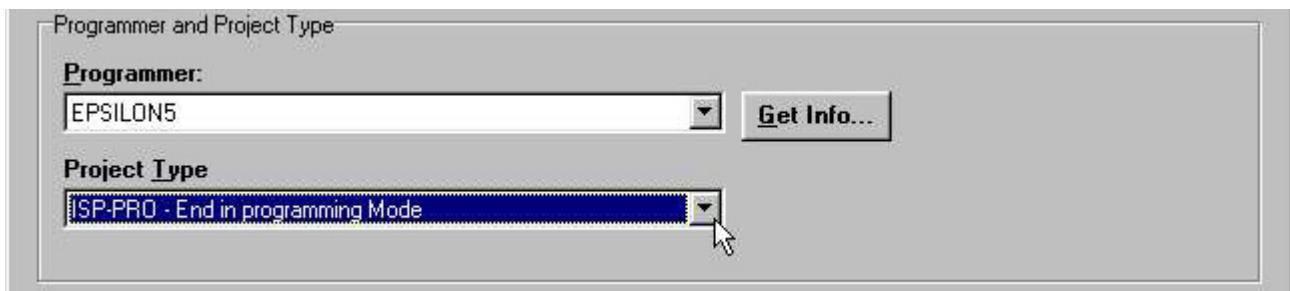
C5) To use Equinox EPSILON5 connected to the PC (instead of stand-alone), it is required to open a project file (extension EDS). This can be done creating a new one from start screen of EQTools, using specific menus and buttons or loading and existing project file. For further information about project file management, please refer to manuale of EQTools.

C6) Who creates a new project, must be sure to perform the settings shown in figure B-10. Who opens an existing project must be sure that these settings have already been performed. These settings assure that:

- The project is programmed correctly by EPSILON5
- Target device is ATmega32L
- File programmed on the target is d\_am32u.hex
- Target component configuration is correct

for further information about configuring a project, please refer to EQTools documentation.

- C7) Who creates a new project, must also indicate to use it as "Test EDS" mode. To enable and perform Flash write operation the user must move back to Flash menu, put the check sign on "Edit Menu" checkbox and press button "Write".
- C8) Window "Write Block to Flash" shows a summar of some current settings to verify them. If are correct, pressing OK starts the memory writing procedure.
- C9) Current status of programming is indicated by a progress bar, when programming completes a message indicates the operation final result.
- C10) To perform configuration bits write operation it is required to move back to Fuses menu, and press button "Write" in frame "Target Fuses".



Fuses	PC Fuse State	Target Fuse State	Description
CKSEL0	1 - UNPROGRAMMED	?	Clock Select 0
CKSEL1	1 - UNPROGRAMMED	?	Clock Select 1
CKSEL2	1 - UNPROGRAMMED	?	Clock Select 2
CKSEL3	1 - UNPROGRAMMED	?	Clock Select 3
SUT0	1 - UNPROGRAMMED	?	Oscillator - Select Start Up Time 0
SUT1	1 - UNPROGRAMMED	?	Oscillator - Select Start Up Time 1
BODEN	1 - UNPROGRAMMED	?	Brown-Out Detector - ENABLE

FIGURE B-10: PROJECT CONFIGURATION USING EQTOOLS

## D) *PROGRAMMING USING ATMEL AVR ISP*

- D1) Control program of **AVR ISP** is **AVR STUDIO**, version 4 or greater. Latest version can be downloaded from Atmel website [www.atmel.com](http://www.atmel.com). You may download it and install it following the instructions on screen.
- D2) Configure **AVR ISP** to use the 10 ways flat cable and connect it to connector CN7 of **GMM TST 2**, connect **AVR ISP** to PC serial port (please refer to instructions at points A2 and A3), configure **GMM TST 2** to program through **AVR ISP** and supply it (please refer to **GMM TST 2** manual).
- D3) Run **AVR STUDIO**. AVR ISP control program can be run by pressing the button with AVR chip as icon.
- D4) Select as CPU ATmega32L, like in figure B-11.
- D5) Load the file previously saved on point A3 by pressing the button "Load".
- D6) Configure CPU as indicated in images of figure B-12.
- D7) Configure the programmer to check signature, erase device and reprogram with verify Flash memory and configuration bits, like indicated in figure B-13.
- D8) Perform the programming sequence by pressing button "Start" indicated in figure B-13.

