QTP 12H

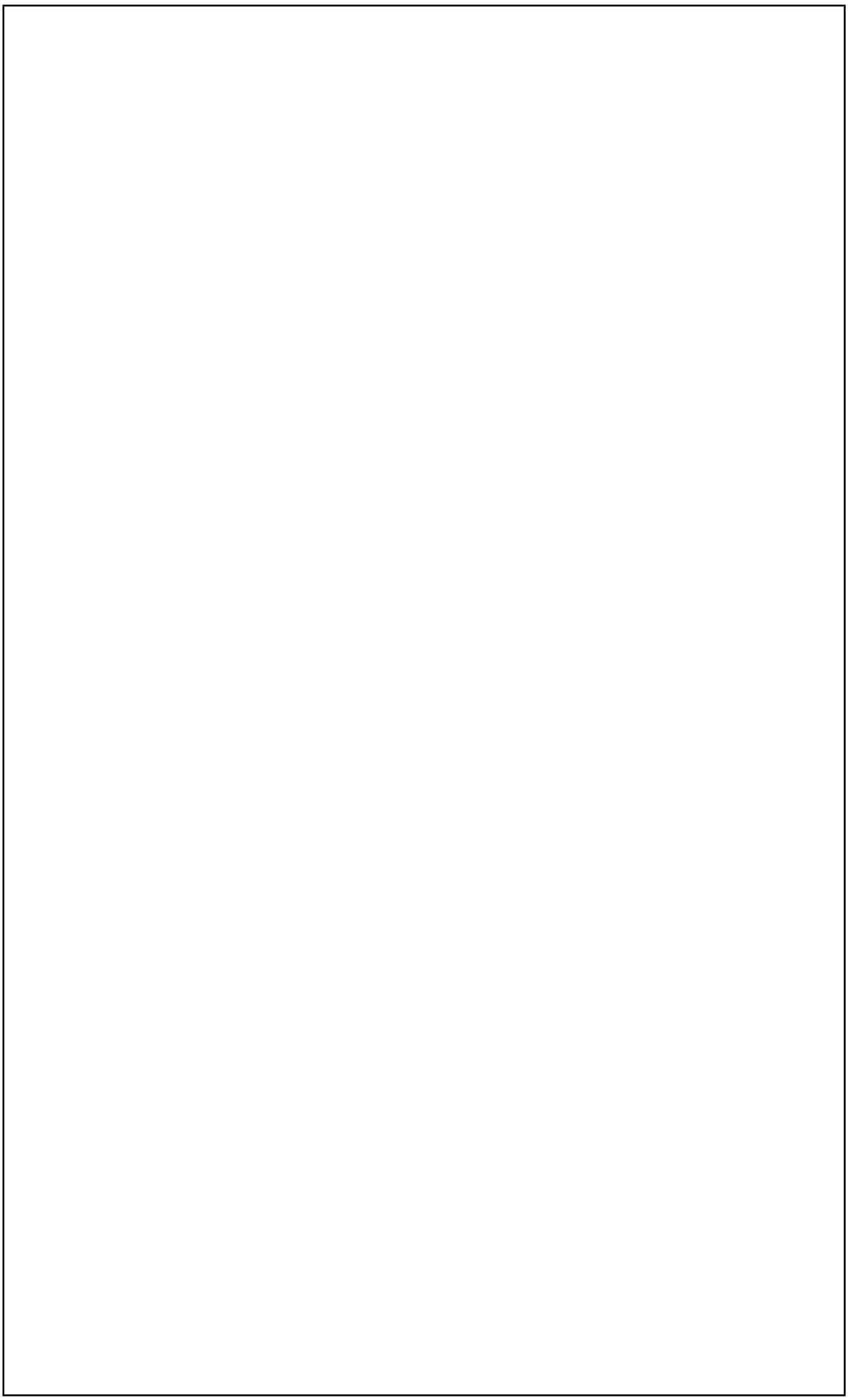
Quick Terminal Panel - Housing withouth keys

USER MANUAL

grifo®
ITALIAN TECHNOLOGY

Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY
E-mail: grifo@grifo.it
http://www.grifo.it http://www.grifo.com
Tel. +39 051 892.052 (a.r.) FAX: +39 051 893.661

QTP 12H Rel. 5.00 *Edition 06 April 2005*



QTP 12H

Quick Terminal Panel - Housing withouth keys

USER MANUAL

Visualization panel provided of very low price. Size: front **72x144 mm**; depth of front **9 mm**; depth of back **28 mm**. Alluminium container with frontal frame in plastic, provided with mounting clamps. Front panel with mask and keyboard in polyester scratch proof. Surface or flush panel mounting. Panel front protection level: **IP-54**. Capable to drive **3** display models fluorescent and **LCD** Alphanumeric **20x2** characters and fluorescent Graphic **140x16** pixels. Maximum size of visible area: **82x18 mm**. Personalization of device by inserting a visible label. **Buzzer** for **BELL** signalations and software driven acoustic signalations. **EEPROM for** settings, messages, patterns, etc. Memorization and representation of up to **97** messages in **EEPROM**, also with scrolling attribute. **CAN** communication line with its own line driver. Serial line in **RS 232** or **RS 422, RS 485, Current Loop**. Network connection by master slave protocol. Local setting to configure operating modes. **8** characters with user defined pattern. Up to **256** different characters built-in the display and so printable. Power supply **DC** or **AC** from **5 Vdc** up to **24 Vac**. Required power ranges from **1.2 W** to **2.3 W**, according to used configuration. On board logic protection through **TransZorb™**.
It is possible to require custom panels and programs.

grifo®
ITALIAN TECHNOLOGY
Via dell' Artigiano, 8/6
40016 San Giorgio di Piano
(Bologna) ITALY
E-mail: grifo@grifo.it
http://www.grifo.it http://www.grifo.com
Tel. +39 051 892.052 (a.r.) FAX: +39 051 893.661

QTP 12H Rel. 5.00 *Edition 06 April 2005*

DOCUMENTATION COPYRIGHT BY grifo®, ALL RIGHTS RESERVED

IMPORTANT

Although all the information contained herein have been carefully verified, grifo® assumes no responsability for errors that might appear in this document, or for damage to things or persons resulting from technical errors, omission and improper use of this manual and of the related software and hardware.
grifo® reserves the right to change the contents and form of this document, as well as the features and specification of its products at any time, without prior notice, to obtain always the best product.
For specific informations on the components mounted on the card, please refer to the Data Book of the builder or second sources.

SYMBOLS DESCRIPTION

In the manual could appear the following symbols:


Attention: Generic danger


Attention: High voltage


Attention: ESD sensitive device

Trade Marks

abaco®, **GPC®**, grifo® : are trade marks of grifo®.
Other Product and Company names listed, are trade marks of their respective companies.

GENERAL INDEX

FIGURES INDEX

INTRODUCTION

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

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

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

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

The reported data are destined- IN EXCLUSIVE WAY- to specialized users, that can interact with the devices in safety conditions for the persons, for the machine and for the enviroment, 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 rispect 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.

HARDWARE AND FIRMWARE VERSION

This handbook make reference to printed circuit version **110703** and to firmware version **1.3** and following ones. The validity of the information contained in this manual is subordinated to the version numbers on the used panel, and so the user must always verify the correct correspondence beetween the notations. The version numbers are reported in several places on the electronic part of the product, and following figure shows the most accessible ones. Obviously if the electronic must be checked, then it must be extracted from the metallic container: a simple pressure on **QTP 12H** connectors, or on the printed circuit reachable from rear container window, is sufficient. When on the front panel there are two black screws, they must be previously unscrewed.

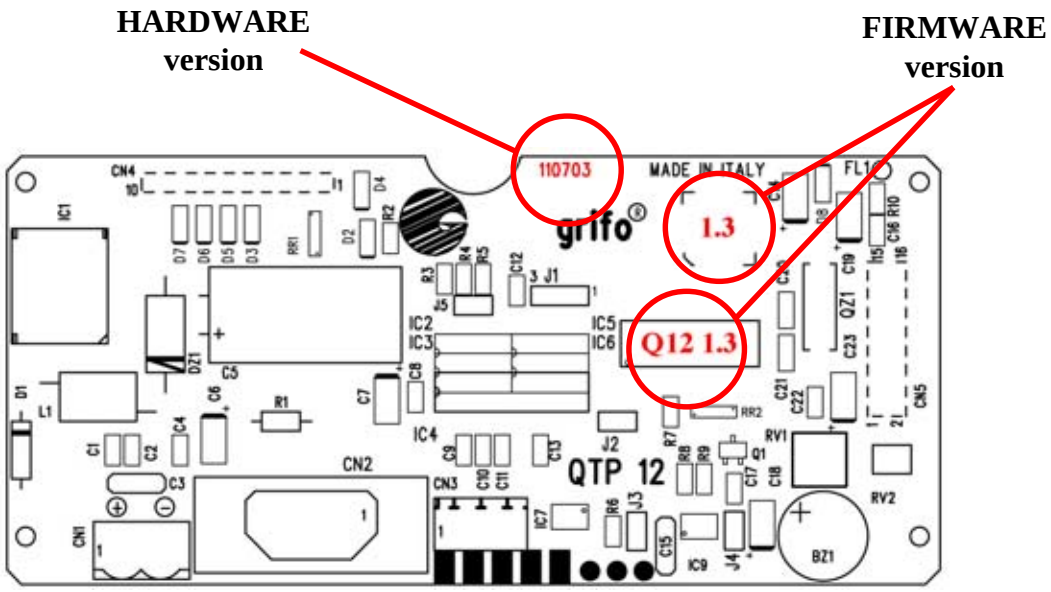


FIGURE 1: LOCATION OF HARDWARE AND FIRMWARE VERSION

The firmware version number can be also directly required to the terminal by using a dedicated command.
Normally the **QTP 12H** is always supplied with the latest firmware version that is available but, for specific requirements, the user can receive also a different version; he must carefully specify this particular condition in the order phase.

GENERAL INFORMATION

QTP 12H is a complete low cost operator panel with small overall dimension, specifically designed for industrial use and for direct mounting on automatic machinery. It is a video terminal suitable to be the direct interface between operator and machinery in any of the control and visualization operations which could be necessary in many civil and/or industrial applications.
QTP 12H is available with **alphanumeric Fluorescent** or LEDs backlight **LCD** displays, with 20 characters for 2 lines or with **Fluorescent** graphic display 140x16 pixels.
QTP 12H is directly connected with the front panel, that is provided of a suitable pocket to inset a customization label.
A pratical and resistent standard DIN 72x144 container allows to install the terminal in surface or flush panel mounting, with frontal protected and breaks in the rear to access the several connectors. Clamps provided with **QTP 12H** allow to mount/unmount it by a simple rectangular hole in the support rack. Also a table, with no further element, can be used.
QTP 12H is the best choice whenever the user needs to show information, in graphic format too, with a very low effort in management software development.
QTP 12H gives the possibility to store in the on board serial **EEPROM** up to 97 messages. These messages can be shown on the display, also in sliding mode, simply sending to the serial port a proper sequence of commands. With this feature the master program space and its execution time are optimized or even erased, in fact messages must not be sent to the panel every time, they are already stored inside EEPROM of the **QTP 12H**. Furthermore it is possible to get messages back through the communication line and read them again. So **QTP 12H** can be used as little mass memory where the user can save and read set-up informations, passwords, identification codes, etc. The horizontal scrolling attribute for the saved messages, let the user displays more information on less space: on the first row of the display up to 200 characters can be shown in a self managed sliding modality.
QTP 12H can be connected to most of the systems available on the market through a serial line that can be buffered with the most common electric protocols; this allows also to make low cost networks that can be made of up to 256 different units. Furthermore, interconnection is warranted by an optional CAN serial line, that increases the utilization ranges and improves overall network performances.
The **QTP 12H** is able to execute an entire range of display commands, either for alphanumeric and graphic visualization, including: clear the entire screen or part of it, cursor position and movement, buzzer activation, lines draw, graphic area setting, fonts selection, characters definition, messages managemebt, etc., with command codes compatibile to **ADDS Viewpoint** standard.
Main features of **QTP 12H**, including the available otions, are as follows:

- Dimension: front size **72x144 mm**; fore depth of **9 mm**; back depth of **28 mm**.
- Remarkably **low price**.
- Alluminium container with frontal frame in plastic, provided with mounting clamps.
- Front panel with mask **in polyester**, scratch proof.
- **Surface** or flush panel **mounting**.
- Protection level of panel front: **IP-54**.
- Capable to drive **3 display models:**
  QTP 12H-C2: alphanumeric **LCD** backlight, **2** rows by **20** characters
  QTP 12H-F2: alphanumeric **Fluorescent**, **2** rows by **20** characters
  QTP 12H-GF2: graphic **Fluorescent**, **140** by **16** pixels
- Maximum size of visible area: **82x18 mm**.
- Obtained by the union of two other **grifo®** subproducts: the electronic and the rear carter of **QTP 12** and the front panel of **QTP 72144**.

