Size: **150 x 100 x 40 mm** (W x H x D) with mounting hole on the card corner; optional plastic container for Ω DIN 46277-1 and DIN 46277-3 rails; interface for **matrix keyboard** with **22 keys**, on low profile connector; complete keys codes configuration through software, from the user; **autorepeat** and **keyclick** functions; direct interface for **22 status LEDs**, provided of blinking attribute, on low profile connector; interface for intelligent **alphanumeric display**, on low profile connector, available in the following models (it must be specified during order phase):

- **LCD display**, backlite or not, with **2** or **4** lines for **20** characters
- **Fluorescent display** with **2** or **4** lines for **20** characters
- **LCD display**, backlite or not, with **1**, **2** or **4** lines for **40** characters
- **Fluorescent display** with **1**, **2** or **4** lines for **40** characters

**Buzzer** programmable as **BELL** or with keyclick function; **E^2** up to **4 Kbyte**, used for permanent storage of set up, messages, key codes, etc.; memorization on **E^2** and visualization of more than **200** different **messages**, even with auto sliding mode; **1A Relay** with changeover contacts, serially controlled; interface for magnetic **badge card reader** on **1**, **2**, **3** track, motorized or manual; **RTC** with **256 Bytes** user accessible **RAM** backed by **Lithium battery**; **RS 232**, **RS 422**, **RS 485** or **Current Loop** serial line; communication selectable between **point to point** and **Master Slave** mode; local **set up** for communication parameters (Baud Rate, Stop bits, Keyclick, etc.); internal power supply capable of driving small external loads; **DC** or **AC power supply** from **5 Vdc** to **24 Vac**; on board logic protection through **TransZorb™**; it is possible to require custom panels and programs.
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

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

FIRMWARE VERSION .............................................................................................................. 1

GENERAL INFORMATION ...................................................................................................... 2
  KEYBOARD ........................................................................................................................ 3
  INDICATOR LEDS .............................................................................................................. 3
  DISPLAY ............................................................................................................................... 4
  SERIAL COMMUNICATION ................................................................................................. 6
  ON BOARD POWER SUPPLY ............................................................................................ 6
  REAL TIME CLOCK ............................................................................................................ 6
  BUZZER ............................................................................................................................... 7
  EEPROM ............................................................................................................................ 7
  RELAYS ............................................................................................................................... 7
  BADGE READER INTERFACE ........................................................................................... 8
  CPU .................................................................................................................................... 8

TECHNICAL FEATURES ........................................................................................................ 9
  GENERAL FEATURES ........................................................................................................ 9
  PHYSICAL FEATURES ...................................................................................................... 10
  ELECTRIC FEATURES ...................................................................................................... 10

INSTALLATION .................................................................................................................... 12
  CONNECTIONS .................................................................................................................. 12
    CN2 - POWER SUPPLYCONNECTOR ........................................................................... 12
    CN3 - SERIAL LINE CONNECTOR .................................................................................. 14
    CN4 - CONNECTOR FOR RELAY CONNECTION ....................................................... 20
    CN6 - FIRST DISPLAY CONNECTOR ............................................................................ 22
    CN7 - SECOND DISPLAY CONNECTOR .......................................................................... 23
    CN8 - THIRD DISPLAY CONNECTOR ............................................................................. 24
    CN5 - CONNECTOR FOR LCD DISPLAY BACKLIGHTING ............................................ 25
    CN9 - BADGE READER CONNECTOR .......................................................................... 26
    CN1 - EXTERNAL MATRIX KEYBOARD AND LEDS CONNECTOR ....................... 28
  CONTRAST REGULATION TRIMMER .............................................................................. 32
  I/O CONNECTION .............................................................................................................. 32
  DISPLAYS MANAGED ...................................................................................................... 33

JUMPERS ................................................................................................................................ 34
  2 PINS JUMPERS ............................................................................................................... 34
  3 PINS JUMPERS ............................................................................................................... 36
  4 PINS JUMPERS ............................................................................................................... 36
  5 PINS JUMPERS ............................................................................................................... 36
  POWER SUPPLY .............................................................................................................. 37
  SERIAL COMMUNICATION SELECTION ...................................................................... 38

SOFTWARE DESCRIPTION ................................................................................................. 41
LOCAL SET UP ................................................................. 41
EXPANDED FIRMWARE ...................................................... 42
NORMAL FIRMWARE ............................................................ 42
RECEIVE BUFFER .............................................................. 42
CHARACTERS VISUALIZATION ON THE DISPLAY .................. 43
COMMUNICATION MODALITIES ........................................ 43
MASTER SLAVE COMMUNICATION MODE .......................... 43
KEYBOARD ACQUISITION .................................................. 46
KEYS CODES ................................................................. 46
COMMANDS FOR CURSOR POSITION .............................. 48
CURSOR LEFT ................................................................. 48
CURSOR RIGHT ............................................................... 48
CURSOR DOWN ............................................................... 48
CURSOR UP ................................................................. 48
HOME ................................................................. 48
CARRIAGE RETURN ........................................................... 49
CARRIAGE RETURN + LINE FEED .................................. 49
ABSOLUTE CURSOR PLACEMENT .................................... 49
COMMANDS FOR CHARACTERS ERASURE .................... 50
BACKSPACE ............................................................ 50
CLEAR PAGE ............................................................... 50
CLEAR LINE ............................................................... 50
CLEAR END OF LINE ..................................................... 50
CLEAR END OF PAGE .................................................... 50
COMMANDS FOR CURSOR ATTRIBUTES MANAGEMENT .......... 51
CURSOR OFF ............................................................... 51
STEADY STATIC CURSOR ON .............................................. 51
BLINKING BLOCK CURSOR ON ......................................... 51
BLINKING "BLOCK" CURSOR .............................................. 51
COMMANDS FOR EEPROM ............................................ 52
REQUEST FOR EEPROM WRITING POSSIBILITY .................. 52
WRITING OF PRESENCE BYTE ........................................ 52
READING OF PRESENCE BYTE .......................................... 52
COMMANDS FOR GENERAL FUNCTIONS ............................ 53
READING OF VERSION NUMBER ........................................ 53
BEEP ............................................................... 53
RELAY ACTIVATION ........................................................ 53
RELAY DEACTIVATION .................................................... 53
COMMANDS FOR LEDS MANAGEMENT ............................ 54
LED ACTIVATION ........................................................ 54
LEDS MASK ACTIVATION ............................................... 54
COMMANDS FOR KEYBOARD MANAGEMENT ..................... 56
KEY RECONFIGURATION .................................................. 56
KEYCLICK ON WITHOUT MEMORIZATION ........................... 56
KEYCLICK OFF WITHOUT MEMORIZATION ........................... 56
KEYCLICK ON WITH MEMORIZATION ................................. 57
KEYCLICK OFF WITH MEMORIZATION ................................. 57
COMMANDS FOR MESSAGE MANAGEMENT ........................................................... 58
READING OF THE LAST STORAGED MESSAGE NUMBER ............................... 58
MESSAGE STORAGE .................................................................................................... 58
MESSAGE READING ..................................................................................................... 59
MESSAGE VISUALIZATION........................................................................................ 59
SCROLLING MESSAGES VISUALIZATION ............................................................ 60
COMMANDS FOR MANAGING THE ON-BOARD REAL TIME CLOCK ............. 61
WRITING OF A BYTE OF THE REAL TIME CLOCK SRAM ............................... 61
READING OF A BYTE OF THE REAL TIME CLOCK SRAM ............................... 61
CLOCK SET-UP ............................................................................................................... 61
CLOCK READING ........................................................................................................ .. 61
TIME VISUALIZATION ON THE DISPLAY .............................................................. 62
DATE VISUALIZATION ON THE DISPLAY .............................................................. 63
COMMANDS FOR BADGE READER MANAGEMENT ........................................... 64
STANDARD RECOGNIZED .......................................................................................... 65
SYNTAX OF THE STRING READ FROM THE BADGE .......................................... 65
READING A BADGE FROM A MOTORIZED READER .......................................... 66
BADGE READER REQUEST OF THE ACQUIRED STRING ................................. 66
READING OF THE BADGE READER CONFIGURATION BYTE ......................... 67
WRITING OF THE BADGE READER CONFIGURATION BYTE ......................... 68
ACQUISITION OF THE AUTOMATIC READER STATUS ...................................... 69
EXECUTION OF A NEW READING PROCEDURE ................................................. 69
CARD EJECTION FROM AUTOMATIC READER .................................................. 70
COMMAND FOR "EATING" THE CARD ................................................................... 70
EXTERNAL CARDS ................................................................................................................. 71
BIBLIOGRAPHY ................................................................................................................... 74
APPENDIX A: COMMAND CODES SUMMARY TABLES .............................................. A-1
APPENDIX B: DISPLAY CHARACTERS ........................................................................... B-1
APPENDIX C: MOUNTING OUTLINE DIMENSIONS .................................................... C-1
TERMINAL DIMENSIONS ........................................................................................  C-1
Ω RAILS MOUNTING .................................................................................................. C-2
APPENDIX D: ALPHABETICAL INDEX ............................................................................. D-1
FIGURES INDEX

FIGURE 1: BLOCK DIAGRAM ............................................................................................................. 5
FIGURE 2: CONSUMPTIONS TABLE .................................................................................................. 11
FIGURE 3: CN2 - POWER SUPPLY CONNECTOR .............................................................................. 12
FIGURE 4: AC POWER SUPPLY 8+24 VAC .................................................................................... 13
FIGURE 5: DC POWER SUPPLY +10+40 Vdc ............................................................................... 13
FIGURE 6: STABILIZED POWER SUPPLY +5 Vdc ......................................................................... 13
FIGURE 7: FETCHING +5 Vdc ......................................................................................................... 13
FIGURE 8: CN3 - SERIAL LINE CONNECTOR .................................................................................... 14
FIGURE 9: RS 232 POINT-TO-POINT CONNECTION EXAMPLE .................................................... 15
FIGURE 10: RS 422 POINT-TO-POINT CONNECTION EXAMPLE ................................................... 15
FIGURE 11: RS 485 POINT-TO-POINT CONNECTION EXAMPLE ................................................... 15
FIGURE 12: RS 485 NETWORK CONNECTION EXAMPLE ............................................................. 16
FIGURE 13: CURRENT LOOP 4 WIRES POINT-TO-POINT CONNECTION EXAMPLE ...................... 17
FIGURE 14: CURRENT LOOP 2 WIRES POINT-TO-POINT CONNECTION EXAMPLE ...................... 17
FIGURE 15: CURRENT LOOP NETWORK CONNECTION EXAMPLE ............................................... 18
FIGURE 16: JUMPERS, CONNECTORS, TRIMMER, ETC LOCATION ..................................................... 19
FIGURE 17: CN4 - REALY OUTPUT CONNECTOR ............................................................................ 20
FIGURE 18: RELAY OUTPUT CONNECTION .................................................................................... 21
FIGURE 19: RELAY OUTPUT SIGNALS LOCATION .......................................................................... 21
FIGURE 20: CN6 - FIRST DISPLAY CONNECTOR .............................................................................. 22
FIGURE 21: CN7 - SECOND DISPLAY CONNECTOR ............................................................................ 23
FIGURE 22: CN8 - THIRD DISPLAY CONNECTOR ............................................................................. 24
FIGURE 23: CN5 - CONNECTOR FOR LCD DISPLAY BACKLIGHTING .............................................. 25
FIGURE 24: CN9 - CONNECTOR FOR MANUAL BADGE ..................................................................... 26
FIGURE 25: CN9 - CONNECTOR FOR MOTORIZED BADGE .............................................................. 26
FIGURE 26: BADGE READER INTERFACE CONNECTION ............................................................... 27
FIGURE 27: CN1 - CONNECTOR FOR MATRIX KEYBOARD AND EXTERNAL LEDS ...................... 28
FIGURE 28: CONNECTING EXTERNAL MATRIX KEYBOARD .......................................................... 29
FIGURE 29: EXTERNAL LEDS CONNECTION .................................................................................... 30
FIGURE 30: COMPONENTS MAP (COMPONENTS SIDE) ................................................................. 31
FIGURE 31: COMPONENTS MAP (SOLDER SIDE) ............................................................................ 31
FIGURE 32: DISPLAYS CONNECTABLE ............................................................................................. 33
FIGURE 33: JUMPERS TABLE ......................................................................................................... 34
FIGURE 34: 2 PINS JUMPERS TABLE .............................................................................................. 34
FIGURE 35: JUMPERS LOCATION .................................................................................................... 35
FIGURE 36: 3 PINS JUMPERS TABLE .............................................................................................. 36
FIGURE 37: 4 PINS JUMPERS TABLE .............................................................................................. 36
FIGURE 38: 5 PINS JUMPERS TABLE .............................................................................................. 36
FIGURE 39: DRIVER FOR SERIAL COMMUNICATION SELECTION ............................................... 39
FIGURE 40: MASTER SLAVE COMMUNICATION EXAMPLE .......................................................... 45
FIGURE 41: DEFAULT KEY CODES .................................................................................................. 47
FIGURE 42: CARD PHOTO .............................................................................................................. 55
FIGURE 43: NUMBER OF MESSAGES THAT CAN BE STORED IN EEPROM ................................. 58
FIGURE 44: BADGE READERS CONNECTABLE ............................................................................... 64
FIGURE 45: FEATURES OF BADGES MANAGES ............................................................................ 65
FIGURE 46: TRACKS SIZE OD BADGES MANAGED ...................................................................... 66
FIGURE 47: AVAILABLE CONNECTIONS DIAGRAM ................................................................. 73
INTRODUCTION

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

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.

FIRMWARE VERSION

This handbook make reference to printed circuit version 230694 and to firmware version 2.0 or 5.8 and following ones. The validity of the information contained in this manual is subordinated to the firmware release number, so the user must always verify the correct correspondence between the notations. Inside the device, the firmware release number is written on the label stuck on the CPU or it can be obtained by a proper command sent through the serial line.

Hardware version is printed in several places, for example on the top of right on the component side.
GENERAL INFORMATION

MDU 01 is a card capable to solve all the operator interface problems, when the customer wants to develop by oneself the panel front size (display, keyboard, etc.). It is specifically designed for industrial use and for direct mounting on automatic machinery control box.

MDU 01 includes all the electronics of a video terminals and it is suitable to be the direct interface between operator and machinery in any of the control or command operations which could be necessary during running or diagnostic of the same.

MDU 01 can directly drive alphanumeric Fluorescent or LCD displays, with or without LEDs plane backlite in the 20 characters for 2 or 4 lines size or 40 characters for 1, 2 or 4 lines size. Moreover the MDU 01 has an interface for a matrix keyboard provided of 7x4 = 28 keys, with software configurable codes, and an interface for 22 status LEDs that can be turned on and off through proper serial commands.

The basic version of MDU 01 can be expanded utilizing the various options available, as: serial EEPROM, up to 4 Kbyte capacity for message saving; Real Time Clock with 256 Bytes of user SRAM, backed by Lithium battery; one relay managed by serial line; magnetic badge card acquisition, both with manual and motorized readers; many communication protocols; etc.

A set of connectors allow card connection to the external electronics in a fast and comfortable manner; so the customer can develop one's specific operator interface with no additional costs. MDU 01 offers the possibility to store up to 201 messages. These messages can be recalled through a simple serial command and put on the display. This allows to minimize the load of main CPU; in addition the control program does not have to keep in its memory all the messages for panel.

It is also possible to retrieve the stored messages through the serial line instead of displaying. This allows to use MDU 01 as a small mass memory device where to write special data like implant configuration, password, ID codes, etc.

To display more characters than the ones that fit in a line, it is possible to put a sliding message on the first line of the display up to 200 characters long.

MDU 01 allows also to make unexpensive network with up to 256 different units thanks to the possibility to use several serial electric protocols amongst the most diffused.

MDU 01 is able to execute an entire range of display commands, including Clear Screen, Position cursor, EEPROM reading or writing, LEDs blinking, etc., compatible with the ADDS ViewPoint video terminal.

Features of MDU 01, including optionals, are below described:

- Size: 150 x 100 x 40 mm (W x H x D) with mounting hole on the card corner.
- Optional plastic container for Ø DIN 46277-1 and DIN 46277-3 rails.
- Interface for matrix keyboard with 22 keys, on low profile connector.
- Complete keys codes configuration through software, from the user.
- Autorepeat and keyclick functions.
- Direct interface for 22 status LEDs, provided of blinking attribute, on low profile connector.
- Interface for intelligent alphanumeric display, on low profile connector, available in the following models (it must be specified during order phase):
  - LCD display, backlit or not, with 2 or 4 lines for 20 characters
  - Fluorescent display with 2 or 4 lines for 20 characters
  - LCD display, backlite or not, with 1, 2 or 4 lines for 40 characters
  - Fluorescent display with 1, 2 or 4 lines for 40 characters
- Buzzer programmable as BELL or with keyclick function.
- EEPROM up to 4 Kbyte, used for permanent storage of set up, messages, key codes, etc.
- Memorization on EEPROM and visualization of more than 200 different messages, even with auto sliding mode.
- 1A Relay with changeover contacts, serially controlled.
- Interface for magnetic badge card reader on 1, 2, 3 track, motorized or manual.
- RTC with 256 Bytes user accessible SRAM backed by Lithium battery.
- RS 232, RS 422, RS 485 or Current Loop serial line.
- Communication selectable between point to point and Master Slave mode.
- Local set up for communication parameters (Baud Rate, Stop bits, Keyclick, etc.).
- Internal power supply capable of driving small external loads.
- DC or AC power supply from 5 Vdc to 24 Vac.
- On board logic protection through TransZorb™.
- It is possible to require custom panels and programs.