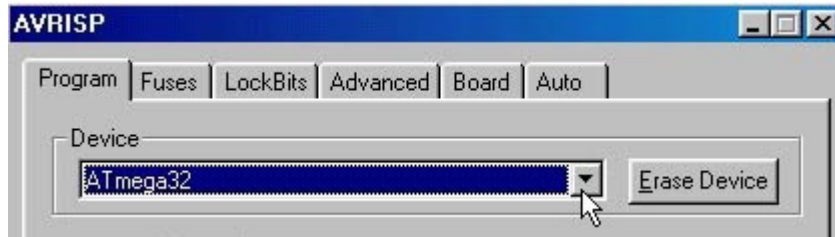


FIGURE B-11: CPU SELECTION USING AVR STUDIO

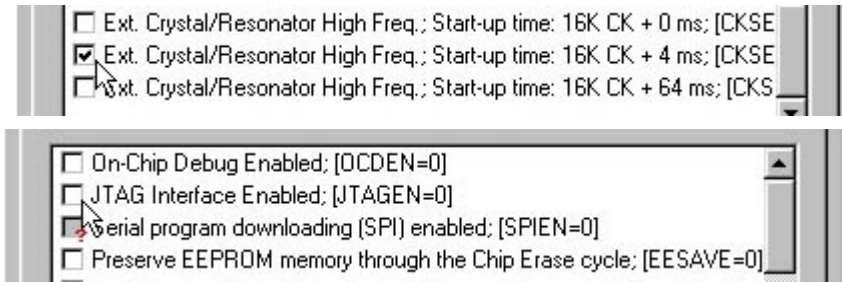


FIGURE B-12: CPU CONFIGURATION USING AVR STUDIO

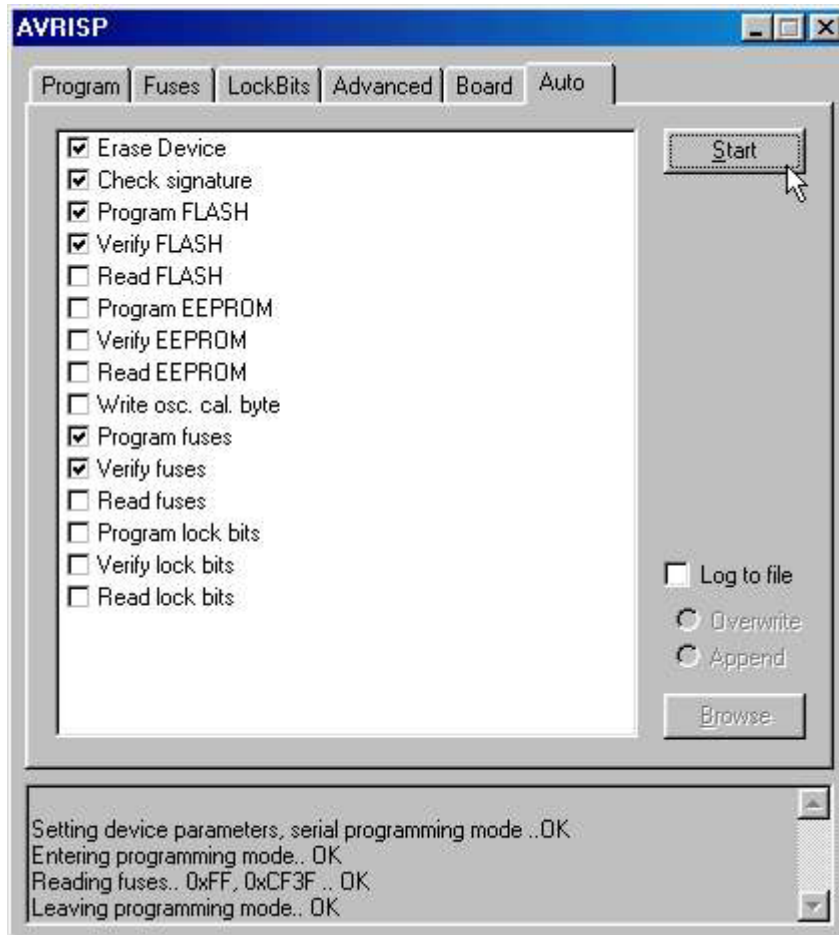


FIGURE B-13: AVR ISP CONFIGURATION USING AVR STUDIO

## D) PROGRAMMING USING PONY PROG

- E1) Pony Prog is a software that allows to program **GMM AM32** on a **GMM TST 2** simply connecting PC serial port to connector CN6. Version 2.06c supports Atmel ATmega32L, you may download it from [www.lancos.com](http://www.lancos.com) and install it following the instructions on screen.
- E2) Connect CN6 of **GMM TST 2** to PC serial port, configure **GMM TST 2** to program through **Pony Prog** and supply it (please refer to **GMM TST 2** manual).
- E3) Run **Pony Prog** and perform calibration through menu Setp | Calibration.
- E4) Select communication library SI Prog API through menu Setup | Communication setup.
- E5) Select "AVR micro" and "ATmega32" from the specific list boxes (see figure B-15).
- E6) Open the file "d\_am32u.hex" previously saved:
- E7) Configure the CPU to keep the EEPROM content while erasing and use an high frequency external quartz, as indicated in figure B-16.
- E8) Configure the programmer to perform ID check, erase the device and reprogram with verify FLASH and configuration bits.
- E9) Perform the programming pressing the button indicated in figure B-17 and be sure that the message "Program Succesful" appears at the end of the programming phases (which may last up to tenth of seconds).