QTP 12H-GF2

QTP 12H-GF2

QTP 12H-C2

QTP 12H-F2

FIGURE 2: PHOTO OF THE AVAILABLE MODELS

- Personalization of device by inserting a label.
- **Buzzer** for BELL signalations and software driven acoustic signalations.
- **EEPROM** for settings, messages, patterns, etc.
- Memorization and representation of up to **97 messages** in EEPROM, even with **scrolling** attribute.
- **CAN** communication **line** with its own line driver.
- Serial line in **RS 232** or **RS 422**, **RS 485**, **current loop**.
- Network connection by master slave protocol.
- Local setting to configure operating modes.
- **8** characters with **user defined pattern**.
- Up to 256 different characters built in the display and so visualizable.
- **Power supply** DC or AC from **5 Vdc** up to **24 Va**c.
- Required power according to configuration used ranges from **1.2 W** and **2.3 W**.
- On board logic protection through **TransZorb™**.
- Customized front panel and program packages (please contact **grifo®**).
- For specific requirements about consumption and price, the LCD diplay can be not backligt (please contact **grifo®**).

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

DISPLAY

QTP 12H is available with three different displays: **graphic fluorescent** with **140x16** pixels, **alphanumeric fluorescent** with **20x2** characters and alphanumeric **LCD** with **20x2** characters backlighted by LED lamp.
LEDs backlighting of LCD model ensures a good visibility even when the environmental lighting changes and if it necessary the user can modify the contrast regulation by acting on a specific trimmer.
Another important features of **QTP 12H** displays is their wide viewing angle that allows a good visibility from each frontal position. Further information on each display are reported in TECHICAL FEATURES chapter.
As described in the chapter dedicated to recognized commands, **QTP 12H** with graphic display (**QTP 12H-GF2**) can execute all kinds of commands (graphic and alphanumeric), while **QTP 12H** with alphanumeric display (**QTP 12H-C2** and **QTP 12H-F2**), of course, cannot execute graphic commands.
The user must choose the right display (and so the right **QTP 12H** model) that is sufficient for the information to visualize and for his visibility requirements.
For specific requirements on current consumption, visibility and price, the card can be provided even with LCD display not backlighted: for detailed information about these options and their availability, please contact directly **grifo®** offices.

SERIAL LINE

The communication with the other units is performed through an asynchronous serial line, that in default configuration, is electrically configured as **RS 232** but using a proper indication in the order, it can be configured in:

RS 422	->	**.RS422** option
RS 485	->	**.RS485** option
Current Loop	->	**.CLOOP** option

The physical protocol of the serial line is completely configurable through a dedicated setup modality that let the user select the values listed in TECHNICAL SPECIFICATIONS chapter, by the simple use of four external keys.
Finally the logic protocol can be point to point or master slave type, using the nineth bit technique; this latter, when used in conjunction with one of the options above described, allows the connection of many **QTPs** on a network and to communicate with terminals of the same or different type, easily and efficiently.

CAN INTERFACE

QTP 12H can have, as option, a complete CAN interface that supports the **BasicCAN** and **PeliCAN 2.0B** standards protocols. With this feature the user can afford and solve many problems as: high speed data trasfer, long distance communication, autonomous errors management, multimaster and multislave networks support, etc.
The code used to order this option is: **.CAN**

ON BOARD POWER SUPPLY

One of the most important peculiarity of **QTP 12H** 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.
As alternative, **QTP 12H** without power supply can be ordered (by using the codes **.5Vdc** or **.ALIM**), in this case a +5 Vdc stabilized power supply must be provided by an external source.
For detailed information about power supply section, please refer to ELECTRIC FEATURES and POWER SUPPLY paragraphs.

BUZZER

QTP 12H has a circuitry that generates a steady sound, based on a capacitive buzzer.
By software, through some specific comands, this circuitry can be enabled, disabled or intermittent, it can generate a simple beep or it can signalize possible malfunctions.
When, after a power on, the card generates a fixed or intermittent sound and it doesn't work correctly, there is a wrong condition that must be resolved: please contact grifo® technicians.

EEPROM

QTP 12H has on board EEPROM (size 2 KBytes) for storing set up, communication protocol, identification name, user characters patterns, messagges, and so on.
Many of the stored data have vital importance so a serial EEPROM has been choosen 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. **QTP 12H** also manages scrolling messages, to show on an unique line more text than it could be visible on a single row, without scrolling.
For detailed information about messages please read COMMANDS FOR MESSAGES MANAGEMENT paragraph.

FIGURE 3: QTP 12H COMPLETE PHOTO

TECHNICAL FEATURES

GENERAL FEATURES

Resources:
IP54 frontal
Slot pocket for personalization label
Buzzer for beep or acoustic feedback
Full duplex RS 232 serial line, it can be buffered in RS 422, RS 485 or current loop (option)
EEPROM size 2 KBytes for configuration, parameters, etc.
CAN interface (option)
Alphanumeric or graphic display in three different models
Trimmer to set LCD display contrast
Switching power supply

Displays:
alphanumeric LCD 20x2 chars, LED backligthing
alphanumeric fluorescent 20x2 chars
graphic fluorescent 140x16 pixels

CPU:
89C5115 or 89C51CC02 with crystal 14.7456 MHz
Default: *89C5115*

Power on time:
150 msec

EEPROM max write time:
8 msec

Timing precision:
2,5 msec

Buzzer intermittent time:
500 msec

Messages shift time:
500 msec

Messages number:
20 or 97
Default: *20*

Receive buffer size:
30 characters

Max units on network:
256

Com. physic protocol:
Baud rate: 1200, 2400, 4800, 9600, 19200, 38400
Stop bit: 1 or 2
Parity: none
Bits x chr: 8, 9
Default: *19200 Baud, 1 Stop, No parity, 8 Bits*

Com. logic protocol:
Selectable between normal and master slave
Default: *normal*

PHYSICAL FEATURES

Size: DIN 72x144: 144 x 72 x 37 mm (W x H x D)
156 x 72 x 80 mm (W x H x D) with clamps
See outline dimension in APPENDIX C

Size of breaking for mount: 138 (min) x 66 (min) x 10 (max) mm (W x H x D)
See outline dimension in APPENDIX C

Characters size: LCD 20x2: 5 x 7 dots, 3.20 x 4.85 mm (W x H)
Fluorescent 20x2: 5 x 7 dots, 2.40 x 4.70 mm (W x H)
Fluorescent 140x16: from 3 x 5 dots, 1.50 x 3.62 mm (W x H)
to 10 x 14 dots, 5.00 x 10.15 mm (W x H)

Weight: 290 g max.

Mounting: Surface or front panel mounting, through provided clamps
At sight on a bearing surface

Temperature range: From 0 to 50 °C

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

Connectors: CN1: quick release screw terminal connector, 2 pins, pitch 5
CN2: D type connector 9 pins, female
CN3: quick release screw terminal connector, 3 pins, pitch 3.5
CN4: 90° strip, 10 pins, male, 2.54 pitch

ELECTRIC FEATURES

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

Power consumption: see next table (*)

DISPLAY Model	*Consumption +5 Vdc*	*Consumption 10÷40 Vdc 8÷24 Vac*
LCD 20x2 alphanumeric backligt: QTP 12H-C2	220 mA	1.5 W
Fluorescent alphanumeric 20x2: QTP 12H-F2	205 mA	1.4 W
Fluorescent graphic 140x16: QTP 12H-GF2	325 mA	2.2 W

FIGURE 4: CONSUMPTIONS TABLE

This table lists the **QTP 12H** power consumption referred to the different display types that can be ordered; for the wide range power supply are described the required power, in place of the current, already corrected with efficiency of the on board power supply section.

RS 232 extravoltage protection: ±15 KV

RS 422-485 line impedance: 60 Ω

RS 422-485 termination:

line termination resistor:	120 Ω
pull-up resistor on positive:	3.3 KΩ
pull-down resistor on negative:	3.3 KΩ

CAN line impedance: 60 Ω

CAN termination circuit: 120 Ω ressitor, disconnectable

(*) The data are referenced to 20 C° environmental work temperature (for further information please refer to chapter POWER SUPPLY).

To reduce consumptions of **QTP 12H** with LCD display it is possible to order particular models without backlighting: for further information please contact directly **grifo®**.

FIGURE 5: REAR VIEW PHOTO

INSTALLATION

In this chapter there are the information for a right installation and correct use of the terminal **QTP 12H**. In detail there are the locations and functions of each connector, of the user settable jumpers, of the trimmer and any other information concerning hardware configuration.

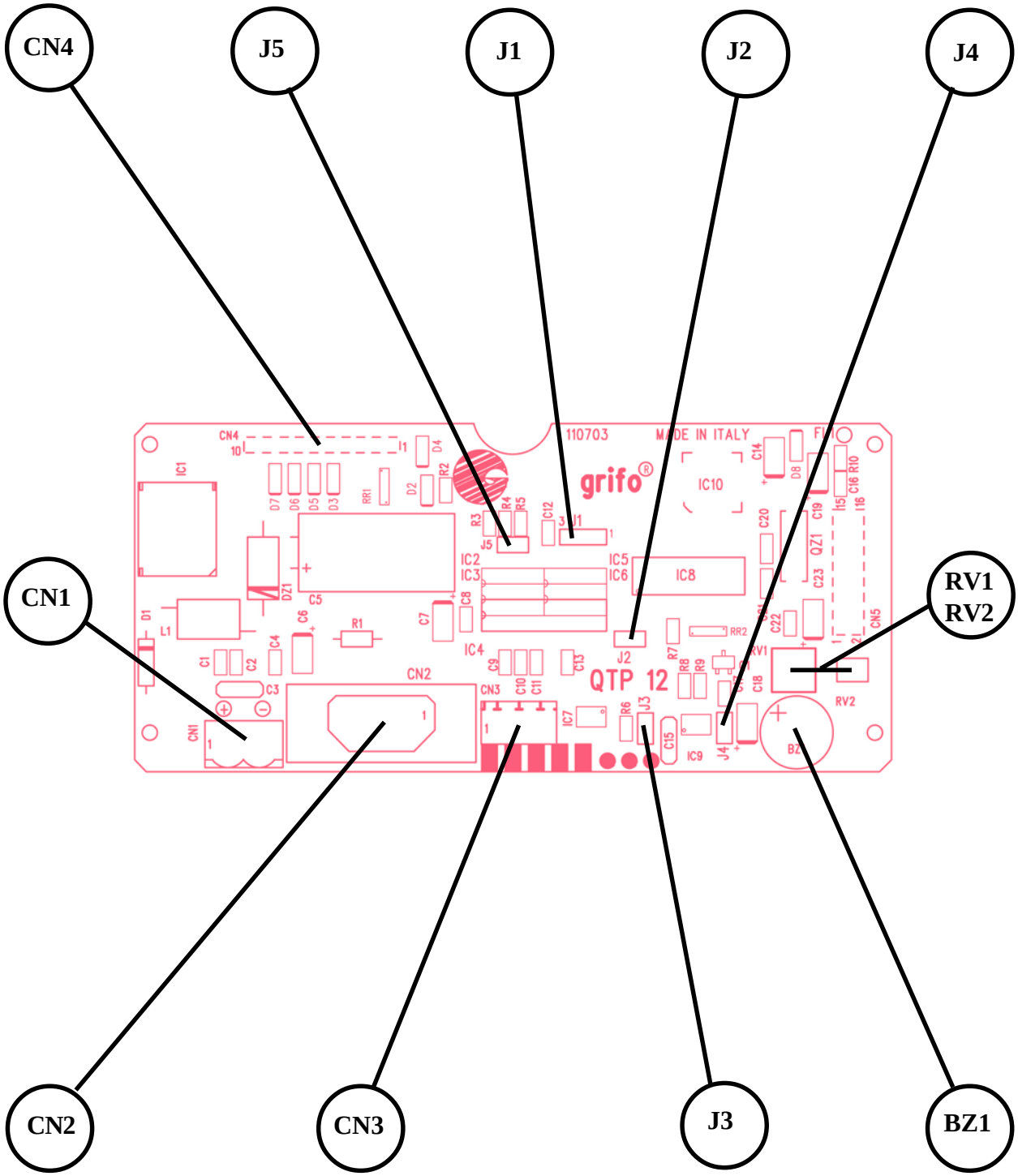


FIGURE 6: JUMPERS, CONNECTORS, TRIMMER, ETC. LOCATION

CONNECTIONS

QTP 12H terminal has 3 external and 1 internal connectors that can be linkeded to other devices or directly to the field, according to system requirements. Below are reported the pin outs, the meaning of the connected signals (including their directions) and some connection examples, that simplify and speed the installation phase. In addition the figures 5 and 6 show the connectors position on the board to simplify their recognitions.
Most part of connectors are accessible from the back of the alluminum container, through a proper breaking in the rear side that allows comfortable insertion and deinsertion.