Here follows a description of the board's functional blocks, with an indication of the operations performed by each one.

**KEYBOARD**

QTP 12 has an interface for a matrix keyboard with 6 rows by 4 columns that can manage up to 22 keys.
These keys are equipped with autorepeat and they are totally software reconfigurable or on the other hand the code sent in RS 232 when a key is pressed can be changed or disabled.
It is also possible to switch on/off the keyclick function, i.e the buzzer short activation each time a key is pressed.
Four keys are used for local setup of some operating parameters as described in paragraph “LOCAL SET-UP”.
Management of this simple keyboard allows MDU 01 to solve in an inexpensive way the problem of data input, even in case several data.
Connection to industrial keyboards allows to use it also in dangerous environments and to warrant a long operative life in any condition.

**INDICATOR LEDS**

MDU 01 has 22 indicator LEDs for different visual signals, that can be enabled, disabled or enabled with blinking attribute, through comfortable serial commands.
This functionality is totally autonomous and it doesn't need any intervention on user side.
LEDs must be connected through the specific low profile connector in common anod and do not require any external circuitry.
The main purpose of LED is to show a visual indication about the card's status, making so easier debug and verify operations even from a long distance from the panel, where the display is not readable.
For this reason there are three different status (off, on and blinking), to divide different operative situations as, for example, an allarm, an alert, a ready to work, etc.
DISPLAY

MDU 01 is available with Fluorescent VFD or backlit LCD alphanumeric displays featuring several characters numbers and size. In detail, following displays can be connected:

<table>
<thead>
<tr>
<th>Display Size</th>
<th>Connection to MDU 01 Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCD 20x2</td>
<td>MDU 01.C2</td>
</tr>
<tr>
<td>LCD 20x4</td>
<td>MDU 01.C4</td>
</tr>
<tr>
<td>LCD 20x4 big</td>
<td>MDU 01.C4B</td>
</tr>
<tr>
<td>LCD 40x1</td>
<td>MDU 01.C14</td>
</tr>
<tr>
<td>LCD 40x2</td>
<td>MDU 01.C24</td>
</tr>
<tr>
<td>LCD 40x4</td>
<td>MDU 01.C44</td>
</tr>
<tr>
<td>VFD 20x2</td>
<td>MDU 01.F2</td>
</tr>
<tr>
<td>VFD 20x4</td>
<td>MDU 01.F4</td>
</tr>
<tr>
<td>VFD 40x1</td>
<td>MDU 01.F14</td>
</tr>
<tr>
<td>VFD 40x2</td>
<td>MDU 01.F24</td>
</tr>
<tr>
<td>VFD 40x4</td>
<td>MDU 01.F44</td>
</tr>
</tbody>
</table>

These connections can be made through specific on board low profile connectors. Pinout of these connectors has been designed to easily connect to the displays, which is made through a flat cable in most cases.

LED backlighting of LCD models ensures a good visibility even when the environmental lighting changes and if necessary the user can modify the contrast regulation by acting on a specific trimmer. Another important feature of MDU 01 displays is their wide viewing angle that allows a good visibility from each frontal position. Further information on each display is reported in “DISPLAYS MANAGED” chapter.

The user must choose the right display (so the right MDU 01 model) that is sufficient for the information to visualize and for his visibility requirements. Please remark that all MDU 01 models listed are provided without display and that this latter must be ordered separately or purchased from a third part. For specific requirements on current consumption, visibility and price the card can be provided with LCD display not backlit. For detailed information about these options and their availability, please contact directly grifo® offices.
Figure 1: Block diagram
SERIAL COMMUNICATION

MDU 01 has one serial communication line. By default it is configured as RS 232, but using a proper indication in the ordering phase, it can be configured in:

<table>
<thead>
<tr>
<th>Serial Line</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS 422</td>
<td>.RS422 option</td>
</tr>
<tr>
<td>RS 485</td>
<td>.RS485 option</td>
</tr>
<tr>
<td>Current Loop</td>
<td>.CLOOP option</td>
</tr>
</tbody>
</table>

The physical protocol of both the serial lines is completely configurable through a dedicated set up modality that let the user select the values listed in "TECHNICAL SPECIFICATIONS" chapter, by the simple keyboard use.
Logic protocol can be point-to-point or master-slave, using the nineth bit technique; this latter allows MDU 01 to be connected in a network and to communicate with terminals of the same type or different type, easily and efficiently.

ON BOARD POWER SUPPLY

One of the most important peculiarity of MDU 01 is its own switching power supply that requires an input voltage variable from 8÷24 Vac or 10÷40 Vdc.
This section generates all the voltages used by the module.
In addition it can supply also all external devices (like LEDs, keyboard, display, etc.) and also small external loads.
This latter action can be easied by a specific option, called .5Vout, that allows to fetch directly the +5 Vdc voltage from on board switching/
As alternative, MDU 01 without power supply can be ordered using the code .5Vdc or .ALIM, in this case +5 Vdc stabilized power supply must be provided by an external source.
For further information please refer to "TECHNICAL SPECIFICATIONS" chapter.

REAL TIME CLOCK

MDU 01 as option can have a Real Time Clock featuring also 256 Bytes of SRAM at user disposal with Lithium battery data back-up.
This device is manageable with appropriate software commands by the user and in this way it is possible to set time and date, to read these data or to have them on the display on a given position.
There are also available two commands for writing and reading data from the on board SRAM of the RTC.
Code to order this option: .RTC
BUZZER

MDU 01 has a circuitry that generates a steady sound, based on a capacitive buzzer. This circuitry can be activated by software through a specific command for generating a simple beep or it can be linked to a key pressure, just to get the keyclick function, or it can signalize possible malfunctions. In latter case, after a power on, the card generates an intermittent sound and doesn't work correctly, there is a wrong condition that must be resolved: please contact grifo® technicians.

EEPROM

MDU 01 has on board EEPROM (size from 256 Bytes to 2 KBytes) for storing set up, communication protocol, keys codes, messages, and so on. Many of the stored data have vital importance so a serial EEPROM has been chosen to obtain the best warranties on validity and maintenance of the saved information, even when power supply is not available. It is possible to memorize up to 97 messages of 20 characters that can be first saved on EEPROM and then read or shown on the display at any moments, just giving a proper command to the terminal, with the right message identification number. MDU 01 also manages scrolling messages, to show on an unique line more text than it could be visible without scrolling. For detailed information about messages please read “COMMANDS FOR MESSAGES MANAGEMENT” paragraph. EEPROM size is chosen according to the application and the user needs. By default, the board is delivered with 256 Bytes of EEPROM so further memory configurations must be ordered by the following option code:

- EEPROM da 2048 Bytes -> opzione .MEX

Other EEPROM sizes described in the following paragraph are particular conditions whose cost and availability must be verified directly with grifo®.

RELAYS

MDU 01 can have as option n.1 Relay equipped with 1 Ampere contact. This latter can be switched on/off via software directly by the user and it makes available, on the appropriate connector, the contact both normally open and normally closed. This option could be necessary when the automatic door opening must be directly controlled by using the terminal placed in the immediate closeness; in this case it will be necessary a power driven by the said relay. Code for this option: .RELAY
BADGE READER INTERFACE

MDU 01 as option can have a circuitry that allows to interface the terminal to a single track magnetic card reader which can be insertion or sliding types for reading the track n.1, 2 or 3. MDU 01 automatically acquire the card and its decodification while the string is memorized into the local memory; in this way the user receives the message already decoded without being obliged to make further operations.

CPU

MDU 01 can accept microcontroller with pin out family 51 compliant like 87C51, 89C51, 87C52, 89C52 (manufactured by INTEL and several other brands). Normally the user has nothing to do with the CPU, in fact the board is delivered with a management firmware already programmed that performs all the tasks required, further information can be found in the chapter “SOFTWARE DESCRIPTION”.
# TECHNICAL FEATURES

## GENERAL FEATURES

**Resources:**
- 22 LEDs software manageable
- Interface for 22 software reconfigurable keys matrix keyboard
- Buzzer for beep, feedback and keyclick
- Full duplex RS 232 serial line, it can be buffered in RS 422, RS 485 or current loop (option)
- EEPROM size up to 4 KBytes for configuration, keys code, etc.
- Real Time Clock backed with Lithium battery and 256 Bytes SRAM (option)
- Magnetic cards reader for one track badge (option)
- Alphanumeric display in seven different models
- Trimmer to set LCD display contrast
- Switching power supply section

**Displays:**
- alphanumeric LCD 20x2; 20x4; 20x4BIG; 40x1; 40x2; 40x4
- alphanumeric fluorescent VFD 20x2; 20x4; 40x1; 40x2; 40x4

**CPU:**
- 87C51, 89C51 with crystal 14.7456 MHz
- 87C52, 89C52 with crystal 12 MHz

**Communication protocol:**
- Baud rate: 1200, 2400, 4800, 9600, 19200, 38400, 187500 Baud (option)
- Stop bit: 1 or 2
- Parity: none
- Bits x chr: 8, 9

**Com logic protocol:**
- Selectable between normal and master slave (Default: normal)

**Receive buffer size:**
- 30 characters

**Badge managed:**
- Track 1: I.A.T.A. ISO 3554
- Track 2: A.B.A. ISO 3554
- Track 3: M.I.N.S.T. ISO 4909
PHYSICAL FEATURES

Size: 150 x 100 x 40 mm (W x H x D)
See outline dimension in APPENDIX C

Weight: 210 g max

Mounting: Surface or flush panel mounting through four holes (diameter 3.5 mm) in the corners of printed circuit board
On a Ω rail container (standard DIN 46277-1 e DIN 46277-3): option code BLOCK.100.148

Length of keys connection: max. 30 cm (in normal conditions)

Keys autorepeat: After 500 ms and then every 100 ms

Temperature range: From 0 to 50 °C

Relative humidity: 20% up to 90% (without condense)

Connectors:
- CN1: low profile, male, vertical, 34 pins
- CN2: quick release screw terminal connector, 2 pins, male, 90°
- CN3: D type connector 9 pins, female
- CN4: quick release screw terminal connector, 3 pins, male, 90°
- CN5: low profile, male, vertical, 2 pins
- CN6: low profile, male, vertical, 16 pins
- CN7: low profile, male, vertical, 16 pins
- CN8: low profile, male, vertical, 20 pins
- CN9: low profile, male, vertical, 10 pins

ELECTRIC FEATURES

Power voltage: +10÷40 Vdc , 8÷24 Vac or +5 Vdc ± 5%

Power consumption: See next table

Power of supply section: 12.5 W

Output voltage: +5 Vdc

Current available on +5 Vdc: 1 A (resistive load)

Maximum voltage on relay: 35 Vdc / 24 Vac

RS 422-485 Termination: line termination resistor: 120 Ω

Here follows the list of MDU 01 power consumption referred to the different display types, already corrected with power supply section efficiency:
To reduce consumptions of **MDU 01** with LCD display it is possible to order them without backlighting: for further information please contact directly **grifo®**.

#### Figure 2: Consumptions table

<table>
<thead>
<tr>
<th>DISPLAY</th>
<th>CONSUMPTION +5 Vdc</th>
<th>CONSUMPTION 10÷40 Vdc</th>
<th>CONSUMPTION 8÷24 Vac</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCD 20x2 backlit: MDU 01-C2</td>
<td>410 mA -&gt; LED OFF</td>
<td>2.6 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>750 mA -&gt; LED ON</td>
<td>4.7 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>LCD 20x4 backlit: MDU 01-C4</td>
<td>200 mA -&gt; LED OFF</td>
<td>1.3 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>540 mA -&gt; LED ON</td>
<td>3.4 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>LCD 20x4 BIG backlit: MDU 01-C4B</td>
<td>230 mA -&gt; LED OFF</td>
<td>1.5 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>570 mA -&gt; LED ON</td>
<td>3.6 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>LCD 40x1 backlit: MDU 01-C14</td>
<td>410 mA -&gt; LED OFF</td>
<td>2.6 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>750 mA -&gt; LED ON</td>
<td>4.7 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>LCD 40x2 backlit: MDU 01-C24</td>
<td>410 mA -&gt; LED OFF</td>
<td>2.6 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>750 mA -&gt; LED ON</td>
<td>4.7 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>LCD 40x4 backlit: MDU 01-C44</td>
<td>810 mA -&gt; LED OFF</td>
<td>5.1 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1150 mA -&gt; LED ON</td>
<td>7.2 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>Fluorescent 20x2: MDU 01-F2</td>
<td>295÷395 mA -&gt; LED OFF</td>
<td>1.9÷2.5 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>635÷735 mA -&gt; LED ON</td>
<td>4.0÷4.7 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>Fluorescent 20x4: MDU 01-F4</td>
<td>380 mA -&gt; LED OFF</td>
<td>2.4 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>720 mA -&gt; LED ON</td>
<td>4.6 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>Fluorescent 40x1: MDU 01-F14</td>
<td>345÷495 mA -&gt; LED OFF</td>
<td>2.2÷3.1 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>585÷735 mA -&gt; LED ON</td>
<td>3.7÷4.6 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>Fluorescent 40x2: MDU 01-F24</td>
<td>795÷1045 mA -&gt; LED OFF</td>
<td>5.0÷6.5 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1135÷1385 mA -&gt; LED ON</td>
<td>7.1÷8.7 W -&gt; LED ON</td>
<td></td>
</tr>
<tr>
<td>Fluorescent 40x4: MDU 01-F44</td>
<td>1445÷2045 mA -&gt; LED OFF</td>
<td>9.0÷12.0 W -&gt; LED OFF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1785÷2385 mA -&gt; LED ON</td>
<td>11.0÷14.9 W -&gt; LED ON</td>
<td></td>
</tr>
</tbody>
</table>
INSTALLATION

In this chapter there are the information for a right installation and correct use of the terminal MDU 01.
In detail there are the locations and functions of each connector, of the user settable jumpers and of the trimmer.
For the connectors it is described the pin outs, the meaning of the connected signals and some connection examples, that simplify and speed the installation phase.

CONNECTIONS

MDU 01 terminal has 9 connectors that can be linkeded to other devices or directly to the field, according to system requirements.
In this paragraph there are connectors pin out, a short signals description (including the signals direction) and connectors location (see figure 16).
Connectors are accessible from top and bottom of the printed circuit board, to easy insertion and extraction.

CN2 - POWER SUPPLY CONNECTOR

CN2 is a 2 or 4 pins quick release, screw termina connector, pitch 5 mm.
On CN2 must be connected the unique power supply voltage for the terminal that can be one out of two different types:

\[
\begin{align*}
+5 \text{ Vdc} & = \text{I/O} - \text{+5 Vdc} \text{ power supply: according to ordered configuration it can be an input or an output.} \\
\text{GND} & = \text{- DC Power supply ground signal.} \\
\text{Vac} & = \text{I - AC power supply lines connected to on board switching section; input can be 8÷24 Vac or +10÷+40 Vdc.} \\
\end{align*}
\]

![CN2 - Power Supply Connector](image)

**Figure 3: CN2 - Power Supply Connector**
NOTE: For further information about power supply configurations, please refer to paragraph “POWER SUPPLY”.

**Figure 4: AC Power Supply 8+24 Vac**

**Figure 5: DC Power Supply +10÷40 Vdc**

**Figure 6: Stabilized Power Supply +5 Vdc**

**Figure 7: Fetching +5 Vdc**
**CN3 - SERIAL LINE CONNECTOR**

CN3 is a D type, 9 pins, female, 90 degrees connector. Through CN2 the user serially communicates with the terminal by using one of the standard electric standards RS 232, RS 422, RS 485 or current loop. Placing of the signals has been designed to reduce interference and electrical noise and to simplify connections with other systems, while the electric protocols follow the CCITT normative.

**Figure 8: CN3 - Serial Line Connector**

Signals description:

- **RX RS 232** = I - RS 232 Receive Data signal.
- **TX RS 232** = O - RS 232 Transmit Data signal.
- **RX- RS 422** = I - Negative signal for RS 422 serial differential receive.
- **RX+ RS 422** = I - Positive signal for RS 422 serial differential receive.
- **TX- RS 422** = O - Negative signal for RS 422 serial differential transmit.
- **TX+ RS 422** = O - Positive signal for RS 422 serial differential transmit.
- **RXTX- RS 485** = I/O - Negative signal for RS 485 serial differential receive and transmit.
- **RXTX+ RS 485** = I/O - Positive signal for RS 485 serial differential receive and transmit.
- **RX- C.L.** = I - Negative signal for current loop serial bipolar receive.
- **RX+ C.L.** = I - Positive signal for current loop serial bipolar receive.
- **TX- C.L.** = O - Negative signal for current loop serial bipolar transmit.
- **TX+ C.L.** = O - Positive signal for current loop serial bipolar transmit.
- **GND** = - Ground signal
Figure 9: RS 232 point-to-point connection example

Figure 10: RS 422 point-to-point connection example

Figure 11: RS 485 point-to-point connection example
Figure 12: RS 485 Network Connection Example

Please remark that in a RS 485 network two forcing resistors must be connected across the net and two termination resistors (120 \( \Omega \)) must be placed at its extremes, respectevely near the Master unit and the Slave unit at the greatest distance from the Master.