FIGURE B-14: LOGO OF PONY PROG

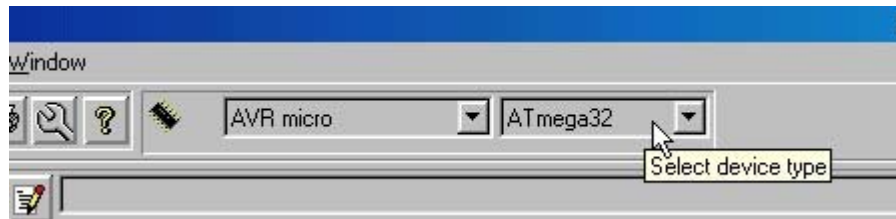


FIGURE B-15: CPU SELECTION WITH PONY PROG

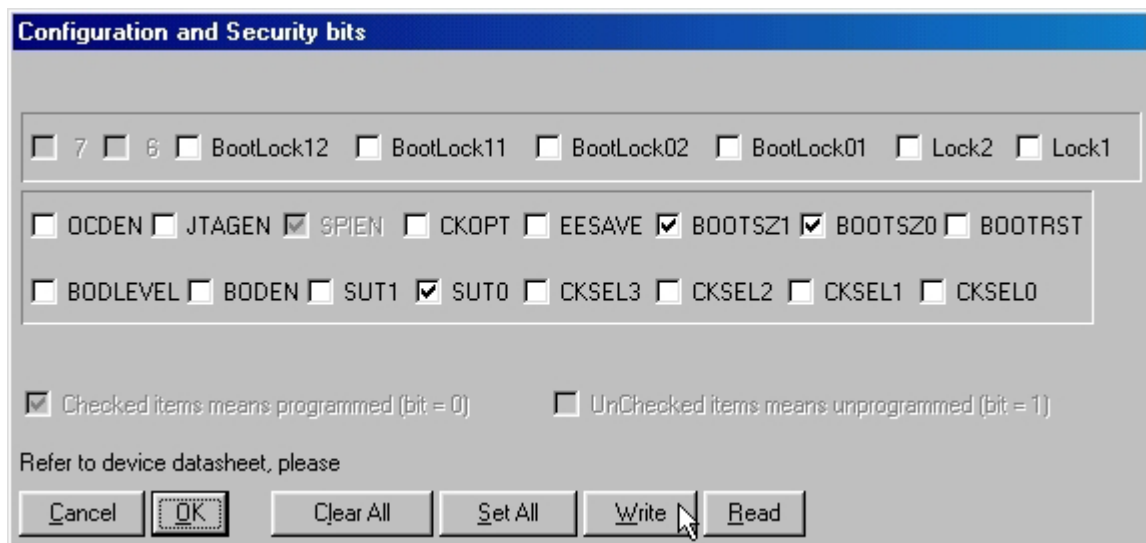


FIGURE B-16: CPU CONFIGURATION WITH PONY PROG



FIGURE B-17: PROGRAMMING USING PONY PROG

## APPENDIX C: GMM TST2 ELECTRIC DIAGRAM

In this appendix are reported the electric diagram of **GMM TST 2** support card that shows the connection modes for Mini Module signals. Detailed information on the board are available in the relative technical manual and the user can use them freely, for example to develop his own card that use the **GMM AM32** as a macro component.