CN1 - POWER SUPPLY CONNECTOR

CN1 is a vertical, 2 pins, male, quick release screw terminal connector, with 5 mm pitch.
On CN1 must be connected the single power supply voltage for the terminal that can be one out of three different types, as described by following figures

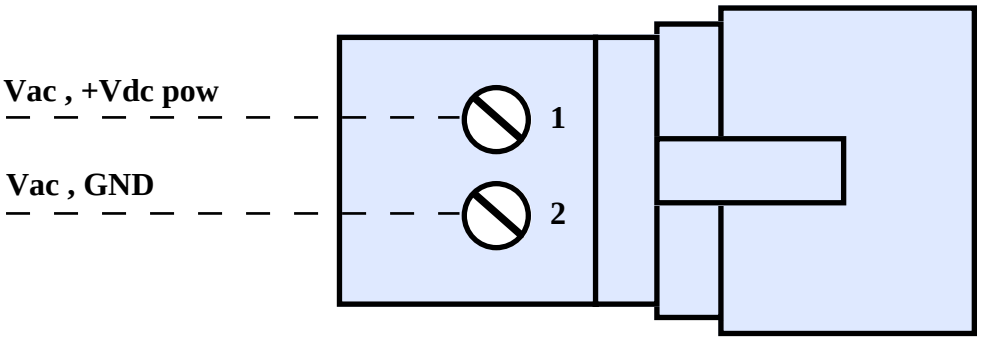


FIGURE 7: CN1 - POWER SUPPLY CONNECTOR

Signals description:

Vac = I - AC power supply lines connected to on board switching section; these signals must be in the range **8÷24 Vac**.

+Vdc pow = I - DC power supply lines connected to on board switching section (**+10÷+40 Vdc**) or stabilized (**+5 Vdc**) voltage connected to on board logic, according to ordered configuration.

GND = - Ground signal for DC power supply.

NOTE For further information about power supply configurations, please refer to paragraph POWER SUPPLY.

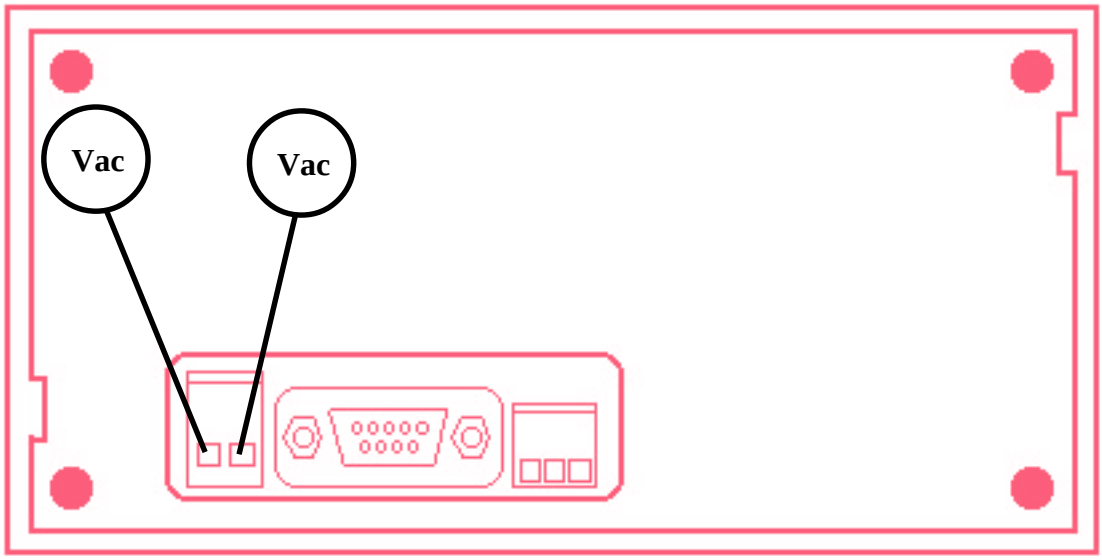


FIGURE 8: AC POWER SUPPLY 8÷24 VAC

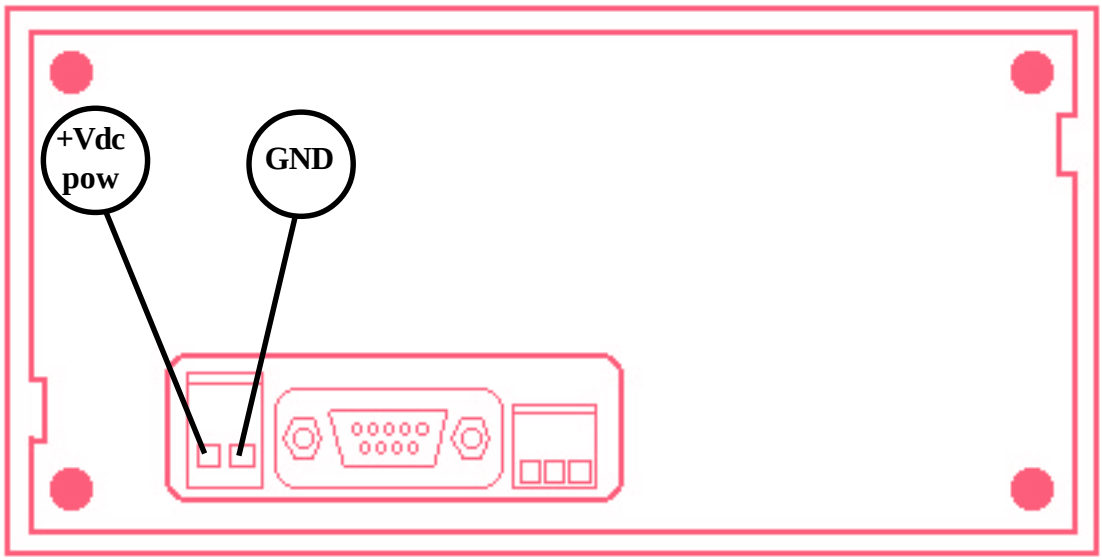


FIGURE 9: DC POWER SUPPLY +10÷40 VDC

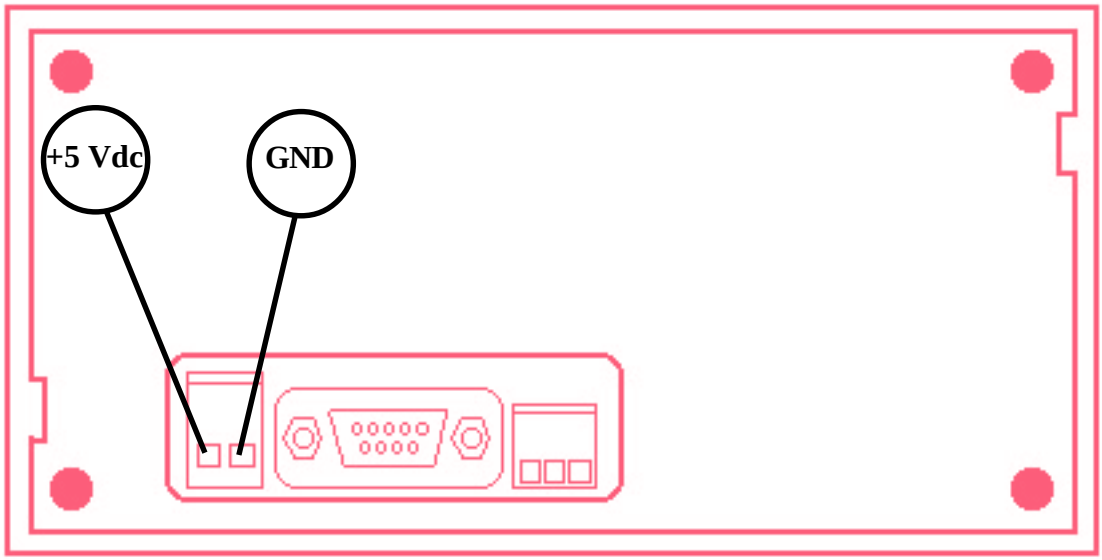


FIGURE 10: STABILIZED POWER SUPPLY +5 VDC

CN3 - CAN INTERFACE CONNECTOR

CN3 is a vertical, 3 pins, male, quick release screw terminal connector, with 3.5 mm pitch. Through CN3 must be connected the CAN serial communication line by following the standard rules defined by the same protocol. Signals placement has been designed to reduce interference and to obtain a fast and comfortable node connection on the field CAN bus.

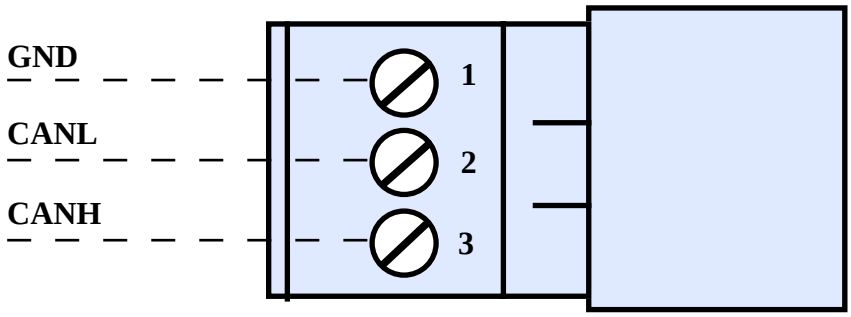


FIGURE 11: CN3 - CAN INTERFACE CONNECTOR

Signals description:

CANH = I/O - Differential line high for CAN interface.
CANL = I/O - Differential line low for CAN interface.
GND = - Ground signal.