Forcing and terminating circuitry is installed on MDU 01 board. It can be enabled or disabled through specific jumers, as explained later.

For Master unit, connect it only if needed (many RS 232-RS 485 converters already feature it).

For further information please refer to TEXAS INSTRUMENTS Data-Book, "RS 422 and RS 485 Interface Circuits", the introduction about RS 422-485.
**Figure 13:** Current Loop 4 wires point-to-point connection example

**Figure 14:** Current Loop 2 wires point-to-point connection example
Possible Current Loop connections are two: 2 wires and 4 wires. These connections are shown in figures 13+15 where it is possible to see the voltage for VCL and the resistances for current limitation (R). The supply voltage varies in compliance with the number of connected devices and voltage drop on the connection cable.

The choice of the values for these components must be done considering that:

- circulation of a 20 mA current must be guaranteed;
- potential drop on each transmitter is about 2.35 V with a 20 mA current;
- potential drop on each receiver is about 2.52 V with a 20 mA current;
- in case of short circuit each transmitter must dissipate at most 125 mW;
- in case of short circuit each receiver must dissipate at most 90 mW.

For further info please refer to HEWLETT-PACKARD Data Book, (HCPL 4100 and 4200 devices).
FIGURE 16: JUMPERS, CONNECTORS, TRIMMER, ETC LOCATION.
CN4 - CONNECTOR FOR RELAY CONNECTION

The connector for the on-board relay connection (option) is named CN4 on MDU 01. It is a 3 pins quick release screw terminal connection; on it there are the contacts of optional relay output: normal open, normal close and common. As described in “ELECTRIC FEATURES” paragraph, we remark that maximum current for CN4 is 1 A, hence to drive heavier loads and adaptor (like a triac, a power relay, etc.) must be used. Code for this option: .RELAY

![Diagram of CN4 connector]

**Figure 17: CN4 - Relay output connector**

Signals description:

- **COMMON** = Relay commun contact
- **N. CLOSED** = Relay contact normal closed
- **N. OPEN** = Relay contact normal open

Both normally open and normally closed contacts are provided with own disturb and transient suppressor filter, 24 Vac MOV.

For further information please see the following figures that show signals connection and their position on the connector.
**Figure 18:** Relay output connection

**Figure 19:** Relay output signals location
CN6 - FIRST DISPLAY CONNECTOR

CN6 is 16 pins low profile, male, vertical connector with 2.54 mm pitch. CN6 features all the control and command signals of LCD display (or LCD compatible) with standard pin out that allows direct connection to most of them.
In detail this connector is used to interface the following displays:

- LCD 20x2, MDU 01-C2
- LCD 20x4, MDU 01-C4
- LCD 20x4 BIG, MDU 01-C4B
- LCD 20x4, MDU 01-C4
- LCD 40x1, MDU 01-C14
- LCD 40x2, MDU 01-C24

so it is present only when one of above models is ordered.
For further information about display connection please see paragraphs “DISPLAYS MANAGED” and “I/O CONNECTIONS”.

![Figure 20: CN6 - First Display Connector](image)

Signals description:

- **A (+)** = O - Anod or positive terminal for LED backlighting of LCD display.
- **K (-)** = O - Kathod or negative terminal for LED backlighting of LCD display.
- **DBx** = I/O - Line x of display 8 bit data bus.
- **R/W** = O - Control signal to select read or write operation.
- **E** = O - Display enable signal.
- **RS** = O - Control signal to select between command or data operation.
- **Vo, VLC** = V - Contrast tension for LCD display.
- **+5 Vdc, VDD** = V - Display power supply tension.
- **GND, VSS** = - Ground signal.
CN7 - SECOND DISPLAY CONNECTOR

CN7 is 16 pins low profile, male, vertical connector with 2.54 mm pitch. CN7 features all the control and command signals of LCD display 40x4 with standard pin out that allows direct connection to most of them. In detail this connector is used to interface the following displays:

LCD 40x4       MDU 01-C44

so it is present only when one of above models is ordered.
For further information about display connection please see paragraphs “DISPLAYS MANAGED” and “I/O CONNECTIONS”.

Signals description:

\[ \begin{align*}
DBx & = \text{I/O - Line } x \text{ of display 8 bit data bus.} \\
R/W & = \text{O - Control signal to select read or write operation.} \\
EU & = \text{O - Top part of display enable signal.} \\
ED & = \text{O - Bottom part of display enable signal.} \\
RS & = \text{O - Control signal to select between command or data operation.} \\
Vo , V_{LC} & = \text{O - Contrast tension for LCD display.} \\
+5 \text{Vdc} , V_{DD} & = \text{O - Display power supply tension.} \\
GND , V_{SS} & = \text{O - Ground signal.} \\
N.C. & = \text{O - Not connected.}
\end{align*} \]
CN8 - THIRD DISPLAY CONNECTOR

CN8 is 20 pins low profile, male, vertical connector with 2.54 mm pitch. CN8 features all the control and command signals of fluorescent VFD display (or compatible) with standard pin out that allows direct connection to most of them. In detail this connector is used to interface the following displays:

- VFD 20x2   MDU 01.F2
- VFD 40x1   MDU 01.F14
- VFD 40x2   MDU 01.F24
- VFD 40x4   MDU 01.F44

so it is present only when one of above models is ordered.
For further information about display connection please see paragraphs “DISPLAYS MANAGED” and “I/O CONNECTIONS”.

![Figure 22: CN8 - Third display connector](image)

**Signals description:**

- **DBx** = I/O - Line x of display 8 bit data bus.
- **/WR** = O - Control signal to select read or write operation.
- **/SEL** = O - Display enable signal.
- **TEST** = O - Display test mode enable signal.
- **BUSY** = I - Status signal to indicate whether display is busy.
- **+5 Vdc** = O - Display power supply tension.
- **GND** = Ground signal.
- **N.C.** = Not connected.
CN5 - CONNECTOR FOR LCD DISPLAY BACKLIGHTING

CN5 is a 2 pins low profile, male, vertical connector. CN5 always features the two signals for LCD displays LED backlighting. Normally CN5 is used with LCD displays models where backlighting lamp is not already connected to display connector (like LCD 40x4 of MDU 01-C44) or when there are different needs to supply it. Please remark that backlighting signals are generated by a specific circuitry of MDU 01 charged to provide stabilized current and voltage from +5 Vdc. The same signals are present also on CN6, that can be used in all standard cases. For further information about display connection please see paragraphs “DISPLAYS MANAGED” and “I/O CONNECTIONS”.

![CN5 Connector](image)

**FIGURE 23: CN5 - CONNECTOR FOR LCD DISPLAY BACKLIGHTING**

Signals description:

A (+) = O - Anod or positive terminal for LED backlighting of LCD display.
K (-) = O - Kathod or negative terminal for LED backlighting of LCD display.

Female connector for CN5 can also be ordered from grifo® using the following codes:

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS2 BAT</td>
<td>2 pins female connectors;</td>
</tr>
<tr>
<td>CSF Cable</td>
<td>Set of crimped and colored cables 1 meter long, to insert into female connector.</td>
</tr>
</tbody>
</table>
CN9 - BADGE READER CONNECTOR

CN9 is a 10 pins, low profile, male 90 degrees connector with pitch 2.54 mm.
CN9 features the singlas for a passive Badge reader in several models: sliding, inserion, motorized, etc.
Following figures show the pin out according to the reader used: manual or motorized.
Signal names are the same in both cases to easy the connection.
Please remark that connector and Badge reader circuit are optional and must be ordered with the following code: .BADGE
For further information about Badge connectable and their management modalities please see the specific paragraph: “COMMANDS TO MANAGE BADGE READERS”.

**FIGURE 24: CN9 - CONNECTOR FOR MANUAL BADGE**

```
   N.C. = = = = = = = = = = = =
   SWITC = = = = = = = = = = =
   H/RCP = = = = = = = = = = =
   OPEN = = = = = = = = = = =
   +5 Vd = = = = = = = = = = =

   1  2
   3  4
   5  6
   7  8
   9  10

   GND
```

**FIGURE 25: CN9 - CONNECTOR FOR MOTORIZED BADGE**

```
   /MRS = = = = = = = = = = =
   /S1  = = = = = = = = = = =
   /RCP = = = = = = = = = = =
   N.C. = = = = = = = = = = =
   N.C. = = = = = = = = = = =

   1  2
   3  4
   5  6
   7  8
   9  10

   /MFS
   /RDP
   /S2
   GND
```

Signals description:

- **OPEN** = - This signal must remain NOT CONNECTED.
- **/RDP** = - Data signal (Read Data Pulse), coming from Badge reader, indicating the track to be used.
/RCP = I - Clock signal (Read Clock Pulse), coming from Badge reader, indicating the track to be used.

/CPS = I - Card Presence Signal, coming from Badge reader.

Switch = I - This signal must be connected only if an insertion reader is used, in particular it must be connected to the N.O. context of switch, present only on that kind of reader, indicating that the card is inserted also only in part. COM contact of this switch must be necessarily connected to GND.

/MRS = O - Motor Reverse Signal directed to Badge reader.

/MFS = O - Motor Forward Signal directed to Badge reader.

/S1 = I - Signal driven by the switch indicating that the is in the insertion slot, coming from Badge reader.

/S2 = I - Signal driven by the switch indicating that the card edge is under the reading head, coming from Badge reader.

+5 Vdc = O - Power supply for Badge reader.

GND = - Ground.

N.C. = - Not connected.

In case of motorized reader, power supply generated on the board is not enough to drive the motors so it must be proved from external source, as described in the paragraph “POXER SUPPLY”.

Management of Badge reader interface is based on CPU I/O signals and interrupts, as described in the following figure. Firmware drives the lines according to settings and interacts with master unit through serial communication line.

**Figure 26: Badge reader interface connection**
CN1 - EXTERNAL MATRIX KEYBOARD AND LEDS CONNECTOR

CN1 is a 34 pins, low profile, vertical, male connector with 2.54 mm pitch. CN1 features all the signals needed to connect 22 keys and 22 external LEDs, as described in the following figures:

```
<table>
<thead>
<tr>
<th>K LED 0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>K LED 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>K LED 2</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td>K LED 21</td>
</tr>
<tr>
<td>K LED 4</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
<td>30</td>
<td>K LED 3</td>
</tr>
<tr>
<td>K LED 6</td>
<td>29</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>N.C.</td>
</tr>
<tr>
<td>K LED 8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K LED 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K LED 12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K LED 14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K LED 16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K LED 18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K LED 20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A LEDS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>KEY COL 2</td>
<td>25</td>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>KEY COL 3</td>
</tr>
<tr>
<td>KEY COL 4</td>
<td>27</td>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>KEY ROW 1</td>
</tr>
<tr>
<td>KEY ROW 2</td>
<td>29</td>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>KEY ROW 3</td>
</tr>
<tr>
<td>KEY ROW 4</td>
<td>31</td>
<td>32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>KEY ROW 5</td>
</tr>
<tr>
<td>KEY ROW 6</td>
<td>33</td>
<td>34</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>N.C.</td>
</tr>
</tbody>
</table>
```

**Figure 27: CN1 - Connector for matrix keyboard and external LEDs**

Signals description:

- **KEY COL n** = 1 - Input line for n-th column of 4x6 external matrix keyboard.
- **KEY ROW n** = 0 - Output line for n-th row of 4x6 external matrix keyboard.
- **K LED n** = 0 - Output signal for kathod of n-th external LED.
- **A LEDS** = - Signal for common anods of external LEDs.
- **N.C.** = - Not connected.
External keys must be connected so that pressing one key connects input KEY COL n and output KEY ROW n.

Following figure shows how such matrix connection must be performed with corresponding 22 keys identification numbers (codes).

These numbers are used in the following paragraphs where keyboard and its management are quoted.

![Connecting External Matrix Keyboard](image)

**Figure 28: Connecting External Matrix Keyboard**

Signals of 4x6 matrix keyboard are TTL and can be connected to any kind of normally open button capable to provide low contact resistance; length of connection must not be over a total of 30 cm unless external favourable conditions allow to extend it.

External LEDs connection must be common anod and requires no other component, as shown in the following figure, which reports also numeration of LEDs used in all following paragraphs.
FIGURE 29: EXTERNAL LEDS CONNECTION
FIGURE 30: COMPONENTS MAP (COMPONENTS SIDE)

FIGURA 31: COMPONENTS MAP (SOLDER SIDE)
CONTRAST REGULATION TRIMMER

On MDU 01 board there is a trimmer that defines the contrast on LCD displays. This trimmer, named CR1, is set by grifo® to obtain the best display visibility in each working conditions and normally the user must not change its position. In case of specific requirements, as external light very low or very high, it can be changed by little rotation in both directions until the visibility is improved.

For recognizing the location of contrast regulation trimmer, please refer to figure 16.

I/O CONNECTION

To prevent possible connecting problems between MDU 01 and the external systems, the user has to read carefully the previous paragraph information and he must follow these instructions:

- For RS 232, RS 422, RS 485 and current loop signals the user must follow the standard rules of each one of these protocols;

- For all TTL signals the user must follow the rules of this electric standard. The connected digital signal must be always referred to card digital ground. For TTL signals, the 0V level corresponds to logic state 0, while 5V level corresponds to logic state 1.

- To avoid driving problems and minimize the effects of eventual disturbs from the field, connectors CN1, CN6, CN7, CN8 and CN9 must be connected with cables of maximum length 30 cm.

- Location of signals on CN6 and CN7 is "reversed" compared to the location of connectors of the same displays: this allows to install a male low profile connector on the back (the side without display) without taking place on the front. So, the connector on the display is "reversed" and can be connected to the corresponding on MDU 01 through a common and cheap flat cable 1-by-1. This latter can be ordered from grifo® with code: FLT 16+16 40 cm.

- Location of signals on CN8 is the same as the one on the displays provided with the same connector. On the display, a male low profile connector on the back (the side without display) is already mounted, it can be connected to the corresponding on MDU 01 through a common and cheap flat cable 1-by-1. This latter can be ordered from grifo® with code: FLT 20+20 40 cm.

- For all the displays featuring a connector mechanically incompatible with the corresponding on MDU 01, just connect the signals with the same name or the same function. To make this, it is possible to use a flat cable with standard female connector by card side and the same connector with opportune connections by display side.
DISPLAYS MANAGED

Hardware structure of MDU 01 and its management firmware can use most alphanumeric displays currently in commerce, both LCD and VFD fluorescent.

As an example below is reported a list of displays tested, complete with type, part number, manufacturer and connector to use on MDU 01.

<table>
<thead>
<tr>
<th>TYPE</th>
<th>MODEL</th>
<th>MANUFACTURER</th>
<th>CONNECTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCD 20x2</td>
<td>SSC2A20DLYY-02</td>
<td>SDEC</td>
<td>CN6</td>
</tr>
<tr>
<td>LCD 20x4</td>
<td>SSC4A20DLYY-02</td>
<td>SDEC</td>
<td>CN6</td>
</tr>
<tr>
<td>LCD 20x4</td>
<td>AC204AYILY02</td>
<td>AMPIRE</td>
<td>CN6</td>
</tr>
<tr>
<td>LCD 20x4 BIG</td>
<td>CMC420L03YBN</td>
<td>CTC</td>
<td>CN6</td>
</tr>
<tr>
<td>LCD 40x1</td>
<td>L4041B1J000T</td>
<td>SEIKO , TECDIS</td>
<td>CN6</td>
</tr>
<tr>
<td>LCD 40x2</td>
<td>L4042B1J000T</td>
<td>SEIKO , TECDIS</td>
<td>CN6</td>
</tr>
<tr>
<td>LCD 40x4</td>
<td>M4024</td>
<td>SEIKO , TECDIS</td>
<td>CN5 + CN7</td>
</tr>
<tr>
<td>VFD 20x2</td>
<td>CU20025ECPB-U1J</td>
<td>NORITAKE ITRON</td>
<td>CN6</td>
</tr>
<tr>
<td>VFD 20x2</td>
<td>M202SD01B</td>
<td>FUTABA</td>
<td>CN8</td>
</tr>
<tr>
<td>VFD 20x4</td>
<td>CU20045SCPB-U2J</td>
<td>NORITAKE ITRON</td>
<td>CN6</td>
</tr>
<tr>
<td>VFD 40x1</td>
<td>M40SD04G</td>
<td>FUTABA</td>
<td>CN8</td>
</tr>
<tr>
<td>VFD 40x2</td>
<td>CU40025SCPB-U1J</td>
<td>NORITAKE ITRON</td>
<td>CN6</td>
</tr>
<tr>
<td>VFD 40x2</td>
<td>M402SD04CA</td>
<td>FUTABA</td>
<td>CN8</td>
</tr>
<tr>
<td>VFD 40x4</td>
<td>M404SD01BA</td>
<td>FUTABA</td>
<td>CN8</td>
</tr>
</tbody>
</table>

FIGURE 32: DISPLAYS CONNECTABLE

Please remark that above mentioned part numbers may vary for secondary features like viewing angle, color of backlight, working temperature range, etc.

To identify the most appropriate model it is suggested to read always the technical documentation of manufacturer; alternatively displays can be purchased directly from grifo®.