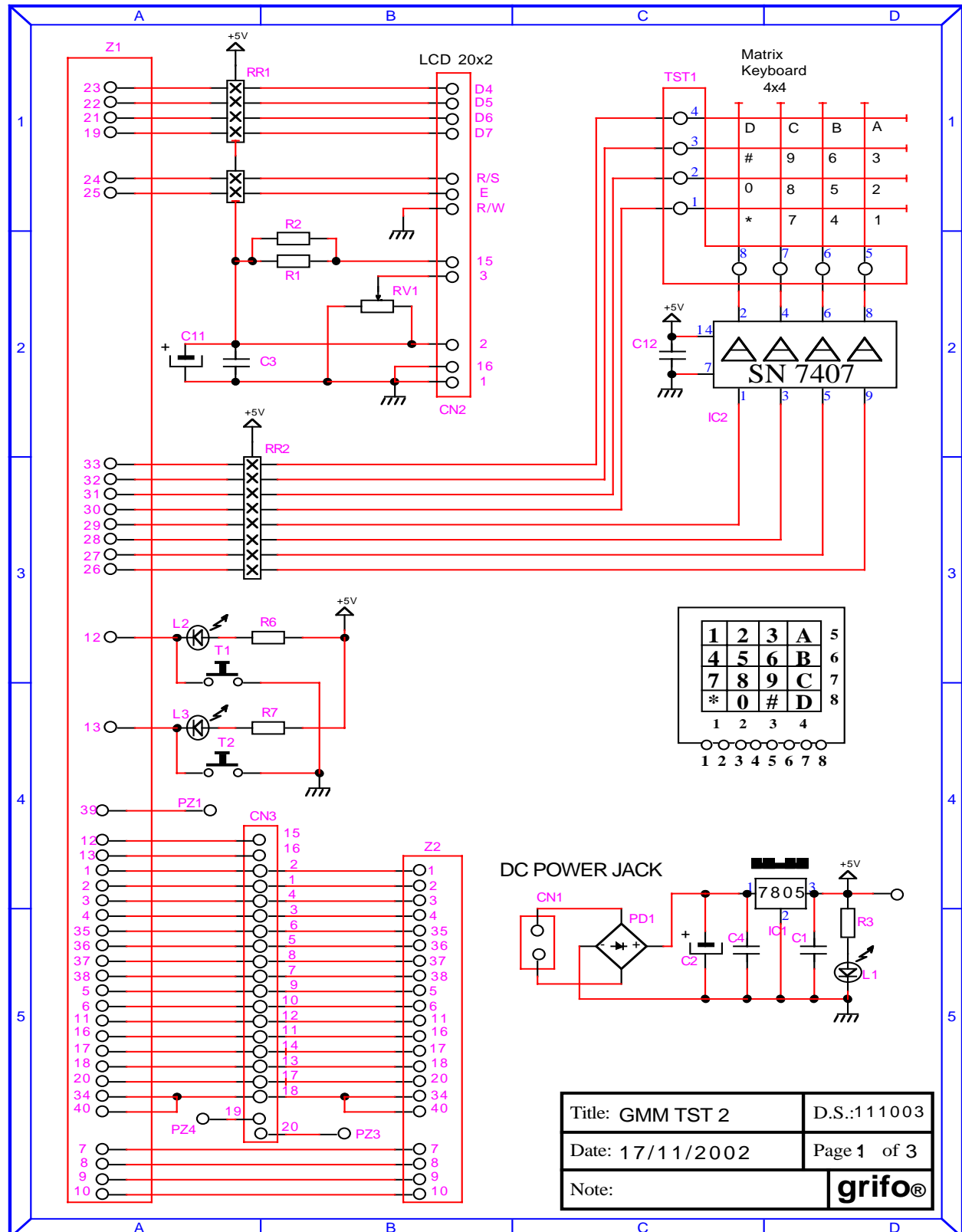


FIGURE C-1: GMM TST 2 ELECTRIC DIAGRAM (1 OF 3)