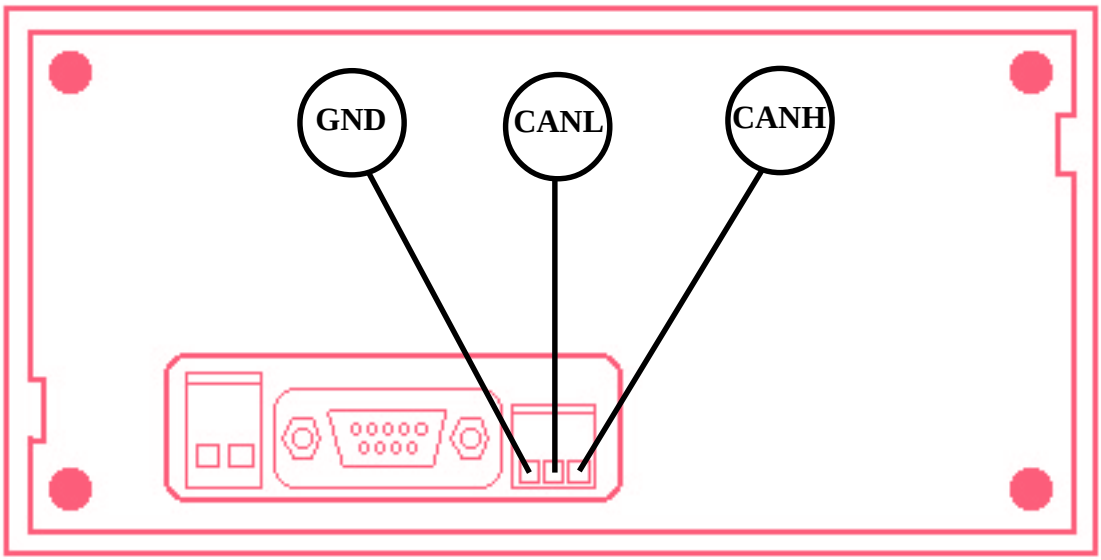


FIGURE 12: CAN LINE CONNECTION

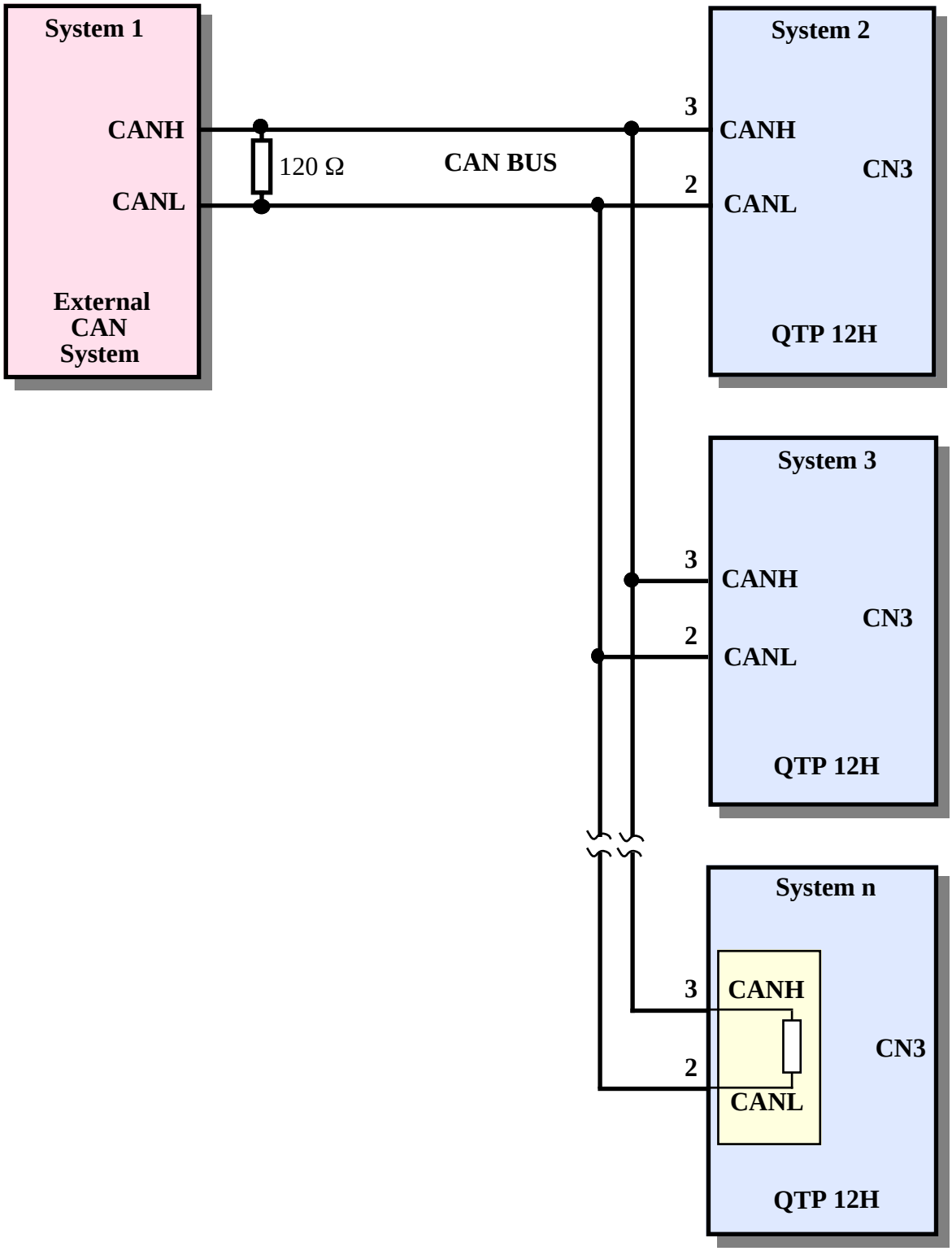


FIGURE 13: CAN INTERFACE CONNECTION EXAMPLE

Please remind that a CAN network must have two termination resistors (120 Ω) placed at its extrems, respectevely near the master unit and the slave unit at the greatest distance from the master.
On **QTP 12H** the terminating circuitry is already installed: it can be connected or not through specific jumper, as explained later.
When the system to link on the CAN line have very different potentials, it is possible to connect also the grounds of the same systems, that is pin 1 of CN3. In this way any possible problems of communication and/or incorrect working, are solved.

CN4 - LOCAL SETUP CONNECTOR

CN4 is a 10 pins, 90°, male, strip connector, with 2.54 mm pitch.
Through CN4 can be connected the four external keys that are used by **QTP 12H** firmware to manage the local setup for configuration. For this connector it is not necessary the complete descriptions of the available signals, but only the information about the keys connection.

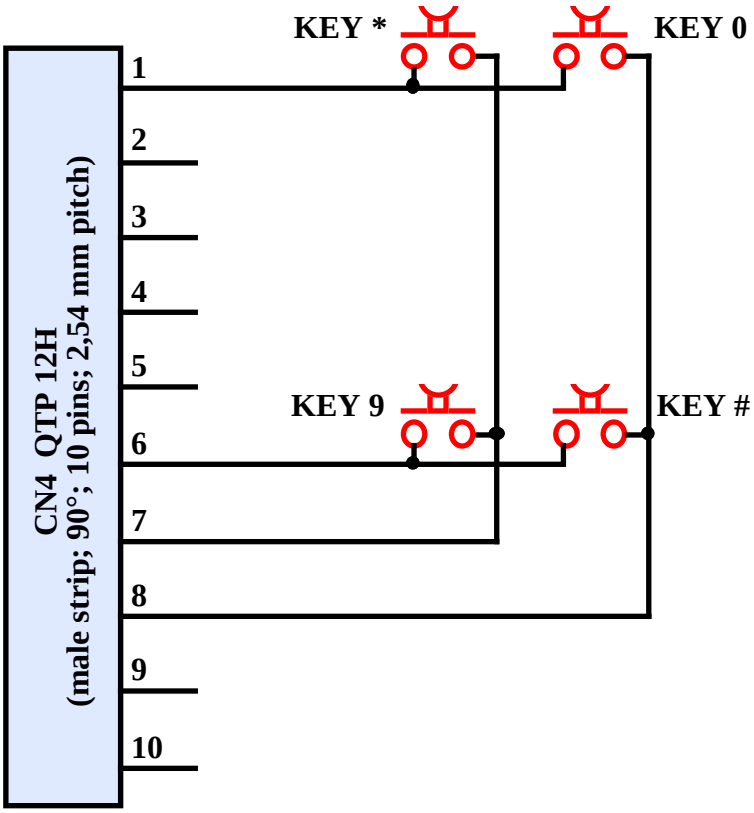


FIGURE 14: KEYS CONNECTION FOR LOCAL SETUP

Further information about the use of these keys are reported in the specific paragraph LOCAL SETUP, while the following notes concern their connection:

- the CN4 connector is not accessible from the rear break of the carter, dedicated to connectors: to allow its connection it is necessary to open the container, as described on APPENDIX C of this manual;
- the keys can be any type of normally open key or push button, that ensure a low contact resistance;
- the connection wires described on figure 14, must be shorter than 50 cm, except when the external environment is well disposed and enlargement become possible;
- to simplifies the connection of external keys it can be used also the product **KEY-12**, that coincides with a panel provided of a membrane keyboard with 12 keys, with dimensions and connections perfectly compatible with CN4. This product is an accessory that can be directly ordered to **grifo®**.
- normally local setup is executed only one time from the customer or the installer, that configures the **QTP 12H** according with the requirements of the developed application. For this reason, the connection of the keys to CN4 is necessary only in this preparation phase, when the the container is already opened for the other personalization operations.

CN2 - SERIAL LINE CONNECTOR

CN2 is a D type, 9 pins, female, vertical connector.
On CN2 are available all the signals of the asyncronous serial line, buffered with one of the diffused electric standards RS 232, RS 422, RS 485 or Current loop, that allows the complete management of the panel. 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 directives of the used standard.

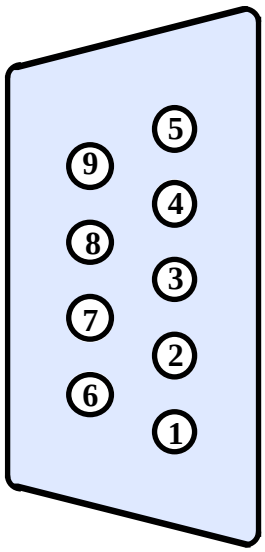


FIGURE 15: CN2 - SERIAL LINE CONNECTOR

Pin	*Signal*	*Direction*	*Description*
RS 232 serial line:			
2	**RX RS232**	= I	- Receive data for RS 232.
3	**TX RS232**	= O	- Transmit data for RS 232.
5	**GND**	=	- Ground signal.
RS 422 serial line:			
1	**RX- RS422**	= I	- Negative receive data for RS 422.
2	**RX+ RS422**	= I	- Positive receive data for RS 422.
3	**TX- RS422**	= O	- Negative transmit data for RS 422.
4	**TX+ RS422**	= O	- Positive transmit data for RS 422.
5	**GND**	=	- Ground signal.
RS 485 serial line:			
1	**RXTX- RS485**	= I/O	- Negative receive and transmit data for RS 485.
2	**RXTX+ RS485**	= I/O	- Positive receive and transmit data for RS 485.
5	**GND**	=	- Ground signal.
Current loop serial line:			
9	**RX- C.L.**	= I	- Negative receive data for Current loop.
8	**RX+ C.L.**	= I	- Positive receive data for Current loop.
7	**TX- C.L.**	= O	- Negative transmit data for Current loop.
6	**TX+ C.L.**	= O	- Positive transmit data for Current loop.