Appendix B reports the characters set of models in the list above, corresponding to board firmware management.
JUMPERS

On MDU 01 there are 6 jumpers and one dip switch for card configuration. Connecting these jumpers, the user can define some parameters of its working modes. Here below is the jumpers list, location and function:

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>N° PINS</th>
<th>PURPOSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1</td>
<td>5</td>
<td>Selects direction and operating mode for RS 422 and RS 485 serial line.</td>
</tr>
<tr>
<td>J2</td>
<td>3</td>
<td>Selects reception driver for RS 422 and RS 485 serial line.</td>
</tr>
<tr>
<td>J3</td>
<td>3</td>
<td>Connects to GND the holes for board mechanical docking.</td>
</tr>
<tr>
<td>J4</td>
<td>4</td>
<td>Selects electric protocol for serial communication.</td>
</tr>
<tr>
<td>J5</td>
<td>2</td>
<td>Connects termination circuitry to RS 485 communication line or RS 422 reception line.</td>
</tr>
<tr>
<td>J6</td>
<td>2</td>
<td>Connects termination circuitry to RS 422 transmission line.</td>
</tr>
</tbody>
</table>

**FIGURE 33: JUMPERS TABLE**

The following tables describe all the right connections of MDU 01 jumpers with their relative functions.

To recognize these valid connections, please refer to the board printed diagram (serigraph) or to figures 30 and 31 of this manual, where the pins numeration is listed; for recognizing jumpers location, please refer to figure 35.

The "*" 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.

Further information about purpose of jumpers are reported in the following paragraphs.

2 PINS JUMPERS

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>CONNECTION</th>
<th>PURPOSE</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>J5</td>
<td>not connected</td>
<td>Does not connect termination circuitry to RS 485 communication or to RS 422 reception line.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td>Connects termination circuitry to RS 485 communication or to RS 422 reception line.</td>
<td>*</td>
</tr>
<tr>
<td>J6</td>
<td>not connected</td>
<td>Does not connect termination circuitry to RS 422 transmission line.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td>Connects termination circuitry to RS 422 transmission line.</td>
<td>*</td>
</tr>
</tbody>
</table>

**FIGURE 34: 2 PINS JUMPERS TABLE**
Figure 35: Jumpers Location
### 3 PINS JUMPERS

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>CONNECTION</th>
<th>PURPOSE</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>J2</td>
<td>position 1-2</td>
<td>Connects reception line to RS 422 or RS 485 reception signal.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>Internal use only. Do not use.</td>
<td></td>
</tr>
<tr>
<td>J3</td>
<td>position 1-2</td>
<td>Does not connect to GND the four holes for board mechanical clamping.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>Connect to GND the four holes for board mechanical clamping.</td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE 36: 3 PINS JUMPERS TABLE**

### 4 PINS JUMPERS

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>CONNECTION</th>
<th>PURPOSE</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>J4</td>
<td>position 1-2</td>
<td>Selects RS 232 electric protocol for serial communication.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>Selects current loop electric protocol for serial communication.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>position 2-4</td>
<td>Selects RS 422 or RS 485 electric protocol for serial communication.</td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE 37: 4 PINS JUMPERS TABLE**

### 5 PINS JUMPERS

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>CONNECTION</th>
<th>PURPOSE</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1</td>
<td>position 1-2 and 3-4</td>
<td>Enable RS 485 serial communication (2 wires half duplex).</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3 and 4-5</td>
<td>Enable RS 422 serial communication (4 wires half duplex or full duplex).</td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE 38: 5 PINS JUMPERS TABLE**
POWER SUPPLY

Terminal **MDU 01** is provided with a power supply section that solves in an efficient and comfortable way the problem to supply the boards in any situation. It generates energy for all sections of the board: control logic, display, backlighting, serial interfaces, Real Time Clock, LEDs, buzzer and keyboard. Here follow voltages required according to the configuration:

Default: This configuration features a switching power supply that requires 10÷40 Vdc or 8÷24 Vac provided through pins 3 and 4 of CN1 (as shown in figures 4 and 5). This allows to supply the terminal using standard inexpensive power sources like transformers, batteries, solar cells, etc. Power supply **EXPS-1** can be connected directly to the terminal starting from mains. Please remark that on board switching supply is provided with diode bridge rectifier, so in case of DC supply, all ground signals of the terminal (GND) are NOT at the same potential of power supply. This is the default version, normally delivered without further requests.

Version **.5Vdc** or **.ALIM**: This configuration is not provided of any supply section, so 5 Vdc ± 5% stabilized supply voltage must be provided by an external source through pins 1 and 2 of CN1 (polarity MUST be respected, as shown in figure 6). This allows to provide energy to the terminal through power supply, other cards, etc. This configuration is OEM only, please contact grifo®.

Version **.5Vout**: This configuration features the same power supply section, of default configuration, in addition it is provided with a connector to fetch the +5 Vdc tension generated by on board switching. This tension is available on pins 1 and 2 of CN1 (as shown in figure 7) and can be used to supply small external loads like backlight lamps, badge readers, other cards of the application, etc. Maximum current from this connector may vary from 115 to 1960 mA, according to the configuration of **MDU 01**.

Selection of power supply section must be performed during the order phase, in fact this means a different hardware configuration that must be made by grifo® technicians. **MDU 01** is provided with a TransZorb™ protection circuit to avoid damages from malfunctioning or broken supply section. It is also provided with a distributed filtering circuitry that saves the terminal from disturbs or noise from the field, improving the overall system performances. For further information please refer to paragraph “ELECTRIC FEATURES”.

**MDU 01** Rel. 5.00
SERIAL COMMUNICATION SELECTION

Serial line of MDU 01 can be buffered in RS 232, RS 422, RS 485 or current loop. By hardware can be selected which one of these electric standards is used, through jumpers connection (as described in the previous tables) and drivers installation. 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 decided by an opportune setup procedure.

In the following paragraphs there are all the information on serial communication configurations. Some devices needed for RS 422, RS 485 and current loop configurations are not mounted on the board in standard configuration; this is why each fist non-standard (non-RS 232) serial configuration must be always performed by grifo® technicians.

This far the user can change in autonomy the configuration following the informations below:

- SERIAL LINE IN RS 232 (default configuration)

<table>
<thead>
<tr>
<th>J1</th>
<th>J2</th>
<th>J4</th>
<th>J5, J6</th>
<th>IC5</th>
<th>IC7</th>
<th>IC8</th>
<th>IC6</th>
<th>IC9</th>
</tr>
</thead>
<tbody>
<tr>
<td>indifferent</td>
<td>indifferent</td>
<td>position 1-2</td>
<td>not connected</td>
<td>driver MAX 202</td>
<td>no device</td>
<td>no device</td>
<td>no device</td>
<td>no device</td>
</tr>
</tbody>
</table>

- SERIAL LINE IN CURRENT LOOP (option .CLOOP)

<table>
<thead>
<tr>
<th>J1</th>
<th>J2</th>
<th>J4</th>
<th>J5, J6</th>
<th>IC5</th>
<th>IC7</th>
<th>IC8</th>
<th>IC6</th>
<th>IC9</th>
</tr>
</thead>
<tbody>
<tr>
<td>indifferent</td>
<td>indifferent</td>
<td>position 2-3</td>
<td>not connected</td>
<td>no device</td>
<td>no device</td>
<td>no device</td>
<td>no device</td>
<td>no device</td>
</tr>
</tbody>
</table>

Please remark that current loop serial interface is passive, so it must be connected an active current loop serial line, that is a line provided with its own power supply, like described in figures 13+15. Current Loop interface can be employed to make both point-to-point and multi-point connections through a 2-wires or a 4-wires connection.

- SERIAL LINE IN RS 422 (option .RS 422)

<table>
<thead>
<tr>
<th>J1</th>
<th>J2</th>
<th>J4</th>
<th>J5, J6</th>
<th>IC5</th>
<th>IC7</th>
<th>IC8</th>
<th>IC6</th>
<th>IC9</th>
</tr>
</thead>
<tbody>
<tr>
<td>position 2-3 and 4-5</td>
<td>position 1-2</td>
<td>position 2-4</td>
<td>(*)</td>
<td>no device</td>
<td>driver SN 75176 or MAX 483</td>
<td>driver SN 75176 or MAX 483</td>
<td>no device</td>
<td>no device</td>
</tr>
</tbody>
</table>

RS 422 electric protocol can be used to make 4-wires full duplex connections both multi-point and point -to-point. Transmitter abilitation, essential in networks, is managed directly by MDU 01 selecting the master-slave logic protocol.
FIGURE 39: DRIVER FOR SERIAL COMMUNICATION SELECTION

Serial line in RS 232
Serial line in current loop
Serial line in RS 422
Serial line in RS 485
- SERIAL LINE IN RS 485 (option RS 485)

<table>
<thead>
<tr>
<th>IC</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IC5</td>
<td>= no device</td>
</tr>
<tr>
<td>IC7</td>
<td>= driver SN 75176 or MAX 483</td>
</tr>
<tr>
<td>IC8</td>
<td>= no device</td>
</tr>
<tr>
<td>IC6</td>
<td>= no device</td>
</tr>
<tr>
<td>IC9</td>
<td>= no device</td>
</tr>
</tbody>
</table>

In this modality the signals to use are pins 1 and 2 of connector CN1, that become transmission or reception lines according to the status managed by firmware, configured with logic protocol master-slave.

RS 485 electric protocol can be used to make 2-wires half duplex connections both multi-point networks and point-to-point.

(*) If using the RS 422 or RS 485 serial line, it is possible to connect the terminating circuit on RS 485 and RS 422 lines by using respectively J5 and J6.

This circuit must be always connected in case of point-to-point connections, while in case of multi-point connections it must be connected only in the farthest boards, that is on the edges of the communication line.

During a reset or a power on, driver RS 485 is in reception or transmission driver RS 422 is disabled, to avoid conflicts on line.

For further information about serial communication please refer to the examples of figures 9÷15.
SOFTWARE DESCRIPTION

As already said MDU 01 terminal is a complete video terminal and for this reason any characters received from serial line, if it is not a command, is shown on the display and codes of any pressed external keys, are transmitted to the control master unit.
These operations are automatically performed by on board firmware that is programmed and executed by the MDU 01 CPU.
The on board firmware manages also a local set up which allows to set the physical communication protocol by using the keys and the display.
This manual contains, in addition to the description of the different functions, a complete list of the recognized command sequences, to be used to benefit of the main features of MDU 01.
For each code or codes sequence, there is a double description i.e: the mnemonic one through the ASCII characters and the numerical one under decimal and hexadecimal form.
The said commands respect the ADDS View Point standard so all the sequences begin with ESC character corresponding to the 27 decimal code (1B Hex).

LOCAL SET UP

Thanks to a proper local set up mode, some parameters of communication protocol and the key click mode can be set by the user with the simple use of 4 keys.
To enter set up mode the user must power on the MDU 01 and simultaneously he must press the keys number 0 and number 11 (please refer to figure 28) for at least half of a second.
When the set up mode is entered, two external LEDs are activated, on the display appears the “-Setup-” or “** Local Setup **” string and with keys number 5 and number 6 the configuration parameters shall be changed.
Please remark that, according to firmware version, there will be a different number of options, as below described:

EXPANDED Firmware

Number 5:
"COMMUNICATION" changes current menu, recognized by the following messages: to change the communication type
"BAUD RATE" to change the communication baud rate
"STOP BIT" to change the stop bit number
"KEYCLICK" to change the keyclick mode
"NAME (Hex)" first figura of hexadecimal identification name
"NAME (Hex)" second figura of hexadecimal identification name

Number 6:
COMMUNICATION: Changes current value of menu, with the following possibilities:
BAUD RATE: Norm. or M.-S. for normal or master-slave protocol (def.=Norm.)
STOP BIT: 187500, 38400, 19200, 96000, 48000, 24000 or 1200 (def.=19200)
KEYCLICK: 1 or 2 with normal protocol (def.=1)
NAME (Hex): 1 with master-slave protocol
ON or OFF (def.=ON)
Changes the figure enclosed in "<>" from 0 to F (def.=01H)
NORMAL FIRMWARE

Number 5: Changes current menu, recognized by the following messages:
- "BAUD" to change the communication baud rate
- "STOP" to change the stop bit number
- "BEEP" to change the keyclick mode

Number 6: Changes current value of menu, with the following possibilities:
- BAUD: 38400, 19200, 9600, 4800, 2400 or 1200 Baud \((\text{def}=19200)\)
- STOP: 1 or 2 \((\text{def}=1)\)
- BEEP: ON or OFF \((\text{def}=ON)\)

With both firmwares, pressing keys number 0 and 11 at the same time will save settings into EEPROM and keep them up to next execution of local setup; then the terminal enters its normal working mode.

The default values before reported are those setted at the end of testing phase, that is the configuration the user receives.

Available options for menus BAUD RATE and STOP BIT define the physical communication protocol with one parameters unchangeable and set to no parity:
- 8 bit per character -> with normal firmware
- 8 bit per character -> expanded firmware and normal protocol
- 9 bit per character -> expanded firmware and master slave protocol

Option of remaining menus are described in the following paragraphs.

NOTE
Please remind that set up mode can be entered only during power up, when previously described condition are recognized in fact if keys are pressed at the same time during normal operation the set up mode will not start and the code of the pressed keys will be transmitted on the serial line.

The local set up is normally executed only one time after the first installation, so the required four external keys can be connected only during this phase and thereafter the MDU 01 can be used without keys as a simple display unit.

RECEIVE BUFFER

MDU 01 is provided with a reception buffer that simplifies the management, in fact it reduces the waiting time of the connected master unit.

Each received character is immediately saved inside this buffer (30 bytes long) and processed at the end of the currently executed operation.

Naturally when commands that require a long execution time (delete commands, EEPROM management commands, etc.) are continuously received, the buffer will become full and will overflow.

When overflow occurs last location of the buffer is overwritten by each next received characters, and these are definitively lost.

The master unit must stop the transmission until the MDU 01 has emptied the receive buffer and it is still ready to receive other data.

In practice the user must insert suitable delays between the commands transmission, to leave sufficient time to MDU 01 for executing the required operations and to avoid the complete filling of reception buffer.
CHARACTERS VISUALIZATION ON THE DISPLAY

MDU 01 shows on its display all the received characters having a code included in the range $32 \div 255$ ($20 \div FF$ Hex) but the one that identifies a command sequence ($27 = 1BH$). The character is visualized on the current cursor position and this latter will go to the next position; if it is placed on the last character of the display (right down corner), it will be placed on home position (left up corner). The correspondence between codes and displayed characters is defined by the following rules:

<table>
<thead>
<tr>
<th>Codes</th>
<th>Characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0 \div 31$ ($00 \div 1F$ Hex)</td>
<td>No representation</td>
</tr>
<tr>
<td>$32 \div 127$ ($20 \div 7F$ Hex)</td>
<td>Standard ASCII</td>
</tr>
<tr>
<td>$128 \div 255$ ($80 \div FF$ Hex)</td>
<td>Special and different according with installed display</td>
</tr>
</tbody>
</table>

About special characters please refer to APPENDIX B and remind that it is possible to get different display models, provided of different special characters, but everything must be directly arranged with grifo®.

COMMUNICATION MODALITIES

MDU 01 features two different serial communication modalities:

**Norm.**  
Normal communication uses 8 bits per character, no parity, stop bit and baud rate can be set by the user through local setup. This communication mode is suitable for connections point-to-point in RS 232, RS 422 and current loop.

**M.-S.**  
Master Slave communication uses 9 bits per character, no parity, one stop bit and baud rate can be set by the user. This communication mode is suitable for connections point-to-point (all electric protocols) on network (with protocols RS 485, RS 422 and current loop). For further information about master slave mode, please refer to next paragraph.

Local set up allows to select communication modality, as described in the specific paragraph, while electric protocol must be defined when the terminal is ordered.

MASTER SLAVE COMMUNICATION MODE

The Master Slave mode uses the 9 bits communication technique. In addition to the 8 data bit also a 9th bit is managed as it is needed for recognizing between a call coming from the "Master" to any of the "Slave" structures and a simple info transmission between Master and the selected device. When 9th bit is placed at 1, the data byte has to contain the name, or identifying code, of the device towards it needs to communicate, while by placing this particular bit at 0, it is possible to take out or supply info at this device.
When MDU 01 is used, the identifying code must be that one set by the local set up programm on the "NAME (Hex)" entry.
When this byte is sent (with 9th bit set to 1) the MDU 01 recognizes itself and it waits the string containing chars, data or commands. In this string there must only be a command that involves the return of an information sent via serial line from MDU 01 part; if there is more than one command the results of the remaining ones are ignored.
Between the transmission of a char. and the next one there must be an interval of time shorter than the Time Out, as elapsed this delay, the MDU 01 will consider the data string ended and it will begin the answering phase. The Time Out value for each baud rate is below described:

<table>
<thead>
<tr>
<th>Baud Rate</th>
<th>Time Out</th>
<th>Character transmission time</th>
</tr>
</thead>
<tbody>
<tr>
<td>187500 Baud</td>
<td>110 µsec</td>
<td>59 µsec</td>
</tr>
<tr>
<td>38400 Baud</td>
<td>550 µsec</td>
<td>287 µsec</td>
</tr>
<tr>
<td>19200 Baud</td>
<td>990 µsec</td>
<td>573 µsec</td>
</tr>
<tr>
<td>9600 Baud</td>
<td>1540 µsec</td>
<td>1146 µsec</td>
</tr>
<tr>
<td>4800 Baud</td>
<td>3080 µsec</td>
<td>2292 µsec</td>
</tr>
<tr>
<td>2400 Baud</td>
<td>6105 µsec</td>
<td>4584 µsec</td>
</tr>
<tr>
<td>1200 Baud</td>
<td>12100 µsec</td>
<td>9167 µsec</td>
</tr>
</tbody>
</table>

Master unit must wait for:

"character transission time"+"Time out"

before reaching the first character of the answering string returned by the MDU 01.
The answer consists in a byte containing the code of the pressed key (FF Hex, no key is pressed) or a data string related to a reading command sent in the previous request.
Please remark that answer is provided also in case the only identification name is requested, simplify the check for keys pressed or invalid commands.