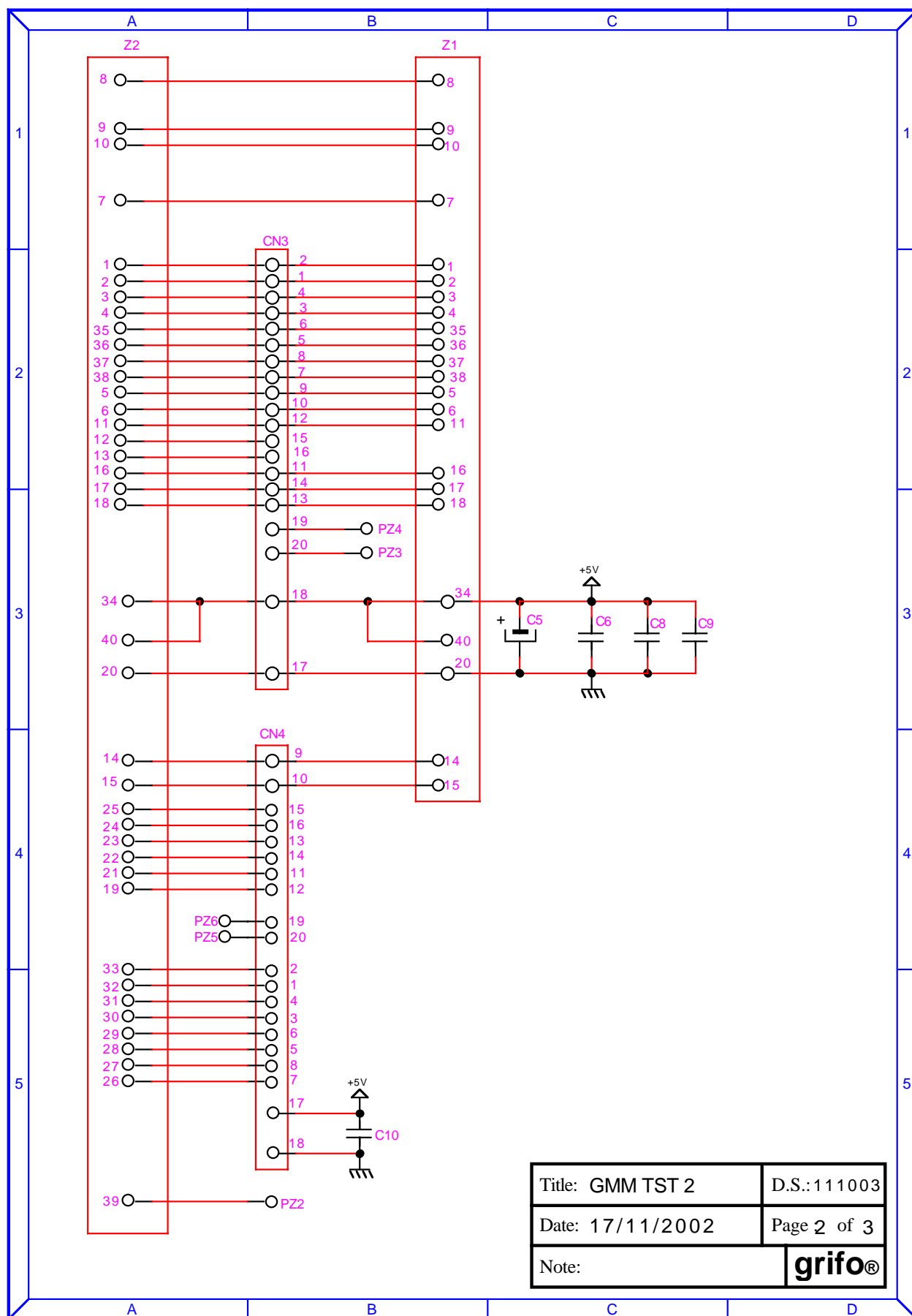


FIGURE C-2: GMM TST 2 ELECTRIC DIAGRAM (2 OF 3)