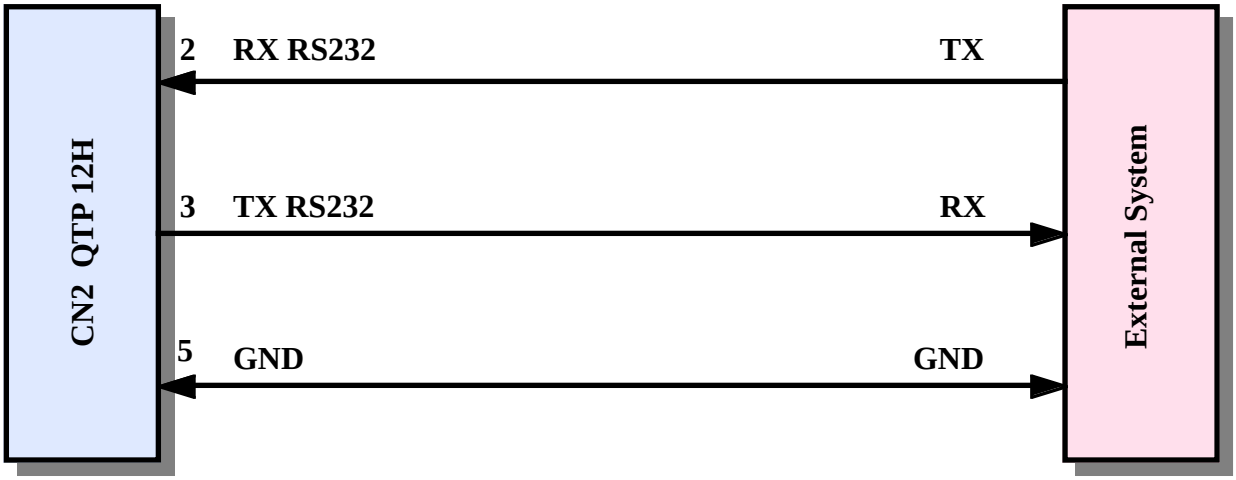


FIGURE 16: RS 232 POINT TO POINT CONNECTION EXAMPLE

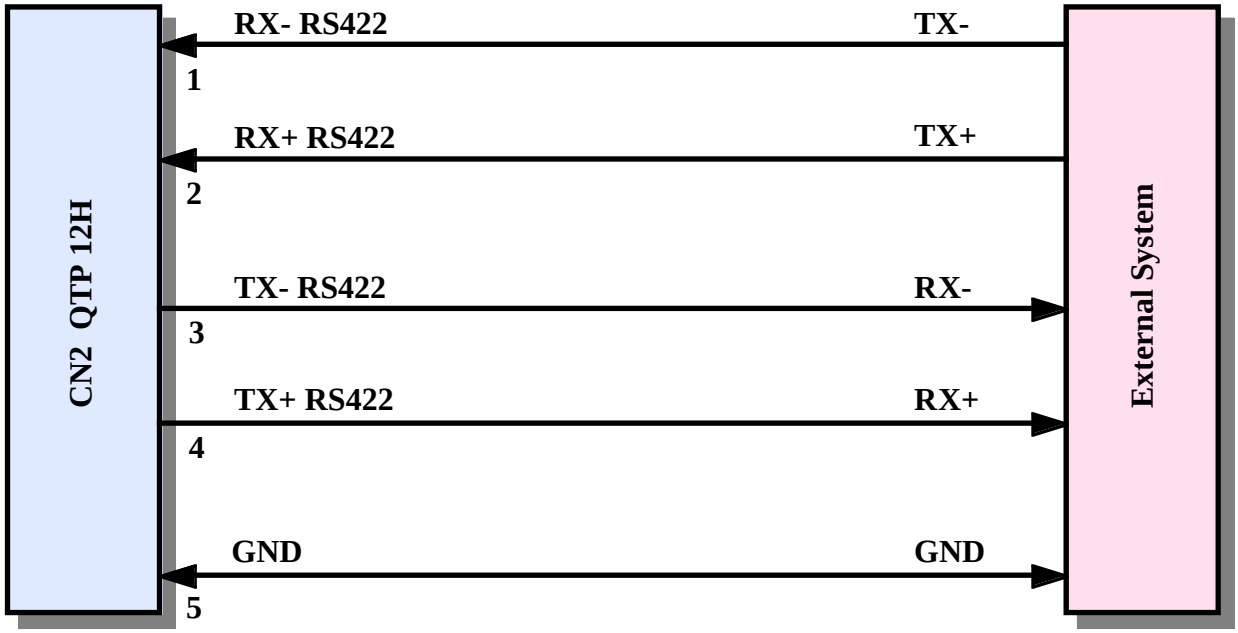


FIGURE 17: RS 422 POINT TO POINT CONNECTION EXAMPLE

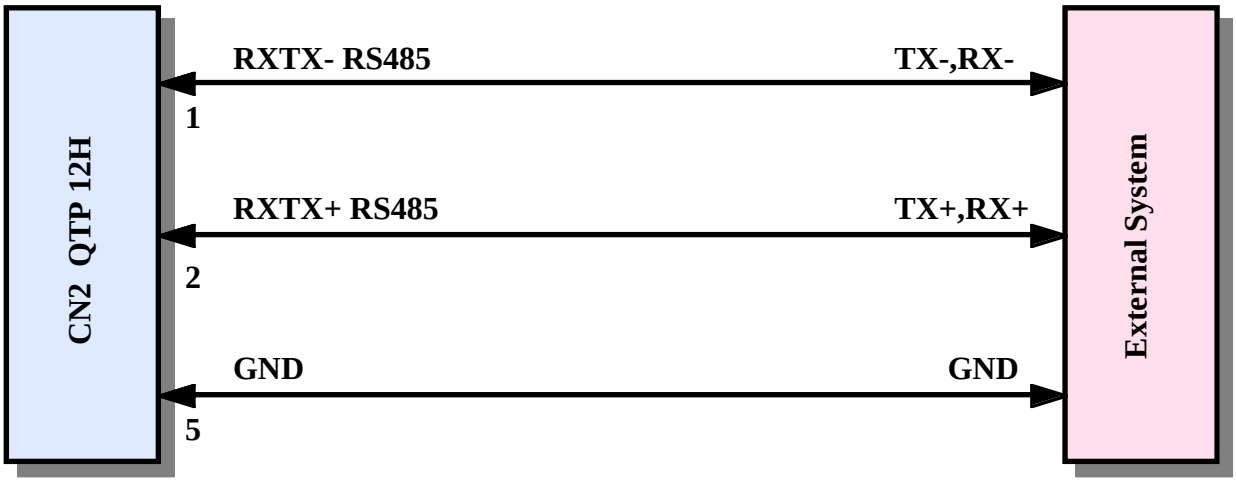


FIGURE 18: RS 485 POINT TO POINT CONNECTION EXAMPLE

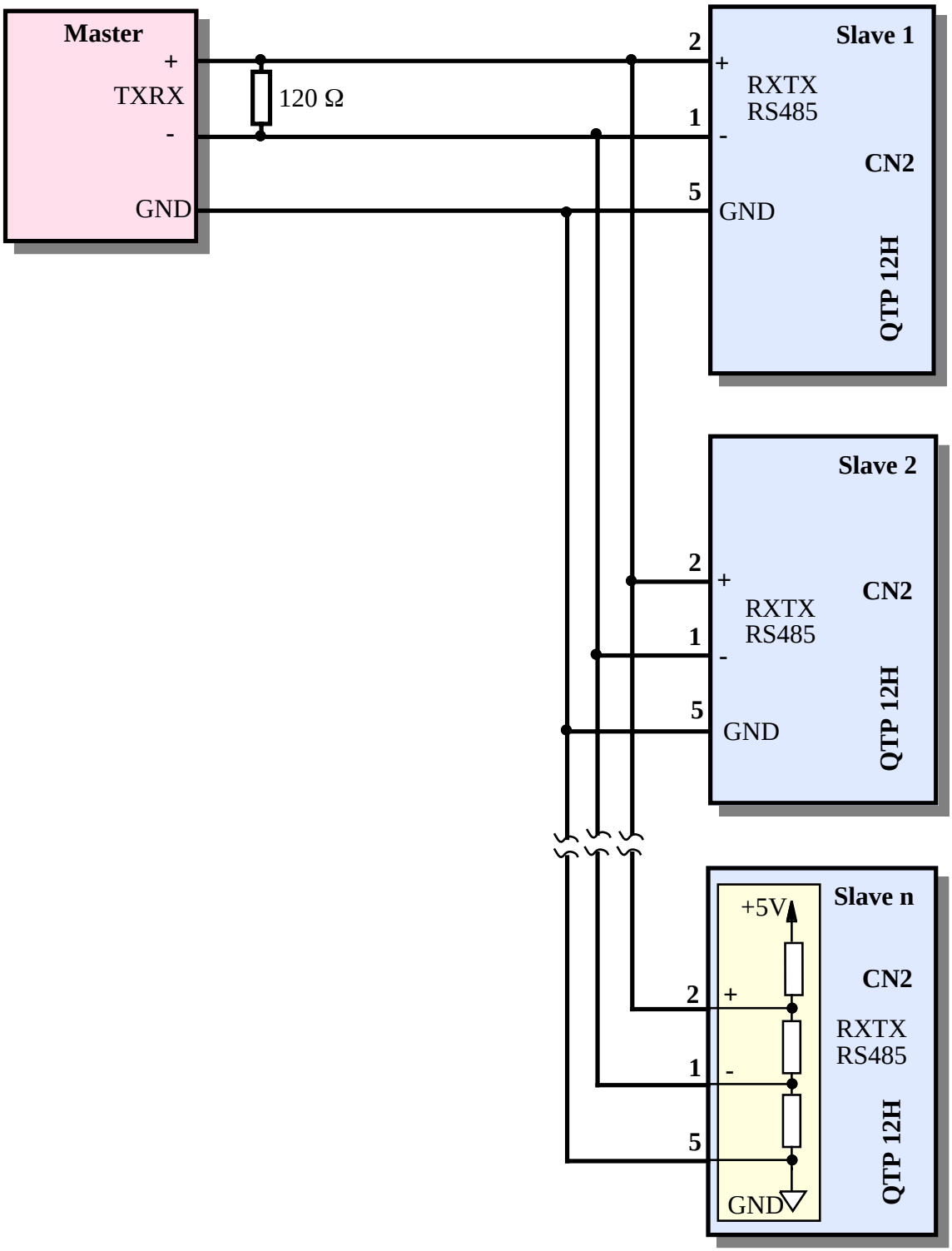


FIGURE 19: RS 485 NETWORK CONNECTION EXAMPLE

Please remark that in a RS 485 network two forcing resistors must be connected across the net and two termination resistors (120 Ω)must be placed at its extrems, respectevely near the Master unit and the Slave unit at the greatest distance from the Master.
Forcing and terminating circuitry is installed on **QTP 12H** 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 refr to TEXAS INSTRUMENTS Data-Book , "*RS 422 and RS 485 Interface Cicuits*", the introduction about RS 422-485.