To explain better the master slave protocol, here follows an example where master unit sends three commands to MDU 01 (reading of version number, a string to show and a check for eventual keys pressed) with baud rate 38.4 KBAud and identification name 80H:
Several demo programs, written in different languages, are provided with **MDU 01**. They implement master slave communication and can be used directly by the user or modified according to the specific needs.

When the master unit is a PC, the user can also take advantage of comfortable **DLL** libraries that allow to manage high level master slave communication, this means without having to worry about management of nineth bit, timings, eventual electric protocol converters, etc.

Also these libraries are provided with the first purchase, complete of user manual, on a CD.

**NOTES:**

1) To ensure right command execution, between a call and the next one it is necessary to wait for a time that is related to the number of commands sent and type of operations they involve.

### FIGURE 40: MASTER SLAVE COMMUNICATION EXAMPLE

<table>
<thead>
<tr>
<th><strong>Master</strong></th>
<th><strong>MDU 01</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sends “Reading of version number” command, that is the characters sequence: 80H with nineth bit set to 1 56H with nineth bit set to 0 delay between characters lower than 550 µsec</td>
<td>Receives character of the command and verifies the Time Out of 550 µ sec</td>
</tr>
<tr>
<td>Waits for 837 µsec</td>
<td>Recognizes command sequence, executes the command and stores answer for next command</td>
</tr>
<tr>
<td>Receives one character of answer</td>
<td>Sends the answer, which is the code of the eventual key pressed with nineth bit set to 0</td>
</tr>
<tr>
<td>Sends a string to show on the display, that is the character sequence: 80H with nineth bit set to 1 1° character of string with nineth bit set to 0 2° character of string with nineth bit set to 0 : : : : : : : delay between characters lower than 550 µ sec</td>
<td>Receives character of the command and verifies the Time Out of 550 µ sec</td>
</tr>
<tr>
<td>Waits for 837 µsec</td>
<td>Recognizes command sequence and shows on the display the characters of the string</td>
</tr>
<tr>
<td>Receives three characters of answer with the version number previously requested</td>
<td>Transmits saved response which is the version number required in previous command, with nineth bit set 0</td>
</tr>
<tr>
<td>Sends key pressed check command, that is the character sequence: 80H with nineth bit set to 1</td>
<td>Receives character of the command and verifies the Time Out of 550 µ sec</td>
</tr>
<tr>
<td>Waits for 837 µsec</td>
<td>Recognizes sequence without commands so performs no operation</td>
</tr>
<tr>
<td>Receives one character of answer corresponding to code of eventual key pressed</td>
<td>Sends the answer, which is the code of the eventual key pressed with nineth bit set to 0</td>
</tr>
</tbody>
</table>
2) If the Master unit cannot communicate using 9 bits, it is possible to simulate this communication mode by using the parity bit and programming its value opportunistically, before any characters transmission, according to this scheme:

**If the character to transmit has EVEN number of "1" bits**
- If $9^{th}$ bit must be 1
  - Set parity to ODD
- If $9^{th}$ bit must be 0
  - Set parity to EVEN

**If the character to transmit has ODD number of "1" bits**
- If $9^{th}$ bit must be 1
  - Set parity to EVEN
- If $9^{th}$ bit must be 0
  - Set parity to ODD

3) If the scrolling messages mode is enabled, the time between two calls, in addition to the time indicated at point 1, must be about 12000 µsec.

**KEYBOARD ACQUISITION**

When MDU 01 recognizes an external key pressure, if normal communication is used it translates it and then transmits the determined code on serial line, by using the rules described in the following paragraph.

If master slave communication is used the key code is sent only upon reception of specific request of master unit, as described in the previous paragraph.

Moreover an auto repeat function of the stroked key is implemented so when MDU 01 recognizes the pressure on a key for a time greater than 0.5 sec, it will start the serial transmission of its code about each 0.1 sec, and it lasts until that specific key is released.

If the keyclick function is enabled when the code of the pressed key is transmitted, the on board buzzer also generates a loud beep that sonorously signalize the event to the user.

**KEYS CODES**

Here are the table which shows the codes that MDU 01 sends on serial line when a key is pressed; the code here is shown in decimal, hexadecimal and ASCII mnemonic format, like for command sequences:
These codes are those transmitted under default condition, i.e. the configuration the user receives, but they can be comfortably reconfigured by using a specific command. This feature really simplifies the management software development in fact the master unit can change the codes according with his requirements and it can also disable the keys.

<table>
<thead>
<tr>
<th>KEY N°</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>65</td>
<td>41</td>
<td>A</td>
</tr>
<tr>
<td>1</td>
<td>66</td>
<td>42</td>
<td>B</td>
</tr>
<tr>
<td>2</td>
<td>67</td>
<td>43</td>
<td>C</td>
</tr>
<tr>
<td>3</td>
<td>68</td>
<td>44</td>
<td>D</td>
</tr>
<tr>
<td>4</td>
<td>69</td>
<td>45</td>
<td>E</td>
</tr>
<tr>
<td>5</td>
<td>70</td>
<td>46</td>
<td>F</td>
</tr>
<tr>
<td>6</td>
<td>71</td>
<td>47</td>
<td>G</td>
</tr>
<tr>
<td>7</td>
<td>72</td>
<td>48</td>
<td>H</td>
</tr>
<tr>
<td>8</td>
<td>73</td>
<td>49</td>
<td>I</td>
</tr>
<tr>
<td>9</td>
<td>74</td>
<td>4A</td>
<td>J</td>
</tr>
<tr>
<td>10</td>
<td>75</td>
<td>4B</td>
<td>K</td>
</tr>
<tr>
<td>11</td>
<td>76</td>
<td>4C</td>
<td>L</td>
</tr>
<tr>
<td>12</td>
<td>77</td>
<td>4D</td>
<td>M</td>
</tr>
<tr>
<td>13</td>
<td>78</td>
<td>4E</td>
<td>N</td>
</tr>
<tr>
<td>14</td>
<td>79</td>
<td>4F</td>
<td>O</td>
</tr>
<tr>
<td>15</td>
<td>80</td>
<td>50</td>
<td>P</td>
</tr>
<tr>
<td>16</td>
<td>81</td>
<td>51</td>
<td>Q</td>
</tr>
<tr>
<td>17</td>
<td>82</td>
<td>52</td>
<td>R</td>
</tr>
<tr>
<td>18</td>
<td>83</td>
<td>53</td>
<td>S</td>
</tr>
<tr>
<td>19</td>
<td>84</td>
<td>54</td>
<td>T</td>
</tr>
<tr>
<td>20</td>
<td>85</td>
<td>55</td>
<td>U</td>
</tr>
<tr>
<td>21</td>
<td>86</td>
<td>56</td>
<td>V</td>
</tr>
</tbody>
</table>
COMMANDS FOR CURSOR POSITION

Here follows the list of the cursor positioning commands.

CURSOR LEFT

*Code:* 21  
*Hex code:* 15  
*Mnemonic:* NACK

The cursor is shifted of one position to the left without modifying the display contents. If the cursor is in Home position, it will be placed in the last position of the last row of the display.

CURSOR RIGHT

*Code:* 6  
*Hex code:* 6  
*Mnemonic:* ACK

The cursor is shifted of one position to the right. If the cursor is placed in the last position of the last row, it will be moved to the Home position.

CURSOR DOWN

*Code:* 10  
*Hex code:* A  
*Mnemonic:* LF

The cursor will be moved to the line below but it will remain in the same column. If the cursor is in the last display line, it will be moved to the first display line.

CURSOR UP

*Code:* 26  
*Hex code:* 1A  
*Mnemonic:* SUB

The cursor will be moved to the line above but it will remain in the same column. If the cursor is in the first display line, it will be moved to the last display line.

HOME

*Code:* 1  
*Hex code:* 1  
*Mnemonic:* SOH

The cursor is moved to Home position i.e first line, first column of the display, or on the other hand the up, left corner.
CARRIAGE RETURN

| Code:    | 13 |
| Hex code: | D |
| Mnemonic: | CR |

The cursor is moved to the beginning of the line where it was located.

CARRIAGE RETURN + LINE FEED

| Code:    | 29 |
| Hex code: | 1D |
| Mnemonic: | GS |

The cursor is moved to the beginning of line above the one where it was located.
If the cursor is at the last display line, it will be moved to the beginning of the first line i.e Home position.

ABSOLUTE CURSOR PLACEMENT

| Code:    | 27 89 r c |
| Hex code: | 1B 59 r c |
| Mnemonic: | ESC Y ASCII(r) ASCII(c) |

The cursor is moved to the absolute position indicated by r and c.
These characters are the row and column values of the new desired position referred to coordinate 0, 0 of the Home position, plus a constant offset of 32 (20 Hex).
If, for example, the user wants to place the cursor on the second line, third column (row 1, column 2), the following byte sequence must be sent:

27 89 33 34 or 1B 59 21 22 Hex or ESC Y ! "

If row and/or column values are not compatible with the installed display, the command is ignored.
COMMANDS FOR CHARACTERS ERASURE

Below are described all the commands that deletes one or more characters from the display.

BACKSPACE

Code: 8
Hex code: 8
Mnemonic: BS

This command moves the cursor one character position to the left and it erase the contents of the reached cell.
If the cursor is in Home position, it will be erased the last character of the last row of the display.

CLEAR PAGE

Code: 12
Hex code: C
Mnemonic: FF

This command clears all data on the display and it moves the cursor to Home position.

CLEAR LINE

Code: 25
Hex code: 19
Mnemonic: EM

This command erases all characters displayed on the current line and it moves the cursor to the first column of the said line.

CLEAR END OF LINE

Code: 27 75
Hex code: 1B 4B
Mnemonic: ESC K

This command erases all characters displayed from the current cursor position to the end of line inclusive. The cursor mantains the previous position.
If, for example, the cursor is at the beginning of a display line, the complete line will be erased.

CLEAR END OF PAGE

Code: 27 107
Hex code: 1B 6B
Mnemonic: ESC k

This command erases all characters displayed from the current cursor position to the end of display inclusive. The cursor mantains the previous position.
If, for example, the cursor is at Home position, the complete display will be erased.
COMMANDS FOR CURSOR ATTRIBUTES MANAGEMENT

Below are listed the command that define the possible cursor attribute. Please remark that cursor can be visible only in alphanumeric mode; in graphic mode it is managed but not shown. It is possible, anyway, to define position and style for cursor also for each position of graphic display using alternatively graphic and alphanumeric commands.

CURSOR OFF

<table>
<thead>
<tr>
<th>Code:</th>
<th>27  80</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>1B 50</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>ESC P</td>
</tr>
</tbody>
</table>

The cursor is not active and it is not more visible.

STEADY STATIC CURSOR ON

<table>
<thead>
<tr>
<th>Code:</th>
<th>27  79</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>1B 4F</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>ESC O</td>
</tr>
</tbody>
</table>

The cursor is activated so it is visible. Now it is a not blinking line placed under the current position character.

BLINKING BLOCK CURSOR ON

<table>
<thead>
<tr>
<th>Code:</th>
<th>27  81</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>1B 51</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>ESC Q</td>
</tr>
</tbody>
</table>

The cursor is enabled and it is visible. The selected cursor type is a blinking rectangular block that is alternatively visualized with the character displayed on the current cursor position.

BLINKING "BLOCK" CURSOR

<table>
<thead>
<tr>
<th>Code:</th>
<th>27  81</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>1B 51</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>ESC Q</td>
</tr>
</tbody>
</table>

The cursor is started so it is visible. Now it is a blinking rectangular form and it is alternatively visualized with the char put on the same.

Note: This command is available only for LCD displays otherwise is ignored.
COMMANDS FOR EEPROM

In the following paragraphs are described all the commands that manage the data saved on MDU 01 on board EEPROM; there are other commands that indirectly use this memory device but they are described in next paragraphs.

REQUEST FOR EEPROM WRITING POSSIBILITY

Code: 27 51
Hex code: 1B 33
Mnemonic: ESC 3

This command checks if the MDU 01 is ready for writing data on its on board EEPROM. This command must be executed any time there are data to be saved on this type of memory.

When MDU 01 receives this command, it answers with the following codes:

6 (06 Hex) (ACK) -> MDU 01 ready
21 (15 Hex) (NACK) -> MDU 01 not ready

If the QTP 12 sends back the NACK code, it is not yet possible to memorize a new data on EEPROM.

WRITING OF PRESENCE BYTE

Code: 27 33 78 byte
Hex code: 1B 21 4E byte
Mnemonic: ESC ! N ASCII(byte)

This command sets the card presence byte with the value indicated in the byte parameter that must be included in 0÷255 range.

This byte has a reserved allocation on the on board EEPROM that, once it is set with the desired value, it allows for example, to verify that MDU 01 runs correctly, or if there are some communication problems on the serial line.

NOTE: This command writes data on the on board EEPROM, so before executing it is better to check the EEPROM writing possibility through the proper command; in fact if it is not ready the command is ignored.

READING OF PRESENCE BYTE

Code: 27 33 110
Hex code: 1B 21 6E
Mnemonic: ESC ! n

The QMDU 01 sends back on the serial line the value of its presence byte.

For example, this command can be useful to verify the presence or the correct running of the terminal.

NOTE: This command uses the on board EEPROM, so before executing it is better to check the EEPROM ready; in fact if it is not ready the command is ignored and 21 (15Hex) = NAK is returned.
COMMANDS FOR GENERAL FUNCTIONS

In the following paragraphs are described all the general purpose commands that manage some of the MDU 01 features.

READING OF VERSION NUMBER

| Code:       | 27  86 |
| Hex code:   | 1B  56 |
| Mnemonic:   | ESC  V |

On the serial line is returned a string of 3 characters containing the program managing version that is resident and executed by MDU 01. For example with firmware version 1.3 the following characters will be transmitted:

50  46  48  or  32  2E  30  Hex  or  2.0

BEEP

| Code:       | 7 |
| Hex code:   | 7 |
| Mnemonic:   | BEL |

The buzzer is enabled for a time of 0.1 second. If buzzer was already enable then it is disabled, so the effect of this command is always recognizable.

RELAY ACTIVATION

| Code:       | 27  56 |
| Code Hex:   | 1B  38 |
| Mnemonic:   | ESC  8 |

The relay on the card, if any, is enabled and the normal open contact is shut (N.O.) and the normal shut contact is open (N.C.)

NOTE: This command works only on MDU 01 with .RELAY option.

RELAY DEACTIVATION

| Code:       | 27  57 |
| Code Hex:   | 1B  39 |
| Mnemonic:   | ESC  9 |

The relay on the card, if any, is desabled and the normal shut contact is shut (N.C.) and the normal open contact is open (N.O.)

NOTE: This command works only on MDU 01 with .RELAY option.
COMMANDS FOR LEDS MANAGEMENT

In the following paragraphs are described all the commands that manage LEDs of the MDU 01. Please remark that LEDs enumeration allows to identify each LED as unique; please refer to figure 29 for the correspondence.

LED ACTIVATION

Code: 27 50 n.LED Attr.
Hex code: 1B 32 n.LED Attr.
Mnemonic: ESC 2 ASCII(n.LED) ASCII(Attr.)

The LED shown in “n.LED” with the specified attribute in “Attr.” is started. LEDs numbers are included in a range of 0+21. The attributes available are as follows:

<table>
<thead>
<tr>
<th>LED Not Enabled</th>
<th>LED Enabled</th>
<th>LED Blinking</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>255 (FF Hex)</td>
<td>85 (55Hex)</td>
</tr>
</tbody>
</table>

ex. If you wish to enable LED n.5 with blinking attribute, the following sequence has to be sent:

27 50 5 85 or 1B 32 05 55 Hex or ESC 2 ENQ U

If the parameter with LED number or that one with the attribute, it is not valid, the command is ignored.

LEDS MASK ACTIVATION

Code: 27 52 byte1 byte2 byte3
Hex Code: 1B 34 byte1 byte2 byte3
Mnemonic: ESC 4 ASCII(byte1) ASCII(byte2) ASCII(byte3)

All LEDs are contemporarily managed as indicated in "byte1", "byte2" and "byte3" following this code:

byte1 (bit 0...7)       LED 0...LED 7
byte2 (bit 0...7)       LED 8...LED 15
byte3 (bit0...5)        LED 16...LED 21

If a bit is set to 0 position, the correspondent LED is OFF, viceversa it will be ON if the correspondent bit is set to 1.
If there are some LEDs having the blinking attribute, this latter will be disabled.
FIGURE 42: CARD PHOTO
COMMANDS FOR KEYBOARD MANAGEMENT

In the following paragraphs are described all the commands that manage external keyboard of the MDU 01.
For information about management and codes returned by the board at key pressure, please refer to paragraph “KEYBOARD ACQUISITION”.
Please remark that keys enumeration allows to identify each key as unique; please refer to figure 28 for the correspondance.