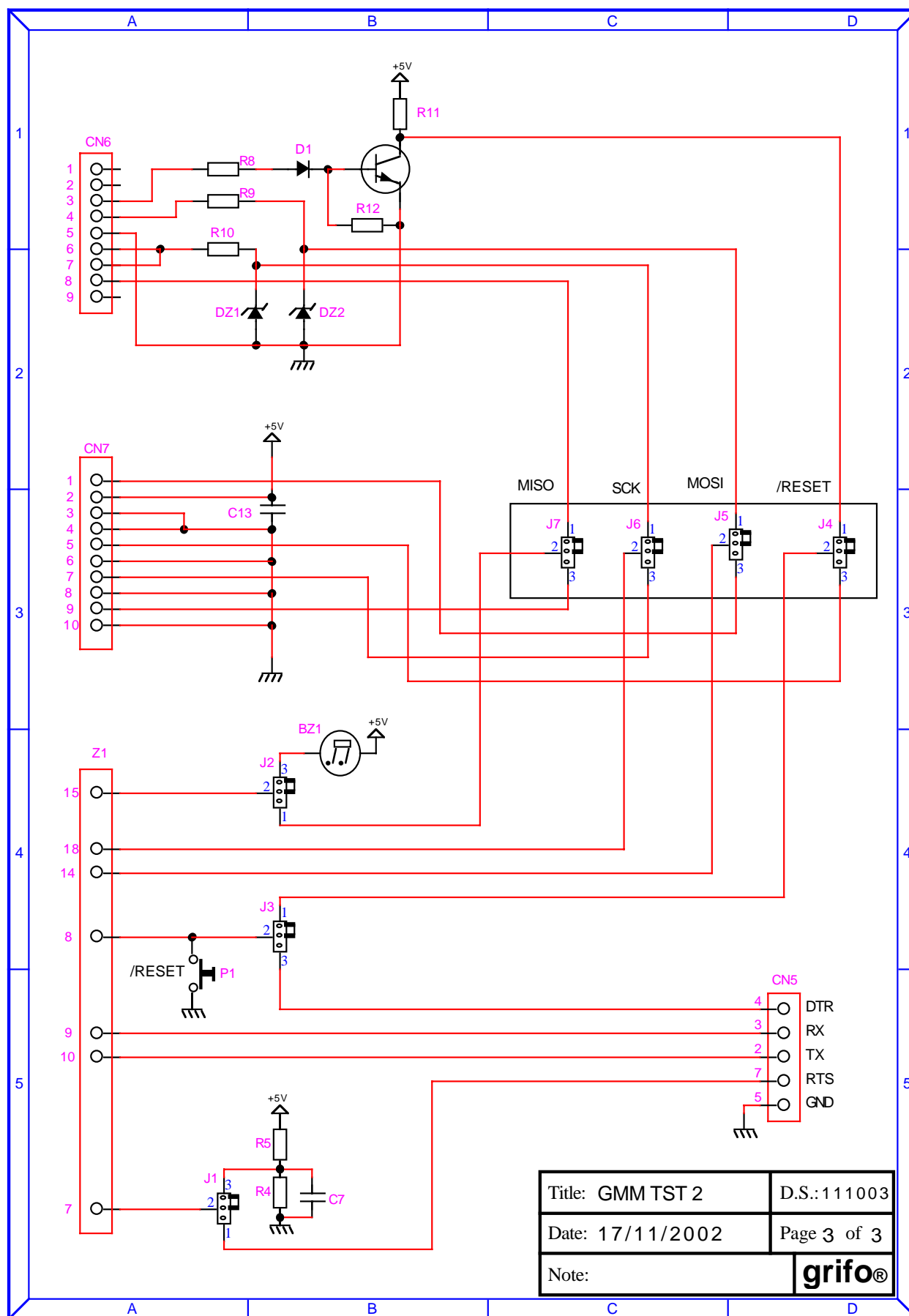


FIGURE C-3: GMM TST 2 ELECTRIC DIAGRAM (3 OF 3)



## APPENDIX D: ALPHABETICAL INDEX

## SYMBOLS

+VDC POW 13, 15, 20

## A

A/D CONVERTER 10, 12, 15, 20

ANALOG COMPARATOR 10, 12, 20

AVR ISP B-7

AVR STUDIO 34

AVRBOOTLOADERGRIFO 28

## B

BACK UP 9, 10, 16, 37

BACK UP BATTERY CONSUMPTION 13

BACK UP BATTERY VOLTAGE 13

BASCOM AVR 30, 34

BATTERY 9, 10, 16, 37

BIBLIOGRAPHY 38

BOOT LOADER 28

BROWN OUT THRESHOLD 13

## C

CARD VERSION 3

CLOCK 6, 12

CN1 14

CONFIGURATION BITS 28

CONNECTIONS 14

COUNTER MAXIMUM FREQUENCY 12

CURRENT CONSUMPTION 13

CURRENT LOOP 6, 9

## D

DDS MICRO C AVR 34

DIGITAL I/O LINES 12

DIP SWITCH 12, 16, 36

DSW1 9

## E

EEPROM 8, 12, 28

EPROM 8, 12, B-5

**F**

FLASH 8, 12, 28

**G**

GMB HR168 22

GMM TST 2 24, B-1, C-1

**H**

HOW TO START 26