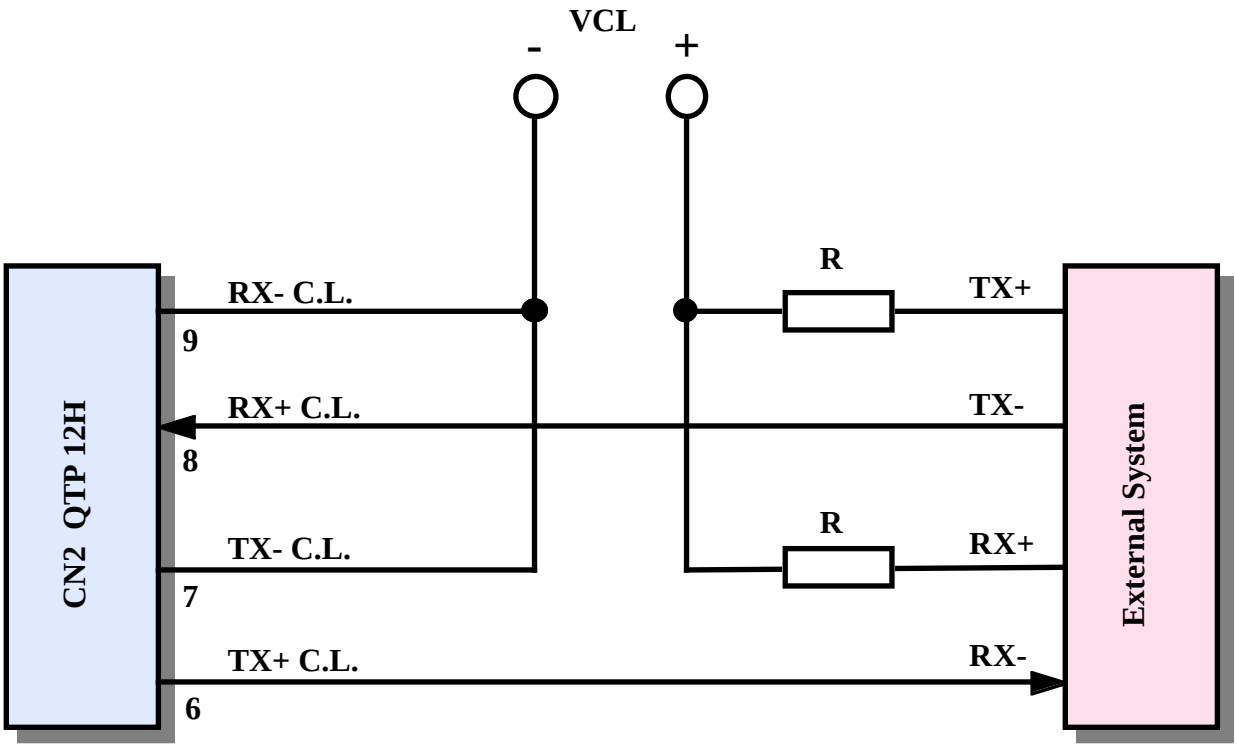


FIGURE 20: CURRENT LOOP 4 WIRES POINT TO POINT CONNECTION EXAMPLE

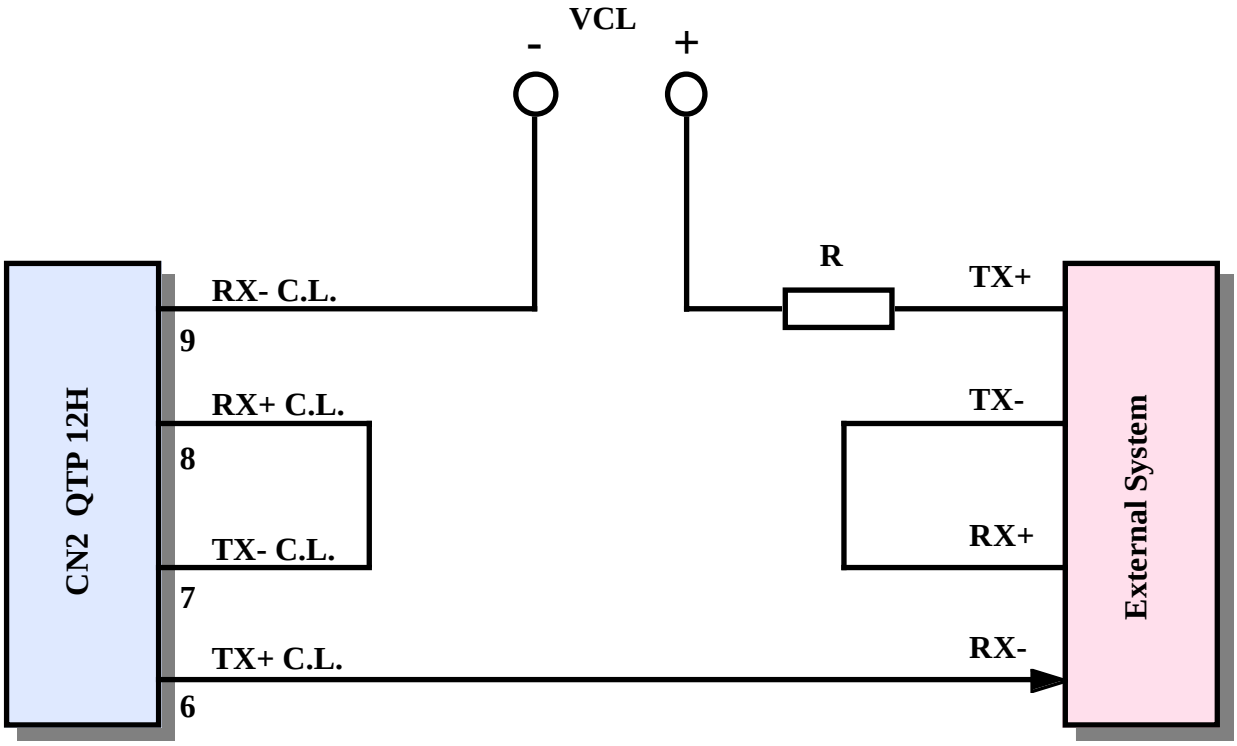


FIGURE 21: CURRENT LOOP 2 WIRES POINT TO POINT CONNECTION EXAMPLE

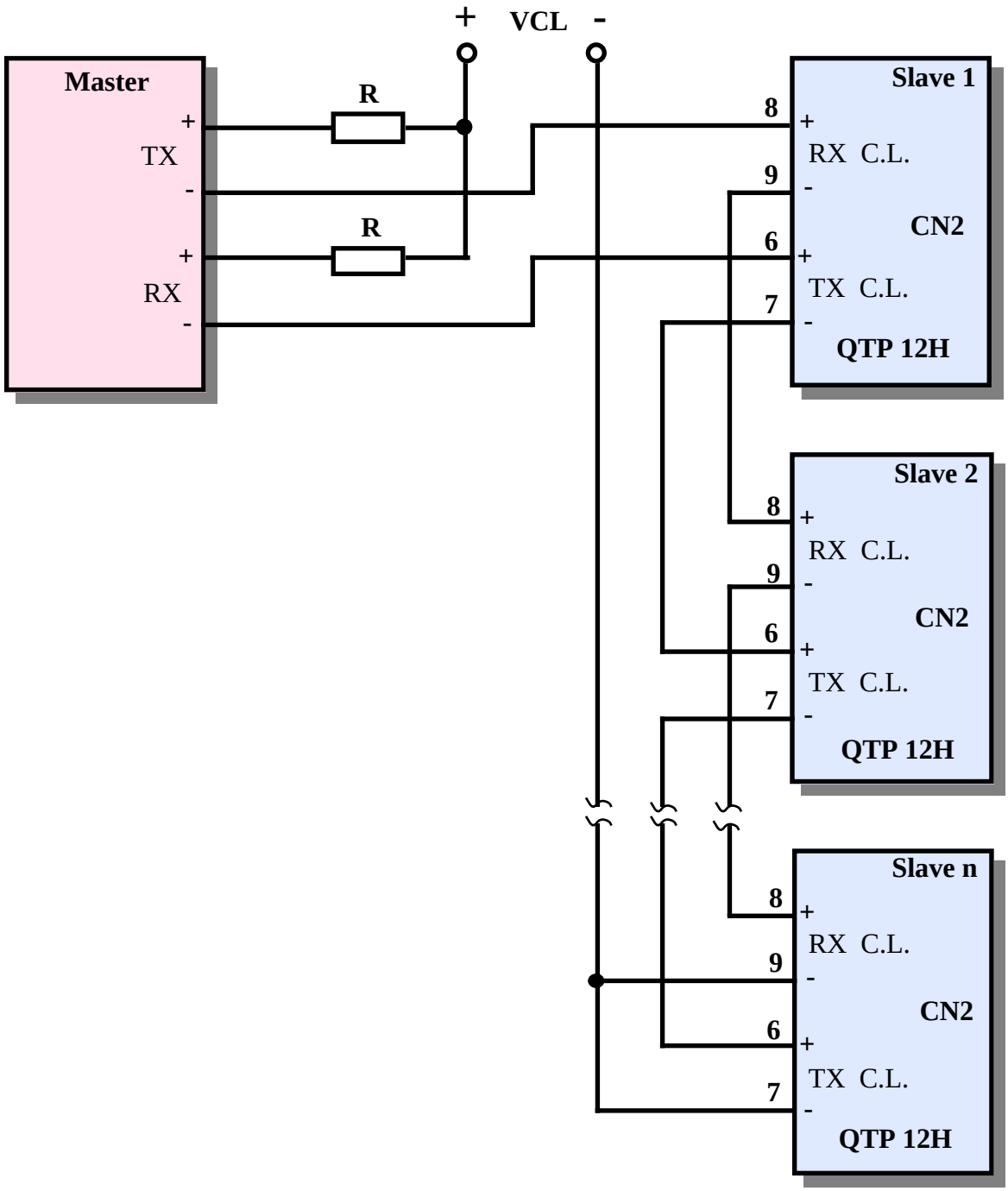


FIGURE 22: CURRENT LOOP NETWORK CONNECTION EXAMPLE

Possible Current Loop connections are two: 2 wires and 4 wires. These connections are shown in figures 20÷22 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 cosidering 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** cwith a 20 mA current;
- in case of shortciruit each transmitter must dissipate at most **125 mW**;
- in case of shortciruit each receiver must dissipate at most **90 mW**.

For further info please refer to HEWLETT-PACKARD Data Book, (**HCPL 4100** and **4200** devices).

CONTRAST REGULATION TRIMMER

On **QTP 12H** board there is a trimmer that defines the contrast on LCD display. This trimmer, named RV1 or RV2 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 6.

JUMPERS

On **QTP 12H** five jumpers for card configuration and connecting them, the user can perform some selections that regards the working conditions of the card. Here below there is the jumpers list, location and function in the possible connection modalities:

JUMPER	CONNECTION	PURPOSE	DEF.
J1	position 1-2	Configures serial line for RS 485 standard elettric protocol (2 wires half duplex).	
	position 2-3	Configures serial line for RS 422 standard elettric protocol (4 wires half duplex or full duplex).	*
J2 , J5	not connected	Do not connect termination and forcing circuitery to RS 422, RS 485 serial line.	*
	connected	Connect termination and forcing circuitery to RS 422, RS 485 serial line.	
J3	not connected	Does not connect 120 Ω termination resistor to CAN line.	*
	connected	Connects 120 Ω line termination resistor to CAN line.	
J4	not connected connected	Reserved.	*

FIGURE 23: JUMPERS TABLE

To recognize these valid connections, please refer to the board printed diagram (serigraph) or to figure 25 of this manual, where the pins numeration is listed; for recognizing jumpers location, please refer to figure 6.
In previous table 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 the **QTP 12H** jumpers are reported in the following paragraphs, that describe the sections where the same jumpers are used.

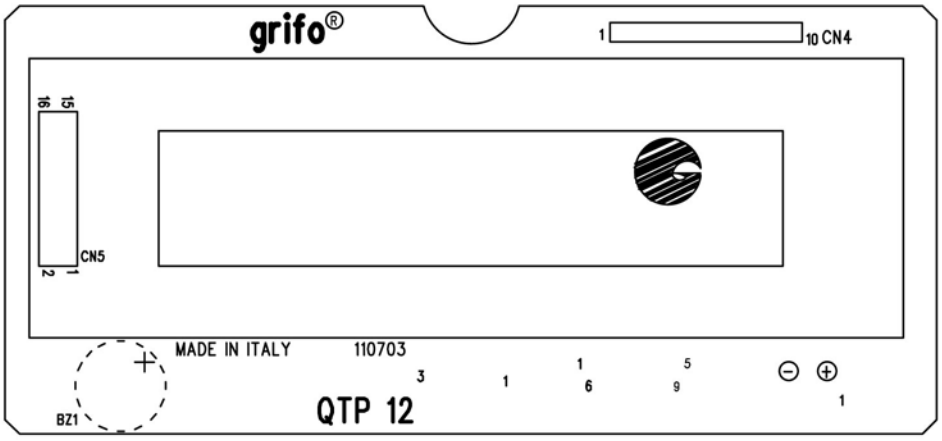


FIGURE 24: COMPONENTS MAP SOLDER SIDE

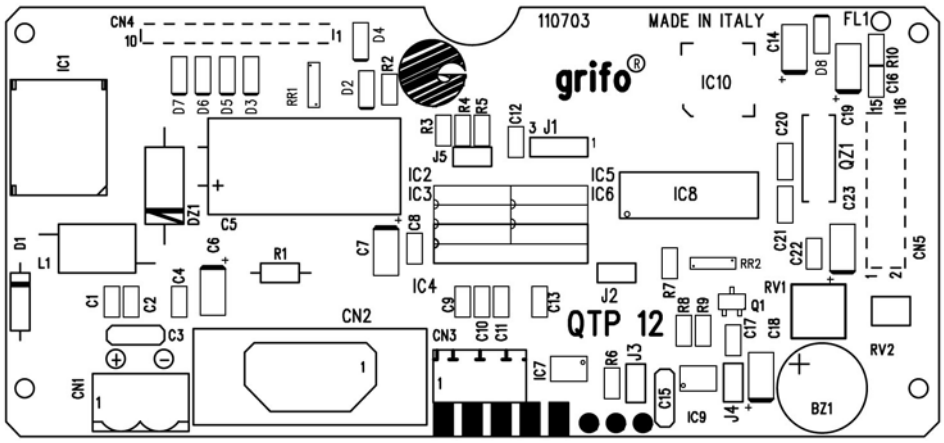


FIGURE 25: COMPONENTS MAP COMPONENTS SIDE

SERIAL LINE CONFIGURATION

Serial line of **QTP 12H** 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 (see homonymous paragraph).
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)

			IC4	= driver MAX 202
			IC2	= no device
J1	=	indifferent	IC5	= no device
J2 , J5	=	not connected	IC3	= no device
			IC6	= no device

- SERIAL LINE IN CURRENT LOOP (option **.CLOOP**)

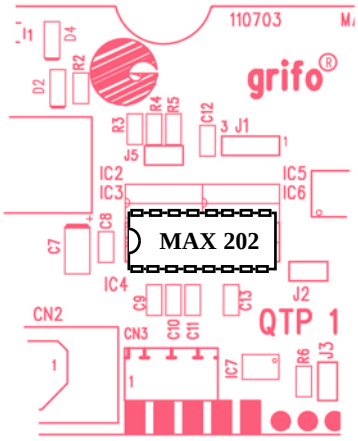
			IC4	= no device
			IC2	= no device
J1	=	indifferent	IC5	= no device
J2 , J5	=	not connected	IC3	= driver HP 4200
			IC6	= driver HP 4100

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 20÷22. Current loop interface can be employed to make both point to point and multi points connections through a 2 wires or a 4 wires connection.