KEY RECONFIGURATION

Code: 27 55 key no. code
Hex Code: 1B 37 key no. code
Mnemonic: ESC 7 ASCII( key no.) ASCII(code)

When the selected key is reconfigured, each time it is pressed, the card will send the new specified code in serial mode.
The number of the key to be reconfigured is obtainable from figure 28 and it must be included in a range of 0÷21 (0÷15 Hex) if this is not done the command is ignored.
The code value can vary in a range of 0÷254 (0÷FE Hex) as the 255 value (FF Hex) indicates that the key must be disabled so when it is pressed the QTP will not send any codes in serial mode.
Figure 41 shows default key codes configuration, that is the one of the board delivered.

NOTE: The said command needs a data writing on the on-board EEPROM so before executing it, it is better to be sure that the card is ready for a new writing on such device otherwise the command will be ignored.

KEYCLICK ON WITHOUT MEMORIZATION

Code: 27 53
Hex Code: 1B 35
Mnemonic: ESC 5

The KeyClick function is switched on so there is a sound feedback when a key is pressed.
This parameter is not stored in the on-board EEPROM so if the card is reset it goes back to the previous condition.

KEYCLICK OFF WITHOUT MEMORIZATION

Code: 27 54
Hex Code: 1B 36
Mnemonic: ESC 6

The KeyClick function is disabled so there is not sound feedback when a key is pressed.
This parameter is not stored in the on-board EEPROM so if the card is reset it goes back to the previous condition.
KEYCLICK ON WITH MEMORIZATION

Code: 27 33 53
Hex Code: 1B 21 35
Mnemonic: ESC ! 5

The KeyClick function is switched on so there is a sound feedback when a key is pressed. This parameter is stored in the on-board EEPROM even if the power voltage fails.

NOTE:
The said command involves a data writing on the on-board EEPROM so before executing it, it is better to be sure that the card is ready for a new writing on such device otherwise the command will be ignored.

KEYCLICK OFF WITH MEMORIZATION

Code: 27 33 54
Hex Code: 1B 21 35
Mnemonic: ESC ! 6

The KeyClick function is disabled so there is not sound feedback when a key is pressed. This parameter is stored in the on-board EEPROM even if the power voltage fails.

NOTE:
The said command needs a data writing on the on-board EEPROM so before executing it, it is better to be sure that the card is ready for a new writing on such device otherwise the command will be ignored.
COMMANDS FOR MESSAGE MANAGEMENT

In the following paragraphs are described all the commands that manage messages on MDU 01. The messages are 20 characters sequence that can be saved on board EEPROM and then reloaded or represented on display, simply by supplying the same message identification number. The most important function of messages is the possibility to show constant information on the display (i.e. allarms, equipment status, etc.) without the transmission of the numerous characters of this information but only the few characters of the commands. Furthermore a comfortable program for PC, named QTP EDIT, allows any user to edit the messages, save and load them on PC disks and transmit/receive them directly to/from MDU 01 serially connected to PC.

MDU 01 features EEPROM from 256 to 4096 bytes.

By default EEPROM size is 512 bytes, while the mid sized chip (2048 bytes) can be ordered using the option .MEX.

Other options should be discussed directly with grifo®.

READING OF THE LAST STORAGED MESSAGE NUMBER

| Code: | 27 110 |
| Hex code: | 1B 6E |
| Mnemonic: | ESC n |

This command returns on the serial line the number of the last messages that can be saved on EEPROM. It depends on the size of EEPROM installed according to the following correspondence:

<table>
<thead>
<tr>
<th>Version</th>
<th>EEPROM Size</th>
<th>N°. last message</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>256 Bytes</td>
<td>9 (9 Hex)</td>
</tr>
<tr>
<td>-</td>
<td>512 Bytes</td>
<td>22 (16 Hex)</td>
</tr>
<tr>
<td>.MEX</td>
<td>2048 Bytes</td>
<td>99 (63 Hex)</td>
</tr>
<tr>
<td>-</td>
<td>2304 Bytes</td>
<td>111 (6F Hex)</td>
</tr>
<tr>
<td>-</td>
<td>2560 Bytes</td>
<td>124 (7C Hex)</td>
</tr>
<tr>
<td>-</td>
<td>4096 Bytes</td>
<td>201 (C9 Hex)</td>
</tr>
</tbody>
</table>

FIGURE 43: NUMBER OF MESSAGES THAT CAN BE STORED IN EEPROM

MESSAGE STORAGE

| Code: | 27 33 67 mess. num. chr. 0... chr.19 |
| Hex code: | 1B 21 43 mess. num. chr. 0... chr.13 Hex |
| Mnemonic: | ESC ! C ASCII(mess. num. ) ASCII(chr.0)...ASCII(chr.19) |
This command stores the 20 characters message, with number indicated as mess. num., on the on board EEPROM.
The 20 chars which form the message must be visualizable on the display so they must be included in the range $32+255$ ($20+FF$ Hex).
The message number must be included in the range of $0+96$.

**NOTE:** This command writes data on the on board EEPROM, so before executing it is better to check the EEPROM writing possibility through the proper command; in fact if it is not ready the command is ignored.

### MESSAGE READING

**Code:** 27 33 69 mess. num.

**Hex code:** 1B 21 45 mess. num.

**Mnemonic:** ESC ! E ASCII(mess. num.)

This command reads the 20 characters message corresponding to mess. num. by the EEPROM and it sends this message on serial line, beginning from the first char of the string.

At the end of the message, the CR+LF codes are sent, too.

The message number must be included in the range of $0+n. max$, as seen previously, if this number is out of range, the command is ignored.

**NOTE:** This command uses the on board EEPROM, so before executing it check the EEPROM ready; in fact if it is not ready the command is ignored and 21 (15 Hex) = NAK is returned.

### MESSAGE VISUALIZATION

**Code:** 27 33 68 mess. num. n

**Hex code:** 1B 21 44 mess. num. n

**Mnemonic:** ESC ! D ASCII(mess. num.) ASCII(n)

This command visualizes n 20 characters messages on the display, beginning from current cursor position.

The first of the n messages is that one having the number corresponding to mess. num. while the remaining messages are those ones immediately subsequents in EEPROM.

The message number must be included in the range of $0+n. max$, as seen previously, if this number is out of range, the command is ignored.

The n quantity of messages to be visualized depends only on the model of the display:

- **Display 20x2 or 40x1:** n in the range 1+2
- **Display 20x4 or 40x2:** n in the range 1+4
- **Display 40x4:** n in the range 1+8

If value of n is out of this range, command is ignored.

The cursor is placed in the next position of the last character visualized; if the last character of the said message occupies the last position of the display, the cursor will be placed in home position.

For example; to visualize the messages number 10 and 11, it will be necessary to send the following sequence:
NOTE 1: This command uses the on board EEPROM, so before executing it check the EEPROM ready; in fact if it is not ready the command is ignored and 21 (15 Hex) = NAK is returned.

NOTE 2: Visualization of messages on Futaba fluorescent display has a duration that depends on display size. Here follow the times for a message set that can fill the whole display:

- Display 20x2 or 40x1 (2 messages): About 8 msec
- Display 40x2 (4 messages): About 16 msec
- Display 40x4 (8 messages): About 32 msec

SCROLLING MESSAGES VISUALIZATION

Code: 27 33 83 mess.num. n.char
Hex code: 1B 21 53 mess.num. n.char
Mnemonic: ESC ! S ASCII(mess.num.) ASCII(n.char)

This command visualizes a n.char characters message on the display first line in sliding mode. The message is shifted from right to left and so the user can visualize a very long string on the display. The string of "n.char" characters begins with the first character of the "mess.num." message already stored in EEPROM and continues with next messages.

The "mess.num." value must be included in the range 0÷n. max, as previously seen, if the value is out of range this command is ignored. The "n.char" parameter is used as follow:

- 0 -> Stops the message visualization in sliding mode.
- 20÷200 -> If a display with 20 characters rows is connected.
- 40÷200 -> If a display with 40 characters rows is connected.

If "n.char." value is out of the specified ranges or it points after the last character stored in EEPROM, the command will be ignored.

The message visualization in sliding mode is positioned on the first display line and the cursor position and attributes are held.

For example, if you wish to visualize a 35 characters string in sliding mode, formed by message 10 (20 characters) and by the first 15 characters of message 11, it will be necessary to send the following sequence:

27 33 83 10 35 or 1B 21 53 0A 23 Hex or ESC ! S LF #

NOTE: This command uses the on board EEPROM, so before executing it check the EEPROM ready. The message visualization in sliding mode is managed in background and so there is a slowing down of serial data interpretation. This is the reason why it is necessary to wait for few msec between the transmission of 20÷30 bytes data blocks. In this way misunderstanding in interpreting the received data is completely void.
COMMANDS FOR MANAGING THE ON-BOARD REAL TIME CLOCK

Here follow commands for Real Time Clock management. Please remark that these commands have meaning only on MDU 01 with option .RTC installed.

WRITING OF A BYTE OF THE REAL TIME CLOCK SRAM

Code: 27 33 71 addr byte
Hex code: 1B 21 47 addr byte
Mnemonic: ESC ! G ASCII(addr) ASCII(byte)
On-board Real Time Clock SRAM byte it is written with the value shown in "byte". The address where memorize the said byte is shown at "addr" and it must be included in the range 32...255 (20...FF Hex) otherwise the command is ignored.

READING OF A BYTE OF THE REAL TIME CLOCK SRAM

Code: 27 33 103 addr byte
Hex code: 1B 21 67 addr byte
Mnemonic: ESC ! g ASCII(addr) ASCII(byte)
The on-board Real Time Clock SRAM byte placed at the address shown at "addr" is sent back via serial line. This byte must be included in the range 32...255 (20..FF Hex) otherwise the command is ignored.

CLOCK SET-UP

Code: 27 33 70 hour min sec day mon yea week
Hex Code: 1B 21 46 hour min sec day mon yea week
Mnemonic: ESC ! F ASCII(hour) ASCII(min) ASCII(sec)
ASCII(day) ASCII(mon) ASCII(yea) ASCII(week)
The on-board Real Time Clock is set with the data contained in the 6 parameters; if one of these ones has a value included in the allowed range, the RTC allocation is not programmed. Here under is listed the meaning of the 7 bytes of above and their allowed range.

hour (0÷23) -> Hour
min (0÷59) -> Minute
sec (0÷59) -> Second
day (1÷31) -> Day of month
mon (1÷12) -> Month
yea (0÷99) -> Year
week (0÷6) -> Day of week: 0 = Sunday
              : 6 = Saturday
CLOCK READING

Code: 27 33 102
Hex Code: 1B 21 66
Mnemonic: ESC ! f

The 7 values that the on-board Real Time Clock manages (hour, minute, second, day, month, year and day of week) are returned.
The meaning of these bytes is explained in paragraph “CLOCK SETTINGS”.

TIME VISUALIZATION ON THE DISPLAY

Code: 27 33 116 r c frm
Hex Code: 1B 21 74 r c frm
Mnemonic: ESC ! t ASCII(r) ASCII(c) ASCII(frm)

The time drawn from the on-board Real Time Clock is displayed starting from the position shown by the "r" and "c" bytes. These codes express the raw and column values of the display at which an offset of 32 (20 Hex) must be added. If the raw and column values are not compatible to the display installed, this command will be ignored. The "frm" parameter is used to specify the visualization format, in particular way:

Bit 0 -> 1 The time is visualized and automatically managed in "r", "c" position.
0 The visualization of the time is interrupted.

Bit 1 -> 1 The time is visualized in AM/PM format: HH:MM:SSm i.e.: HH (hours), MM (minutes), SS (seconds), m (a=AM or p=PM).
0 The time is visualized in 24H format: HH:MM:SS i.e.: HH (hours), MM (minutes), SS (seconds).

Bit 2÷7 -> 0 Reserved for future expansion. They must be set to "0" value.

For example, if you wish to visualize the time starting from 3rd row and 6th column in 24 H format, it will be necessary to send the following sequence:
27 33 116 34 37 1 or 1B 21 74 22 25 01 Hex or ESC ! t " % SOH

Note: The time drawn is managed in background and so there is a slowing down of serial data interpretation. This is the reason why it is necessary to wait for few msec between the transmission of 20÷30 bytes data blocks. In this way misunderstanding in interpreting the received data is completely void.
DATE VISUALIZATION ON THE DISPLAY

Code: 27 33 100 r c frm
Hex code: 1B 21 64 r c frm
Mnemonic: ESC ! d ASCII(r) ASCII(c) ASCII(frm)

The date drawn from the on-board Real Time Clock is displayed starting from the position shown by the "r" and "c" bytes. These codes express the raw and column values of the display at which an offset of 32 (20 Hex) must be added. If the raw and column values are not compatible to the display installed, this command will be ignored.

The "frm" parameter is used to specify the visualization format, in particular way:

Bit 0 -> 1 The date is visualized and automatically managed in "r", "c" position.
0 The visualization of the date is interrupted.

Bit 1 -> 1 The format that visualizes the date is as follows: MM-DD-YY, i.e MM (month), DD (day), YY (year).
0 The format that visualizes the date is as follows: DD-MM-YY, i.e DD (day), MM (month), YY (year).

Bit 2÷7 -> 0 Reserved for future expansion. They must be set to "0" value.

Please note that the week day is not managed.
For example, if you wish to visualize the date starting from 4th row, 12th column in MM-DD-YY format, it will be necessary to send the following sequence:

27 33 100 35 43 3 or 1B 21 64 23 2B 03 Hex or ESC ! d # + ETX

NOTE: The date drawn is managed in background and so there is a slowing down of serial data interpretation. This is the reason why it is necessary to wait for few msec between the transmission of 20÷30 bytes data blocks. In this way misunderstanding in interpreting the received data is completely void.
COMMANDS FOR BADGE READER MANAGEMENT

MDU 01 board is able to acquire single track magnetic badge readers. These latter can be both insertion and manual types, able to read track n.1, 2 or 3 (please refer to figures 24 and 25 for more information).

MDU 01 by means of a specific command can be set in compliance with the badge reader at which it is connected.

Only passive readers can be connected; this means only reader provided with reading head and capable to generate TTL signals for communication with MDU 01.

This kind of readers are surely the cheapest, smallest and easy to use in commerce; here follows a short list of them:

<table>
<thead>
<tr>
<th>TYPE</th>
<th>MODEL</th>
<th>MANUFACTURER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual; sliding; track 1</td>
<td>3S4YR-HSR3</td>
<td>OMRON</td>
</tr>
<tr>
<td>Manual; sliding; track 2</td>
<td>3S4YR-HSR4</td>
<td>OMRON</td>
</tr>
<tr>
<td>Manual; sliding; track 3</td>
<td>3S4YR-HSR5</td>
<td>OMRON</td>
</tr>
<tr>
<td>Manual; sliding; tracks 1, 2</td>
<td>3S4YR-HSR6</td>
<td>OMRON</td>
</tr>
<tr>
<td>Manual; sliding; tracks 2, 3</td>
<td>3S4YR-HSR7</td>
<td>OMRON</td>
</tr>
<tr>
<td>Manual; insertion; track 1</td>
<td>3S4YR-SBR3-20</td>
<td>OMRON</td>
</tr>
<tr>
<td>Manual; insertion; track 2</td>
<td>3S4YR-SBR4-20</td>
<td>OMRON</td>
</tr>
<tr>
<td>Manual; insertion su track 3</td>
<td>3S4YR-SBR5-20</td>
<td>OMRON</td>
</tr>
<tr>
<td>Motorized, insertion; track 1</td>
<td>3S4YR-MCR3</td>
<td>OMRON</td>
</tr>
<tr>
<td>Motorized; insertion; track 2</td>
<td>3S4YR-MCR4</td>
<td>OMRON</td>
</tr>
<tr>
<td>Motorized; insertion; track 3</td>
<td>3S4YR-MCR5</td>
<td>OMRON</td>
</tr>
<tr>
<td>Motorized; insertion; tracks 1, 2</td>
<td>3S4YR-MCR6</td>
<td>OMRON</td>
</tr>
<tr>
<td>Motorized; insertion; tracks 2, 3</td>
<td>3S4YR-MCR7</td>
<td>OMRON</td>
</tr>
<tr>
<td>Motorized; insertion; tracks 1, 2, 3</td>
<td>MTM-290-3A-2020</td>
<td>NEURON</td>
</tr>
</tbody>
</table>

**Figure 44: Badge Readers Connectable**

Following commands are used to perform any operation with all kinds of badge connectable. Of course they can have effect only if .BADGE option is installed.
NOTE:
When MDU 01 is configured to manage the automatic Badge Reader, it can manage 2 Kbytes of EEPROM max (instead of 4 Kbytes) and it can not manage the Real Time Clock. These latter options are so installable on MDU 01 only if these terminals do not manage the Badge Reader.