**I**I<sup>2</sup>C BUS 8, 10, 12, 15, 20I<sup>2</sup>C BUS PULL-UP RESISTOR 13

ICC AVR 32, 34

IMPEDANCE ANALOG SIGNALS GENERATORS 13

INTERRUPT 21, 37

ISP 15

**J**

JTAG 21, 34

**L**

LEDS 12, 14, 36

**M**

MEMORY 8

MIPS 6

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

MSI 01 6

**P**

PONY PROG B-9

PORTS 6

POWER ON TIME 12

POWER SUPPLY 13, 15, 20

PROTECTION FUSES 28

PULL-UP 8, 37

PWM 10, 12, 20

**Q**

QTP 8

**R**

REAL TIME CLOCK 10, 12, 37

RELATIVE HUMIDITY 13

REVISION NUMBER 3

RISC 6

RS 232 6, 9, 12, 16, 18, 20

RS 232 PROTECTION 13

RS 422 6, 9

RS 485 6, 9

RTC 6, 8, 9, 10, 15, 16, 37

**S**

SCL 37

SDA 37

SIZE 13

SLAVE ADDRESS 37

SPI 8, 12, 20

SRAM 8, 9, 10, 12, 16, 37

**T**

TEMPERATURE RANGE 13

TIMER/COUNTER 10

TTL 6, 9, 12, 15, 16, 18, 20

**U****UEP49 B-1, B-3**

USART 9, 15

**W**

WATCH DOG 10, 12

WEIGHT 13