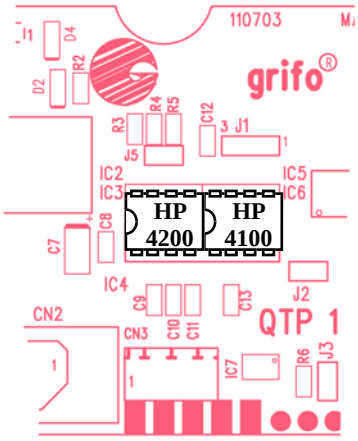
- SERIAL LINE IN RS 422 (option **.RS 422**)

			IC4	= no device
			IC2	= driver SN 75176 or MAX 483
J1	=	position 2-3	IC5	= driver SN 75176 or MAX 483
J2 , J5	=	(*)	IC3	= no device
			IC6	= no device

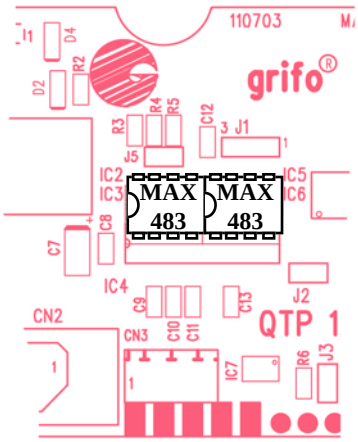
RS 422 electric protocol can be used to make 4 wires, full duplex, connections either in multi points or point to point systems.
Transmitter abilitation, essential in networks connections, is managed directly by **QTP 12H** firmware by selecting the master slave logic protocol.



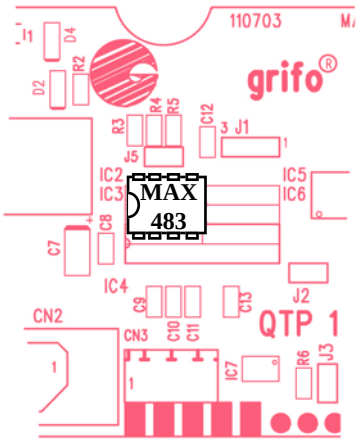
Serial line in RS 232



Serial line in current loop



Serial line in RS 422



Serial line in RS 485

FIGURE 26: LOCATIONS OF DRIVERS FOR SERIAL COMMUNICATION

- SERIAL LINE IN RS 485 (option **.RS 485**)

			IC4	= no device
			IC2	= driver SN 75176 or MAX 483
J1	=	position 1-2	IC5	= no device
J2 , J5	=	(*)	IC3	= no device
			IC6	= no device

In this modality the signals to use are pins 1 and 2 of connector CN2, that become transmission or reception lines according to the status defined by firmware; the last must be configured with logic protocol master slave. The RS 485 electric protocol can be used to make 2 wires half duplex connections both in multi points networks and point to point connection.

(*) When the RS 422 or RS 485 serial line are used, it is possible to connect the termination and forcing circuit on the line, by using J2 and J5 jumpers. This circuit must be always connected in case of point to point connections, while in case of multi points connections it must be connected olny in the farest boards, that is on the edges of the commmunication line.
During a power on, the RS 485 driver is in reception and RS 422 transmission driver is disabled, to avoid confilcts on line.

For further information about serial communication please refer to the connection examples of figures 16÷22.

CAN INTERFACE

Jumper J3 connects or does not connect CAN termination resistor, as described in figure 23. The CAN BUS must be a differential line with 60 Ω of impedance so termination reistors must be connected to obtain this impedance value.
In detail, this connection must be always made in case of point to point communications, while in multi points communications it must be connected only in the cards at the greatest distance, that is at the ends of the CAN line (please see example of figure 13).
Correct CAN termination contributes remarkably to correct communication; in fact the **QTP 12H** on board interface can suppres transients and avoids radio frequence and electromagnetic disturbs, only if connection to the filed is made correctly.
CAN line is not galvanically isolated (as descrived in following paragraph POWER SUPPLY) from board supply voltage. Ground of CAN line is connected to on board GND and it is available on a pin of CN3 connector. This latter can be used to equilibrate difference of potentials amongst several CAN systems, but also to shield physical connection, by using shielded cable, to obtain the greatest protection against external noise.

FIGURE 27: PHOTO OF QTP 12H-C2

FIGURE 28: PHOTO OF QTP 12H-F2

FIGURE 29: PHOTO OF QTP 12H-GF2

POWER SUPPLY

Terminal **QTP 12H** is provided with a power supply section that solves in a efficent 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, CAN interface and buzzer.
Here follow the voltages required according to card configuration together with the relative right connection:

Default version: This configuration includes a switching power supply that requires 10÷40 Vdc or 8÷24 Vac provided through CN1 (polarity must be respected in case of DC supply). This allows to supply the terminal using standard industrial nad commercial power sources like transformers, batteries, solar cells, etc. A comfortable and inexpensive solution for default version power supply can be the **EXPS-1** product that can be connected directly to the terminal starting from mains.
Please remark that on board switching supply is provided with single diode rectifier, so in case of DC supply, all ground signals of the terminal (GND) are at the same potential.
This is the default version, normally delivered without further specifications, in the order.

.5Vdc or **.ALIM** version: This configuration is not provided of any power supply section, so a +5 Vdc ± 5% stabilized supply voltage must be provided by an external source through CN1 (polarity must be respected also in this case). This allows to provide energy to the terminal through power supply, other cards, etc.
This version is a particular OEM configuration only, to directly agree upon **grifo®**.

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.
The **QTP 12H** is always provided with a **TransZorb™** protection circuit to avoid damages from incorrect voltages and/or break down of power supply section. It is also provided with a distribuited 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.

FIGURE 30: POWER SUPPLY EXPS-1 PHOTO

SOFTWARE DESCRIPTION

As already stated **QTP 12H** is a complete video terminal. It shows on the display any characters received from serial line, except the commands that are recognized and executed, and it transmits back, on the same serial line, to the master unit the possible results of the executed commands. These operations are automatically performed by on board firmware that is programmed and executed by the **QTP 12H** CPU.
The on board firmware manages also a local set up which allows to set the physic communication protocol by using the display of **QTP 12H** and a simple external circuitry.
This chapter describes the main features of **QTP 12H**, while the following one reports a detailed description of the recognized comand sequences, that can be used to benefit of all the functionalities of the terminal.
In corrispondence of the first order, on the received **grifo®** CD, are supplied many complete and usefull demo programs either in executable and source format; these can be used as received with no modifications, for a first test of the product and then changed, or partially used, to develop the user application program.

LOCAL SET UP

Thanks to a proper local set up mode, some parameters of communication protocol can be set and the EEPROM content can be restored. This mode can be easily and intuitively used, thanks to the on board display plus four external keys, that must be connected to CN4 internal connector.
In detail the user must:

a) Separate the group metallic carter+plastic frame from the group front panel+printed circuit. A simple pressure on backside **QTP 12H** connectors, or on the printed circuit from the backside connector window, is normally sufficient. If the front panel is fixed with two additional screws, this must be previously uncrewed (a detailed descriptionof these separation is reported in APPENDIX C).

b) Connects the 4 external keys to CN4 connector, as described in CN4 - LOCAL SETUP CONNECTOR paragraph. Please remind that the following description uses the names of the keys reported on figure 14 of this paragraph. When the accessory **KEY-12** has been ordered, it is sufficient to insert its female connector to CN4, by observing the pins numeration of both the printed circuit.

c) Yield the **QTP 12H** + keyboards on a bearing surfacethat allows the display reading and the contemporaneous pression of the keys.

d) Power on the **QTP 12H** and simultaneously press the keys ***** and **0** for at least half of a second.

e) At this point setup mode is entered, on the display appears the "**** Local Setup ****" string and with keys **9** and **#** the current configuration parameters, and its current values, shall be changed as below described:

f) Press the key **#** to change current menu, recognized by the following messages:

"**COMMUNICATION**"	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 digit of hexadecimal identification name
"**NAME (Hex)**"	second digit of hexadecimal identification name
"**EEPROM DATA**"	initializes data in EEPROM
"**SAVE and EXIT**"	to exit from set up mode

g) Press the key **9** to change current value ofdisplayed menu, with the following possibilities:

COMMUNICATION:	**Norm.** or **M.-S.** for normal or master-slave protocol	(*def.=Norm.*)
BAUD RATE:	**38400**, **19200**, **9600**, **4800**, **2400** or **1200** baud	(*def.=19200*)
STOP BIT:	**1** or **2** with normal protocol	(*def.=1*)
	1 with master-slave protocol	
KEYCLICK:	**ON** or **OFF**	(*def.=ON*)
NAME (Hex):	Changes the digit enclosed in "><" from **0** to **F**	(*def.=80H*)
EEPROM DATA:	**NOINI** or **INIT**	(*def.=NOINI*)
SAVE and EXIT:	exits setup and configures **QTP 12H** with selected parameters	

h) Once the necessary values have been set by using the modalities described in points **f** and **g**, select the **SAVE and EXIT** menu and press the key **9** to exit from local setup.

i) Removes the keys connected to CN4 and remount everything, by following the previous step in reversed order.

Once exited from setup mode, the selected parameters are saved on EEPROM and they are mantained until another local setup is executed; immediately after the terminal starts its normal functionality. The *default* values before reported are those setted at the end of testing phase, that is the configuration the user receives.
The setting of KEYCLICK doesn't care and so it can be freely defined with any one of the two possible values with no modifications of **QTP 12H** operations.
Available options for menus BAUD RATE and STOP BIT define the pyhisical communication protocol that has other two parameters unchangeable and set to no parity and 8 bits per character (or 9 bits if master slave is selected). Options of remaining menus are described in the following paragraphs.

NOTE
Please remind that setup mode can be entered only during power on, when previously described condition are recognized in fact if external keys are pressed at the same time during normal operation then setup mode will not start.
The local set up is normally executed only one time after the first installation, so it regards expert staff and not the final user of **QTP 12H** that use it as a simple operator interface unit. So the required four external keys must be connected only during this phase and normally, during the preparation, the container is already opened for the other personalization steps.

RECEIVE BUFFER

QTP 12H is provided with a reception buffer that simplify 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 command.
Naturally when commands that requires a long execution time (delete commands, EEPROM management commands, etc.) are continuosly received, the buffer will become full and wil loverflow. When overflow occours last location of the buffer is overwritten by each next received characters, and these are defenitely lost.
The master unit must stop the transmission until the **QTP 12H** 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 **QTP** for executing the required operations and to avoid the complete filling of reception buffer.