STANDARD RECOGNIZED

Firmware of MDU 01 can read magnetic badge compliant with the following international standards:

<table>
<thead>
<tr>
<th>Track</th>
<th>Standard</th>
<th>Normative</th>
<th>Saving method</th>
<th>Saving density</th>
<th>Characters configuration</th>
<th>Size (characters)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Track 1</td>
<td>I.A.T.A.</td>
<td>ISO 3554 ANSI X4.16-1976</td>
<td>F2F</td>
<td>210±5% bit x inch</td>
<td>7 bit with parity</td>
<td>79 (alphanumerics)</td>
</tr>
<tr>
<td>Track 2</td>
<td>A.B.A.</td>
<td>ISO 3554 ANSI X4.16-1976</td>
<td>F2F</td>
<td>75±3% bit x inch</td>
<td>5 bit with parity</td>
<td>40 (numerics)</td>
</tr>
<tr>
<td>Track 3</td>
<td>M.I.N.T.S.</td>
<td>ISO 4909 DIN 4909 ANSI X4.16-1976</td>
<td>F2F</td>
<td>210±5% bit x inch</td>
<td>5 bit with parity</td>
<td>107 (numerics)</td>
</tr>
</tbody>
</table>

Detailed below:

**Figure 45: Features of Badges Managed**

**Figure 46: Tracks Size of Badges Managed**
SYNTAX OF THE STRING READ FROM THE BADGE

MDU 01 stores the information of a reading operation and, if successful, also characters read from the card. If MDU 01 is configured to send immediately these characters to the serial line (bit 3 of configuration byte set to 0, as follows), this string will be sent:

\[ C2 \text{ Hex} \quad C2 \text{ Hex} \quad \text{cod} \quad \text{n.car.} \quad \text{car.1} \quad \ldots \quad \text{car } n \quad 0D \text{ Hex} \quad 0A \text{ Hex} \]

The first two characters of the string are hexadecimal value C2 repeated twice, to indicate that following information are related to badge reader. This particular code should not be linked to a key, to avoid misunderstanding in recognition. Should the association be unavoidable, the badge string can be recognized anyway measuring the time between the two characters.

Third byte, called **cod**, contains information about the last acquisition, in detail:

**Bit 7**  \( \rightarrow \)  1  These information are sent for the first time.

0  These information have already been sent at least once, so they have not been acquired from other badges.

This bit is useful in case a string is requested more than once, so it can be used to determine if it comes from another acquisition or is a repetition of last acquisition. Of course, if MDU 01 is configured to send immediately the string, in the first (autonomous) transmission this bit will always be 1.

**Bit 6+0**  \( \rightarrow \)  0  No string available now.

6  A valid and complete string has been received.

21  A string has been received, but CRC is not correct, or START character is missing.

26  A valid string has been received, but it is not complete because END character is missing.

With a sliding reader, a string is considered valid only if both START and END characters are recognized, so in this case code 26 will never be returned, because the badge must be always read completely. For an insertion reader, a partial acquisition may happen, so a string can be considered valid even if END character is not recognized.

Fourth byte, called **n. car.**, contains the number of characters acquired from magnetic badge; the characters will follow in the string, except for the last **CR+LF** sequence.

If the string is not valid this byte will be 0 and only **CR+LF** sequence will follow.

If the string is valid, after **n. car.**, ASCII codes of the characters acquired will follow, between START and END.

As said, the string will be terminated with a **CR+LF** sequence.
READING A BADGE FROM A MOTORIZED READER

When a badge is inserted in a motorized reader, **MDU 01** immediately acquires the string in it. In detail “forward motor” remains active until the whole badge has slid under the reading head, then the string acquired is interpreted. If a mistake in reading occurs, other two attempts of acquisition will take place before declaring that string not valid; during this phase the "reverse motor" is activated in order to re-position the initial part of the card to be read for the new reading. Once the reading procedure is over, the result of such reading is memorized and, sent in serial mode as it happens when it is question of the manual badge reader. The card stays inside the badge reader until the ejection command is sent. **MDU 01** has a further command that allows to "eat" the card; more exactly the "forward motor" is activated up to the card goes out from the back side of the reader. Just to have the card dropped into the little basket. This command is executed only when the flag of the configuration byte is abled. Here below there are the commands for managing such structures.

**NOTE:**
When **MDU 01** is configured to manage the motorized Badge Reader, it can manage 2 Kbytes of EEPROM max (instead of 4 Kbytes) and it can not manage the Real Time Clock. These latter options are so installable on **MDU 01** only if these terminals do not manage the Badge Reader.

BADGE READER REQUEST OF THE ACQUIRED STRING

```
Code:  27 76  
Hex code:  1B 4C  
Mnemonic:  ESC L  
```

The answering string is sent via serial line with the sintax already described. If **MDU 01** is set for communicating in Master-Slave mode, the answering string will not contain **CR+LF** chars. This command is available also when **MDU 01** is configured for sending the acquired string to the Badge reader.

READING OF THE BADGE READER CONFIGURATION BYTE

```
Code:  27 33 98  
Hex code:  1B 21 62  
Mnemonic:  ESC ! b  
```

The byte having the Badge reader configuration connected to the **MDU 01** is sent in serial mode. The meaning of this byte is the same one described in the following paragraphs. By this command it is so possible to know in any moment the configuration of **MDU 01**.
WRITING OF THE BADGE READER CONFIGURATION BYTE

**Code:** 27 33 66 byte  
**Hex code:** 1B 21 42 badge  
**Mnemonic:** ESC ! B ASCII(badge)

The byte with the parameters for Badge reader is storaged in the on-board EEPROM and kept there even if power voltage fails. Therefore **MDU 01** is re-configurable for the badge reader management in compliance with the parameters the said byte supplies: this configuration wil be restored each time powered and will maintain this configuration until a new setting byte will be sent. The meaning of the byte is:

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>The Badge reader connected to the <strong>MDU 01</strong> is of the <strong>Insertion</strong> Type (only if the reader is of manual insertion type, because if the reader is of automatic insertion type, this bit is ignored).</td>
</tr>
<tr>
<td></td>
<td>The Badge reader connected to the <strong>MDU 01</strong> is of Sliding Type.</td>
</tr>
<tr>
<td>1</td>
<td>The string memorized on to the magnetic card is acquired when this latter is in the <strong>insertion phase</strong> in the reader (only if the reader is of manual insertion type, because if the reader is of automatic insertion type, this bit is ignored).</td>
</tr>
<tr>
<td></td>
<td>The string memorized on to the magnetic card is acquired when this latter is during the <strong>disinsertion phase</strong> from the reader.</td>
</tr>
<tr>
<td>2</td>
<td>The Badge reader connected to the <strong>MDU 01</strong> reads <strong>Track 1</strong></td>
</tr>
<tr>
<td>1</td>
<td>The Badge reader connected to the <strong>MDU 01</strong> reads <strong>Track 2 or 3</strong></td>
</tr>
<tr>
<td>3</td>
<td>When a string is acquired from the Badge reader, this one is storaged and immediately sent in serial mode. This function is not available in <strong>Master-Slave mode</strong>.</td>
</tr>
<tr>
<td></td>
<td>When a string is acquired from the Badge reader, this one is only storaged and will be sent in serial mode only if the <strong>MDU 01</strong> will receive the specific command.</td>
</tr>
<tr>
<td>4</td>
<td>The magnetic card is inserted into the badge reader, starting from <strong>its beginning</strong> (only if the reader is of Manual Insertion Type, when the reader is of sliding or automatic type, this bit is ignored)</td>
</tr>
<tr>
<td></td>
<td>The magnetic card is inserted into the Badge reader starting from <strong>its end</strong>.</td>
</tr>
<tr>
<td>5</td>
<td>The badge reader is of <strong>Manual</strong> type.</td>
</tr>
<tr>
<td></td>
<td>The badge reader is of <strong>Automatic</strong> type.</td>
</tr>
<tr>
<td>6</td>
<td>Possibility of &quot;EATING&quot; the card <strong>ON</strong> (only if the reader is of automatic type, when the reader is of manual type this bit is ignored).</td>
</tr>
<tr>
<td></td>
<td>Possibility of &quot;EATING&quot; the card <strong>OFF</strong> (only if the reader is of automatic type, when the reader is of manual type this bit is ignored).</td>
</tr>
<tr>
<td>7</td>
<td>The badge reader management is <strong>ON</strong></td>
</tr>
<tr>
<td></td>
<td>The badge reader management is <strong>OFF</strong></td>
</tr>
</tbody>
</table>

When the management of the automatic badge reader is **ON**, the **MDU 01** will execute the inizialization of the device itself by acting the motor in a "Reverse mode" for about 0.5 seconds, just to eject a card may be kept inside.

For example if you wish to configure the **MDU 01** for managing the Insertion Badge Reader which acquires the track 2 having the reading of the card during the disinsertion phase, it will be necessary...
to send the following sequence:

27 33 66 6 or 1B 21 42 06 Hex or ESC ! B ACK

for getting a prompt response from MDU 01. If you send:

27 33 66 14 or 1B 21 42 0E Hex or ESC ! B SO

the string acquired by MDU 01 will be received only if MDU 01 will ask for it by using the specific command.

NOTE:
This command needs a data writing in on-board EEPROM so before executing it be sure that the card is ready for the new writing on that device otherwise the command will be ignored.

ACQUISITION OF THE AUTOMATIC READER STATUS

<table>
<thead>
<tr>
<th>Code</th>
<th>Hex code</th>
<th>Mnemonic</th>
</tr>
</thead>
<tbody>
<tr>
<td>27 67</td>
<td>1B 43</td>
<td>ESC C</td>
</tr>
</tbody>
</table>

A byte containig some status info (card inside, motor ON ect.) is sent back. This byte has the following values:

0 -> No card inside the badge reader and motor is OFF.
1 -> The reader has a card inside and motor is OFF.
2 -> No card inside and motor is ON. This code is sent back, for example, during the inizializing phase of the reader itself after a power-on of the MDU 01 or while the card is "EATEN".
3 -> The reader has a card inside and motor is ON. This code is sent back, for example, during the card reading or during insertion/ejection phases of the same.
255 -> The badge reader is of manual type or the management of the card device is OFF.

EXECUTION OF A NEW READING PROCEDURE

<table>
<thead>
<tr>
<th>Code</th>
<th>Hex code</th>
<th>Mnemonic</th>
</tr>
</thead>
<tbody>
<tr>
<td>27 33 76</td>
<td>1B 21 4C</td>
<td>ESC ! L</td>
</tr>
</tbody>
</table>

A new reading procedure of the card inside the automatic badge reader is executed. If there are other operations on the same device or the MDU 01 is configured for the manual reader, this command is ignored.

NOTE: Each card reading procedure include 3 attempts max, if a mistake in acquiring the same occurs.
CARD EJECTION FROM AUTOMATIC READER

<table>
<thead>
<tr>
<th>Code</th>
<th>Hex code</th>
<th>Mnemonic</th>
</tr>
</thead>
<tbody>
<tr>
<td>27 33 101</td>
<td>1B 21 65</td>
<td>ESC ! e</td>
</tr>
</tbody>
</table>

The ejection of the card inside the automatic badge reader is executed. This command is ignored when: a) no card inside the reader, b) other operations are taking place on the device, c) the MDU 01 is configured for the manual reader.

COMMAND FOR "EATING" THE CARD

<table>
<thead>
<tr>
<th>Code</th>
<th>Hex code</th>
<th>Mnemonic</th>
</tr>
</thead>
<tbody>
<tr>
<td>27 33 84</td>
<td>1B 21 54</td>
<td>ESC ! T</td>
</tr>
</tbody>
</table>

The card inside the badge reader is "eaten" (it goes out from the back of the reader itself just to drop-it into its basket). This operation is executed only if the corrispondent bit in the configuration byte is cleared (bit 6=0). This command is ignored when: a) no card inside the reader, b) other operations are taking place on the device, c) the MDU 01 is configured for the manual reader.
EXTERNAL CARDS

The typical application of MDU 01 are those that require interaction between operator and controlled machine for data, status, measures, information exchanges. The card can be connected to a wide range of programmable devices provided of one serial communication line produced by grifo®, or to many system of other companies.

Hereunder some of these cards are briefly described; ask the detailed information directly to grifo®, or search it on grifo® CD or web site, if required.

**GPC® 553**
General Purpose Controller 80C552
80C552 µP, 22÷33 MHz; 1 RS 232 line (software); 1 RS 232 or RS 422-485 or Current Loop line; 16 TTL I/O lines; 8 A/D 10 bits lines; 3 Timers Counters; 64K EPROM; 64K RAM; 32K RAM and RTC backed; 32K DIL EEPROM; 8K serial EEPROM; 2 PWM lines; 1 Activity LED; Watch dog; 5 readable DIPs; LCD Interface; **ABACO® I/O BUS**.

**GPC® 150**
General Purpose Controller 84C15
Microprocessor Z80 at 16 MHz; implementation completely CMOS; 512K EPROM or FLASH; 512K SRAM; RTC; Back-Up through external Lithium battery; 4M serial FLASH; 1 serial line RS 232 plus 1 RS 232 or RS 422-485 or current loop; 40 I/O TTL; 2 timer/counter; 2 watch dog; dip switch; EEPROM; A/D converter with resolution 12 bit; activity LED.

**GPC® 153**
General Purpose Controller Z80
84C15 µP, 10÷16 MHz; Full CMOS; 1 RS 232 line; 1 RS 232 or RS 422-485 or Current Loop line; 16 TTL I/O lines; 8 A/D 12 bits lines; 2÷4 Timers Counters; 512K EPROM or FLASH; 512K RAM and RTC backed; 8K serial EEPROM; Buzzer; 1 Activity LED; Watch dog; 8 readable DIPs; LCD Interface; **ABACO® I/O BUS**.

**GPC® 184**
General Purpose Controller Z80195
Microprocessor Z80195 at 22 MHz; implementation completely CMOS; 512K EPROM or FLASH; 512K RAM; Back-Up with Lithium battery internal or external; 1 serial line RS 232 + 1 RS 232 or RS 422-485 or current loop + 1 TTL; 18 I/O TTL; 4 timer/counter 8 bits; 2 timer 16 bits; Watch Dog; Real Time Clock; activity LED; EEPROM; interface for **ABACO® I/O BUS**.

**GPC® 154**
“4” Type General Purpose Controller Z80
84C15 µP, 10 MHz; full CMOS; 1 RS 232 line; 1 RS 232 or RS 422-485 line; 16 TTL I/O lines; 512K EPROM or FLASH; 512K RAM and RTC backed; 8K serial EEPROM; 2÷4 timers/counters; Watch dog; 2 readable DIPs; LCD Interface; **ABACO® I/O BUS**; 5Vdc power supply. Size 100x50 mm.

**GPC® 324/D**
“4” Type General Purpose Controller 80C32/320
80C32 or 80C320 µP, 14÷22 MHz; Full CMOS; 1 RS 232 line; 1 RS 232 or RS 422-485 or Current Loop line; 4÷16 TTL I/O lines; 3 Timers Counters; 64K EPROM; 64K RAM; 32K RAM backed; 32K DIL E2; 8K serial EEPROM; Watch dog; 1 readable DIP; LCD Interface; **ABACO® I/O BUS**; 5Vdc Power supply; Size: 100x50 mm.
GPC® 884
General Purpose Controller Am188ES
Microprocessor AMD Am188ES up to 40 MHz; 16 bits; implementation completely CMOS; serie 4 format; 512K EPROM or FLASH; 512K SRAM backed with Lithium battery; RTC; 1 RS 232 serial line + 1 RS 232 or RS 422-485 or current loop; 16 I/O TTL; 3 timer/counter; watch dog; EEPROM; 11 signals A/D converter with 12 bit resolution; interface for ABACO® I/O BUS.

GPC® 114
General Purpose Controller 68HC11
Microprocessor 68HC11A1 at 8 MHz; type 4 format; 32K EPROM; 32K SRAM backed with Lithium battery; 32K EPROM, SRAM, EEPROM; RTC; 1 serial line RS 232, RS 422 or RS 485; 10 TTL I/O lines; 3 timers/counters; watch dog; 8 A/D converter signals with 8 bits resolution; 1 synchronous serial line; extremely low power consumption; interface for ABACO® I/O BUS.

GPC® 150
General Purpose Controller 84C15
Microprocessor Z80 at 16 MHz; implementation completely CMOS; 512K EPROM or FLASH; 512K SRAM; RTC; Back-Up through external Lithium battery; 4M serial FLASH; 1 serial line RS 232 plus 1 RS 232 or RS 422-485 or current loop; 40 I/O TTL; 2 timer/counter; 2 watch dog; dip switch; EEPROM; A/D converter with resolution 12 bit; activity LED.

GPC® 550
General Purpose Controller 80C552
Microprocessor 80C552 at 22 MHz. 32K EPROM; 32 K RAM; 32 K EEPROM or SRAM; RTC; serial EEPROM; serial lines 1 RS 232 + 1 RS 232 or RS 422-485 or current loop; 40 I/O TTL; 2 lines of PWM; 16 bits timer/counter; watch dog; dip switch; 8 lines 10 bit A/D converter; interface for BUS ABACO®; CAN line galvanically isolated. Unique power supply +5 Vdc; EUROCARD format.

MSI 01
Multi Serial Interface 1 line
Interface card for TTL serial line that is buffered in RS 232, RS 422, RS 485, or current loop line. The TTL line is on a mini screw connector and the buffered one is on standard plug connector.

IBC 01
Interface Block Communication
Conversion card for serial communication, 2 RS 232 lines; 1 RS 422 or RS 485 line; 1 optical fibre line; selectable DTE/DCE interface; quick connection for DIN 46277-1 and 3 rails.

GPC® 188F
General Purpose Controller 80C188
80C188 µP 20MHz; 1 RS 232 line; 1 RS 232, RS 422-485 or Current Loop line; 24 TTL I/O lines; 1M EPROM or 512K FLASH; 1M SRAM Lithium battery backed; 8K serial EEPROM; RTC; watch dog; 8 dip switch; 3 timer counter; 8 13 bit A/D lines; Power failure; activity LEDs.

GPC® 15A
General Purpose Controller 84C15
Full CMOS card, 10+20 MHz 84C15 CPU; 512K EPROM or FLASH EPROM; 128K RAM; 2K or 8K backed RAM+RTC; 8K serial EEPROM; 1 RS 232 serial line; 1 RS 232, RS 422, RS 485 or current loop line; 40 TTL I/O lines; 2 counters timers; 2 watch dogs; 2 dip switches, buzzer.
Figure 47: Available Connections Diagram
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 MDU 01.

Manual HEWLETT PACKARD: Optoelectronics Designer’s Catalog
Manual NATIONAL SEMICONDUCTOR: Linear Databook - Volume 1
Manual MAXIM: New Releases Data Book - Volume IV
Manual MAXIM: New Releases Data Book - Volume V
Manual PHILIPS: 80C51 - Based 8-Bit Microcontrollers
Manual PHILIPS: IC12 - FC bus
Manual PHILIPS: Application notes and development tools for 80C51 microcontrollers
Manual TEXAS INSTRUMENTS: The TTL Data Book - SN54/74 Families
Manual TEXAS INSTRUMENTS: RS-422 and RS-485 Interface Circuits
Manual SGS-THOMSON: Industrial and Computer peripheral ICs
Manuale SGS-THOMSON: Power supply application manual
Manual XICOR: Data Book

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.
APPENDIX A: COMMAND CODES SUMMARY TABLES

The tables of this appendix list a summary of all the command sequences recognized by MDU 01.

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home</td>
<td>01</td>
<td>01</td>
<td>SOH</td>
</tr>
<tr>
<td>Cursor left</td>
<td>21</td>
<td>15</td>
<td>NACK</td>
</tr>
<tr>
<td>Cursor right</td>
<td>06</td>
<td>06</td>
<td>ACK</td>
</tr>
<tr>
<td>Cursor down</td>
<td>10</td>
<td>0A</td>
<td>LF</td>
</tr>
<tr>
<td>Cursor up</td>
<td>26</td>
<td>1A</td>
<td>SUB</td>
</tr>
<tr>
<td>Carriage return</td>
<td>13</td>
<td>0D</td>
<td>CR</td>
</tr>
<tr>
<td>Carriage return+line feed</td>
<td>29</td>
<td>1D</td>
<td>GS</td>
</tr>
<tr>
<td>Absolute cursor position</td>
<td>27 89</td>
<td>1B 59</td>
<td>ESC Y ASCII(r) ASCII(c)</td>
</tr>
<tr>
<td>Back space</td>
<td>08</td>
<td>08</td>
<td>BS</td>
</tr>
<tr>
<td>Clear page</td>
<td>12</td>
<td>0C</td>
<td>FF</td>
</tr>
<tr>
<td>Clear line</td>
<td>25</td>
<td>19</td>
<td>EM</td>
</tr>
<tr>
<td>Clear end of line</td>
<td>27 75</td>
<td>1B 4B</td>
<td>ESC K</td>
</tr>
<tr>
<td>Clear end of page</td>
<td>27 107</td>
<td>1B 6B</td>
<td>ESC k</td>
</tr>
<tr>
<td>Cursor off</td>
<td>27 80</td>
<td>1B 50</td>
<td>ESC P</td>
</tr>
<tr>
<td>Steady cursor on</td>
<td>27 79</td>
<td>1B 4F</td>
<td>ESC O</td>
</tr>
<tr>
<td>Blinkling cursor on</td>
<td>27 77</td>
<td>1B 4D</td>
<td>ESC M</td>
</tr>
<tr>
<td>Blinkling block cursor on</td>
<td>27 81</td>
<td>1B 51</td>
<td>ESC Q</td>
</tr>
</tbody>
</table>

**FIGURE A1: COMMAND CODES SUMMARY TABLE (1 OF 4)**
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request of EEPROM writing</td>
<td>27 51</td>
<td>1B 33</td>
<td>ESC 3</td>
</tr>
<tr>
<td>Writing of presence byte</td>
<td>27 33 78 byte</td>
<td>1B 21 4E byte</td>
<td>ESC ! N ASCII(byte)</td>
</tr>
<tr>
<td>Reading of presence byte</td>
<td>27 33 110</td>
<td>1B 21 6E</td>
<td>ESC ! n</td>
</tr>
<tr>
<td>Key code reconfiguration</td>
<td>27 55 key num. cod.</td>
<td>1B 37 key num. cod.</td>
<td>ESC 7 ASCII(key num.) ASCII(cod.)</td>
</tr>
<tr>
<td>Keyclick on without memorization</td>
<td>27 53</td>
<td>1B 35</td>
<td>ESC 5</td>
</tr>
<tr>
<td>Keyclick off without memorization</td>
<td>27 54</td>
<td>1B 36</td>
<td>ESC 6</td>
</tr>
<tr>
<td>Keyclick on with memorization</td>
<td>27 33 53</td>
<td>1B 21 35</td>
<td>ESC ! 5</td>
</tr>
<tr>
<td>Keyclick off with memorization</td>
<td>27 33 54</td>
<td>1B 21 36</td>
<td>ESC ! 6</td>
</tr>
<tr>
<td>Definition of user character</td>
<td>27 66 nchar Pat0...Pat7</td>
<td>1B 42 nchar Pat0...Pat7</td>
<td>ESC B ASCII(nchar) ASCII(Pat0)...ASCII(Pat7)</td>
</tr>
<tr>
<td>Definition and memorization of user character</td>
<td>27 33 66 nchar Pat0...Pat7</td>
<td>1B 21 42 nchar Pat0...Pat7</td>
<td>ESC ! B ASCII(nchar) ASCII(Pat0)...ASCII(Pat7)</td>
</tr>
<tr>
<td>Reading of max message number</td>
<td>27 110</td>
<td>1B 6E</td>
<td>ESC n</td>
</tr>
<tr>
<td>Message storage</td>
<td>27 33 67 mess. num. car.0...car.19</td>
<td>1B 21 43 mess. num. car.0...car.13</td>
<td>ESC ! C ASCII(mess. num.) ASCII(car.0)...ASCII(car.19)</td>
</tr>
<tr>
<td>Message reading</td>
<td>27 33 69 mess. num.</td>
<td>1B 21 45 mess. num.</td>
<td>ESC ! E ASCII(mess. num.)</td>
</tr>
<tr>
<td>Visualization of n messaggi</td>
<td>27 33 68 mess. num. n</td>
<td>1B 21 44 mess. num. n</td>
<td>ESC ! D ASCII(mess. num.) ASCII(n)</td>
</tr>
<tr>
<td>Scrolling message visualization</td>
<td>27 33 83 mess. num. n.char</td>
<td>1B 21 53 mess. num. n.char</td>
<td>ESC ! S ASCII(mess. num.) ASCII(n.char)</td>
</tr>
</tbody>
</table>

**Figure A2: Command codes summary table (2 of 4)**
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>BEEP</td>
<td>07</td>
<td>07</td>
<td>BEL</td>
</tr>
<tr>
<td>Reading of version number</td>
<td>27 86</td>
<td>1B 56</td>
<td>ESC V</td>
</tr>
<tr>
<td>LED Activation</td>
<td>27 50</td>
<td>1B 32 n.LED Attr.</td>
<td>ESC 2 ASCII(n.LED) ASCII(Attr.)</td>
</tr>
<tr>
<td>LEDs Mask Activation</td>
<td>27 52</td>
<td>1B 34 byte1 byte2 byte3</td>
<td>ESC 4 ASCII(byte1) ASCII(byte2) ASCII(byte3)</td>
</tr>
<tr>
<td>Relay Activation</td>
<td>27 56</td>
<td>1B 38</td>
<td>ESC 8</td>
</tr>
<tr>
<td>Relay Deactivation</td>
<td>27 57</td>
<td>1B 39</td>
<td>ESC 9</td>
</tr>
<tr>
<td>Writing of the Badge Reader configuration byte</td>
<td>27 33</td>
<td>1B 21 42 byte1</td>
<td>ESC ! B ASCII(byte)</td>
</tr>
<tr>
<td>Reading of the Badge Reader configuration byte</td>
<td>27 33</td>
<td>1B 21 62</td>
<td>ESC ! b</td>
</tr>
<tr>
<td>Badge Reader request of the acquired string</td>
<td>27 76</td>
<td>1B 4C</td>
<td>ESC L</td>
</tr>
<tr>
<td>Acquisition of the automatic reader status</td>
<td>27 67</td>
<td>1B 43</td>
<td>ESC C</td>
</tr>
<tr>
<td>Execution of a new reading procedure</td>
<td>27 33</td>
<td>1B 21 4C</td>
<td>ESC ! L</td>
</tr>
<tr>
<td>Card ejection from automatic reader</td>
<td>27 33</td>
<td>1B 21 65</td>
<td>ESC ! e</td>
</tr>
<tr>
<td>Command for &quot;EATING&quot; the card</td>
<td>27 33</td>
<td>1B 21 54</td>
<td>ESC ! T</td>
</tr>
</tbody>
</table>

**Figure A3: Command codes summary table (3 of 4)**
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock SET UP</td>
<td>27 33 70 “string”</td>
<td>1B 21 46 “string”</td>
<td>ESC ! F ASCII(“string”)</td>
</tr>
<tr>
<td>Clock reading</td>
<td>27 33 102</td>
<td>1B 21 66</td>
<td>ESC ! f</td>
</tr>
<tr>
<td>Time visualization on the display</td>
<td>27 33 116 r c frm</td>
<td>r c frm</td>
<td>ESC ! t ASCII(r) ASCII(c) ASCII(frm)</td>
</tr>
<tr>
<td>Date visualization on the display</td>
<td>27 33 100 r c frm</td>
<td>r c frm</td>
<td>ESC ! d ASCII(r) ASCII(c) ASCII(frm)</td>
</tr>
<tr>
<td>Writing of a byte of the Real Time Clock RAM</td>
<td>27 33 71 addr byte</td>
<td>1B 21 47 addr byte</td>
<td>ESC ! G ASCII(addr) ASCII(byte)</td>
</tr>
<tr>
<td>Reading of a byte of the Real Time Clock RAM</td>
<td>27 33 103 addr</td>
<td>1B 21 67 addr</td>
<td>ESC ! g ASCII(addr)</td>
</tr>
</tbody>
</table>

**Figure A4: Command codes summary table (4 of 4)**
APPENDIX B: DISPLAY CHARACTERS

The following tables shows the characters sets displayed on MDU 01 for all the possible received characters, according with ordered display and model. Even the not ASCII characters (or special characters) change when the display type changes and if the user requires a character set different from those described in the following figures, he can take a direct contact with grifo®.

**Figure B1: MDU 01-C4BIG, C14, C24, C44 Characters Table**

![Character Table Image]
**Figure B3: MDU 01-F14 Characters Table**
**Figure B4: MDU 01-F2, F4, F24 Characters Table**

<table>
<thead>
<tr>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0001</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0010</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0011</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0100</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0101</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0110</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0111</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1001</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1010</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1011</td>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1100</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1101</td>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1110</td>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1111</td>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Note: The table contains character representations for different bit configurations.*
FIGURE B5: MDU 01-F44 CHARACTERS TABLE
APPENDIX C: MOUNTING OUTLINE DIMENSIONS

TERMINAL DIMENSIONS

Here are dimensions of terminal MDU 01 external metallic container and frontal plastic frame. Dimensions are in mm, scale is 1:1.

Dimensions of container only, area occupied can be slightly greater considering also mounting clamps and screws, up to a maximum of 150 x 120 x 40 mm (W x H x D).

**Figure C1: Dimensions MDU 01**
A plastic container with direct clamps for DIN 46277-1 and 3 is enough to place the card on omega rails. This option can be ordered with code: **BLOCK.100.148**. This mounting option allows also to match MDU 01 with other cards 100 mm long like GPC® CPU cards, display interfaces, field interfaces, etc. simply using a longer container to obtain a unique element. Above mentioned plastic support is Weidmuller RS/100 (code 414487), and can be ordered to grifo® with option code **EXT-WMlll**, where lll indicates the desired length in mm.
APPENDIX D: ALPHABETICAL INDEX

SYMBOLS

.5VDC  37
.5VOUT  37
.ALIM  37

A
A (+)  25

B
BADGE  9, 26
BUFFER  42

C

CHARACTERS  43

COMMANDS FOR BADGE READER MANAGEMENT

ACQUISITION OF THE AUTOMATIC READER STATUS  69
BADGE READER REQUEST OF THE ACQUIRED STRING  67
CARD EJECTION FROM AUTOMATIC READER  70
EXECUTION OF A NEW READING PROCEDURE  69
READING A BADGE FROM A MOTORIZED READER  67
READING OF THE BADGE READER CONFIGURATION BYTE  67
STANDARD RECOGNIZED  65
SYNTAX OF THE STRING READ FROM THE BADGE  66
WRITING OF THE BADGE READER CONFIGURATION BYTE  68

COMMANDS FOR CHARACTERS ERASURE

BACKSPACE  50
CLEAR END OF LINE  50
CLEAR END OF PAGE  50
CLEAR LINE  50
CLEAR PAGE  50

COMMANDS FOR CURSOR ATTRIBUTES MANAGEMENT

BLINKING "BLOCK" CURSOR  51
BLINKING BLOCK CURSOR ON  51
CURSOR OFF  51
STEADY STATIC CURSOR ON  51

COMMANDS FOR CURSOR POSITION

ABSOLUTE CURSOR PLACEMENT  49
CARRIAGE RETURN  49
CARRIAGE RETURN + LINE FEED  49
CURSOR DOWN  48
CURSOR LEFT  48
CURSOR RIGHT  48
CURSOR UP  48
HOME  48
COMMANDS FOR EEPROM
- READING OF PRESENCE BYTE  52
- REQUEST FOR EEPROM WRITING POSSIBILITY  52
- WRITING OF PRESENCE BYTE  52

COMMANDS FOR GENERAL FUNCTIONS
- BEEP  53
- READING OF VERSION NUMBER  53
- RELAY ACTIVATION  53
- RELAY DEACTIVATION  53

COMMANDS FOR KEYBOARD MANAGEMENT
- KEY RECONFIGURATION  56
- KEYCLICK OFF WITH MEMORIZATION  57
- KEYCLICK OFF WITHOUT MEMORIZATION  56
- KEYCLICK ON WITH MEMORIZATION  57
- KEYCLICK ON WITHOUT MEMORIZATION  56

COMMANDS FOR LEDS MANAGEMENT
- LED ACTIVATION  54
- LEDS MASK ACTIVATION  54

COMMANDS FOR MANAGING THE ON-BOARD REAL TIME CLOCK
- CLOCK READING  62
- CLOCK SET-UP  61
- DATE VISUALIZATION ON THE DISPLAY  63
- READING OF A BYTE OF THE REAL TIME CLOCK SRAM  61
- TIME VISUALIZATION ON THE DISPLAY  62
- WRITING OF A BYTE OF THE REAL TIME CLOCK SRAM  61

COMMANDS FOR MESSAGE MANAGEMENT
- MESSAGE READING  59
- MESSAGE STORAGE  58
- MESSAGE VISUALIZATION  59
- READING OF THE LAST STORAGED MESSAGE NUMBER  58
- SCROLLING MESSAGES VISUALIZATION  60

COMMUNICATION PROTOCOL  9

CONNECTIONS  12
- CN1  28, 32
- CN2  12
- CN3  14
- CN4  20
- CN5  25
- CN6  22, 32
- CN7  23, 32
- CN8  24, 32
- CN9  26, 32

CONTRAST REGULATION TRIMMER  32
- CPU  9
- CR1  32

CURRENT AVAILABLE ON +5 VDC  10
- CURRENT LOOP  14, 32, 38, 43
- CURRENT LOOP NETWORK  18
D
DISPLAY 4

E
EXPANDED Firmware 41
EXTERNAL CARDS 71

F
FIRMWARE VERSION 1

I
I/O CONNECTION 32

J
JUMPERS 34
  2 PINS JUMPERS 34
  3 PINS JUMPERS 36
  4 PINS JUMPERS 36
  5 PINS JUMPERS 36

K
K (-) 25
KEY CODES 47
KEYS 28

L
LCD 4, 11, 22, 33
LCD 40X4 23
LCD DISPLAYS LED BACKLIGHTING 25
LEDS 28
LENGTH OF KEYS CONNECTION 10
LOCAL SET UP 41

M
MASTER SLAVE 43
MASTER SLAVE COMMUNICATION EXAMPLE 45
MATRIX KEYBOARD 28, 46
MAXIMUM VOLTAGE ON RELAY 10
MOUNTING 10
MOV 20
N
NETWORK 43
NORMAL FIRMWARE 42

O
OUTPUT VOLTAGE 10

P
POINT-TO-POINT 43
POWER CONSUMPTION 10
POWER SUPPLY 12, 37
POWER VOLTAGE 10

R
RECEIVE BUFFER 42
RELATIVE HUMIDITY 10
RELAY 20
RESOURCES 9
RS 232 14, 32, 38, 43
RS 422 14, 32, 38, 43
RS 485 14, 32, 40, 43
RS 485 NETWORK 16

S
SET UP 41
   BAUD RATE 41
   COMMUNICATION 41
   KEYCLICK 41
   NAME 41
   STOP BIT 41
SIZE 10

T
TEMPERATURE RANGE 10
TTL 32

V
VFD 4, 11, 22, 24, 33
VFD 40X4 24

W
WEIGHT 10