DATA STORED ON EEPROM

The on board EEPROM of **QTP 12H** is used to store data used and/or changed through the specific commands. The choice of EEPROM memory type has been performed to obtain the best warranties on data validity and endurance, naturally even when power supply is not available. The detailed description on each one of the data saved on EEPROM is reported in the following chapter, in the paragraphs relative to commands that use them.
When the card is received from an order or a reparation, the EEPROM is supplied already set with its default values, that are:

presence byte	->	255 (FFH)
patterns of user defineable characters	->	255 (FFH)
messages	->	255 (FFH)

Whenever the user desires to reset the default configuration on all data saved in EEPROM, the firmwares provide this possibility through the menu EEPROM DATA of local setup. The described data can be left unchanged by selecting the NOINI option, or they can be set to their default values, by selecting the INIT option. When INIT option has been selected, once exit from local setup a string is shown on the display with a scrolling bar of ***** (asterisk) that inform about the status progress of the operation. The displayed ***** are 10 and the execution time of the described initialization phase is 20 seconds approximately.
The user must be very careful with EEPROM initialization, in fact all previously saved data are defenitely lost.

CHARACTERS VISUALIZATION ON DISPLAY

QTP 12H shows on its display all the received characters having a code included in the range **0÷255** (**00÷FF Hex**) including the one that identifies a command sequence (27 = 1BH), as described later. 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 corrispondence between codes and displayed characters is defined by the following rules:

Codes	Characters
0 ÷ 15 (00÷0F Hex)	User defineable
16 ÷ 31 (10÷1F Hex)	Special and different, according with installed display
32 ÷ 127 (20÷7F Hex)	Standard ASCII
128 ÷ 255 (80÷FF Hex)	Special and different, according with installed display

To allow representation of special characters, that have same codes of some one character commands, a specific command has been provided that selects the operating mode of **QTP 12H** among the two available:

command the special characters are not displayed and the relative commands are executed;
representation the special characters are always displayed.

After a power on it is automatically selected the command mode to make immediately utilizable each functionality. The commands composed by a sequence of two or more characters, that always start with ESC = 27 = 1BH, are anyhow interpreted and executed independently from the selected operating mode.
Every models of **QTP 12H** has 8 user characters that can be defined and/or stored and shown on the display, as explained in the further paragraph USER CHARACTERS COMMANDS. When on **QTP 12H-GF2** is selected the graphic mode, the user characters are not displayed, independently from the selected operating mode.
About special characters please refer to APPENDIX B and remind that it is possible to get also different display models, provided of different special characters, but everything must be directly prearranged with **grifo®**.

FIGURE 31: PHOTO OF CHARACTERS AVAILABLE ON QTP 12H-GF2

COMMUNICATION MODALITIES

QTP 12H features two different serial communication modalities on its asynchronous serial line:

Norm. Normal communication uses 8 bits per character, no parity plus stop bit and baud rate selected by by user through local setup. This communication mode is suitable for point to point connections in RS 232, RS 422 and Current loop.

M.-S. Master Slave communication uses 9 bits per character, no parity, one stop bit plus baud rate selected by user through local setup. This communication mode is suitable for point to point connections (with all electric protocols) or network (with RS 485, RS 422 and Current loop electric protocols). 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 or changed as described in SERIAL LINE CONFIGURATION paragraph.

MASTER SLAVE COMMUNICATION MODE

The master slave mode uses the 9 bits communication technique. In addition to the 8 data bits also a nineth bit is managed and it recognizes between a call coming from the "**master**" device to any of the "**slave**" structures, and a normal info transmission between master and the currently selected device.
When 9th bit is placed at 1, the 8 data bits of the same character has to contain the identification name, of the device required for communication, while by placing this particular bit at 0, it is possible to take out or supply info at the selected device.
When **QTP 12H** is used, the identification name must be that one selected by the local set up program on the "NAME (Hex)" entries.
When this byte is sent (with 9th bit set to 1) the **QTP 12H** recognizes itself and it waits the string containing chars, data or commands. In this string there must only be a comand that involves the return of an answer to send via serial line from **QTP** part; if there is more than one command with answers, the results of the remaining ones are ignored.
Between the transmission of a character and the next one there must be an time interval shorter than the **Time Out**, in fact when this delay is elapsed , the **QTP 12H** will consider the data string terminated and it will begin the answering phase. The Time Out values for each baud rates is below described:

Baud Rate	Time Out	Character transmission time
38400 Baud	550 µsec	287 µsec
19200 Baud	990 µsec	573 µsec
9600 Baud	1540 µsec	1146 µsec
4800 Baud	3080 µsec	2292 µsec
2400 Baud	6105 µsec	4584 µsec
1200 Baud	12100 µsec	9167 µsec

Master unit, once completed the transmission of the last character of the command sequence, must wait for:

"character transission time"+"Time out"

before to receive the first character of the answering string, transmitted by the **QTP 12H**. The answer consists in a byte containing the code **255** = **FF Hex**, if no answer are available, or a characters sequence that coincide with the answer of the command sent in the previous interrogation. Please remark that answer is provided also when master unit transmit a command sequence with only the identification name: this simplifies the check for available answers or invalid commands.

To explain better the master slave protocol, here follows an example where master unit sends three commands to **QTP 12H** (reading of version number, a string to show and a check for possible answers) with a 38.4K baud rate and identification name 80H:

Master	*QTP 12H*
Sends "Reading of version number" command, that is the characters sequence: 80H with nineth bit set to 1 1BH with nineth bit set to 0 56H with nineth bit set to 0 delay between characters lower than 550 µsec	Receives character of the command and verifies the Time Out of 550 µ sec
Waits for 837 µsec	Recognizes command sequence, executes the command and stores answer for next command
Receives one character of answer	Sends the answer, which is the code 255=data not available, with nineth bit set to 0
Sends a string to show on the display, that is the characters 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 µsec	Receives character of the command and verifies the Time Out of 550 µ sec
Waits for 837 µsec	Recognizes command sequence and shows on the display the characters of the string
Receives three answer characters with the version number previously requested	Transmits saved response which is the version number required by previous command, with nineth bit set 0
Sends check command for answer data, that is the characters sequence: 80H with nineth bit set to 1	Receives character of the command and verifies the Time Out of 550 µ sec
Waits for 837 µsec	Recognizes sequence without commands so performs no operation
Receives one or more characters corresponding to possible answer data	Sends the answer, which is the code 255 or possible data, with nineth bit set to 0

FIGURE 32: EXAMPLE OF MASTER SLAVE COMMUNICATION

Several demo programs, coded in different programming languages, are provided with **QTP 12H**. 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 master slave communication at high level , this means without having to worry about management of nineth bit, timings, possible electric protocol converters, etc. Also these libraries are provided with the first purchase, complete of user documentation, on a floppy disk or a CD rom.

NOTES:

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

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 properly, before any characters tranmission, according to this scheme:

If the character to transmit has EVEN number of "1" bits

*If 9th bit must be **1***	->	*Set parity to ODD*
*If 9th bit must be **0***	->	*Set parity to EVEN*

If the character to transmit has ODD number of "1" bits

*If 9th bit must be **1***	->	*Set parity to EVEN*
*If 9th 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.**

DEMO PROGRAMS

In correspondence of the first purchase together with **QTP 12H** it is supplied a floppy disk or a **grifo®** CD where are saved numerous demo programs that allow to test and weigh immediately the received product. These programs are provided both in executable and source format and they are coded with many high level programming languages (C, PASCAL, BASIC, etc.) either for PC platforms or **grifo®** microprocessor cards (as **GPC®**, Mini Module, etc).
As described in HOW TO START paragraph the programs named PRQTP12H.* use all the commands of **QTP** with a simple iteraction with the user; but many other demos are supplied capable, for example, to: drive **QTP** connected to a serial network, manage messages, use the master slave protocol with DLL libraries, perform scenographic presentation, etc. The user can examine the remarks of these demos and decide himself if they are interesting.
All the demo programs can be used directly or modified or partially used, according to applications requirements, without any autorizathion, license or additional cost. Furthermore in case of unusual requirements or combinations, specific new demo programs can be obtained, after proper agreenment with **grifo®**.

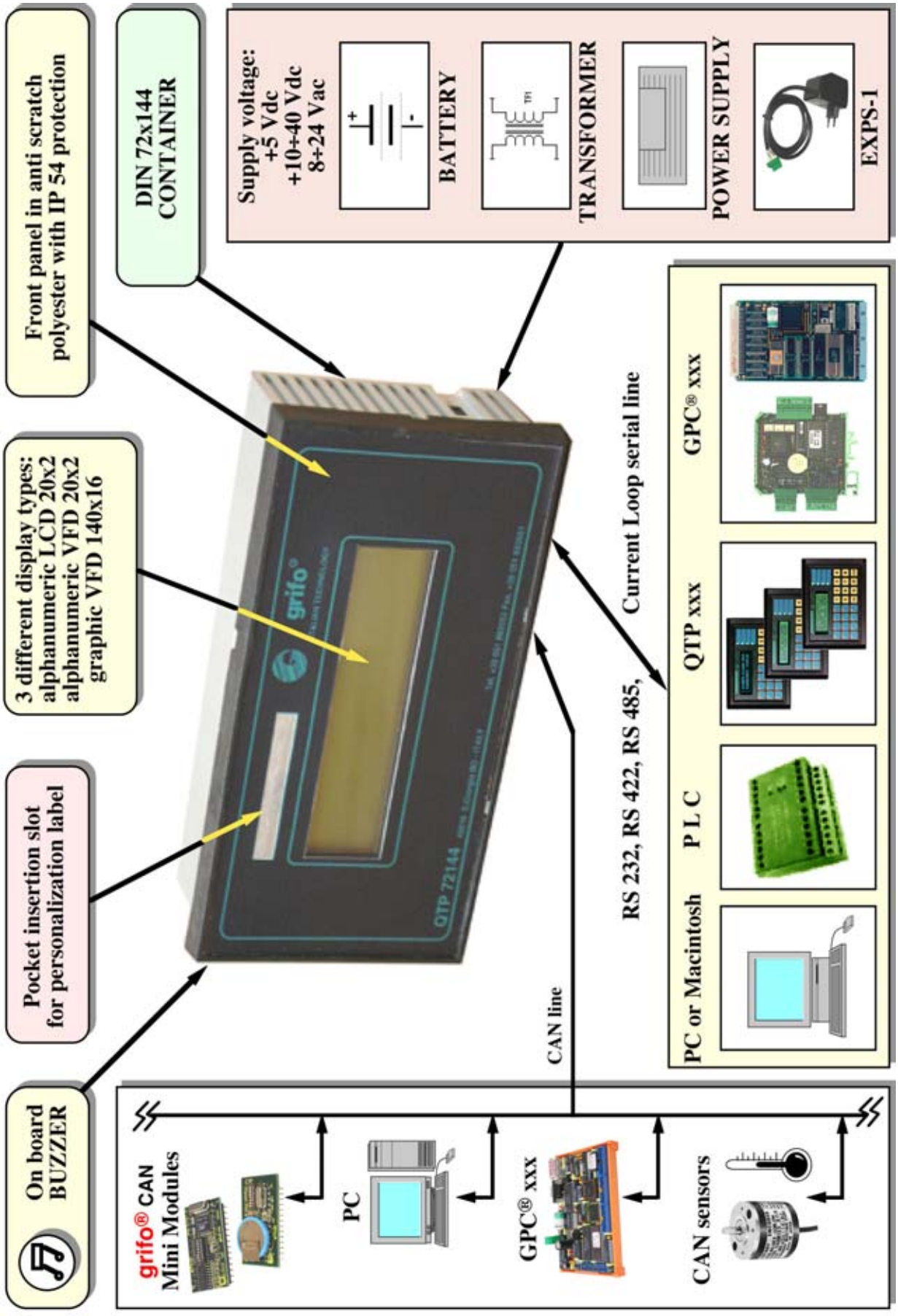


FIGURE 33: AVAILABLE CONNECTIONS DIAGRAM

HOW TO START

In this paragraph are listed the operations that must be performed to start using the **QTP 12H** in a pratical and fast way, solving the typical beginners problems. The paragraph contains interesting information even for the users that already know the product and its operating modes, in fact there is the description of a fast functional test. The following steps assume that a standard Personal Computer (provided of one free RS 232 serial line and a generic operating system, up to Windows 98) is available, to allow any user to execute them.

A) *Communication line connection:*

A1) Perform the serial connection described in figure 34 or on the other hand connect the two communication signals (TX RS232, RX RS232) and the reference ground signal (GND), to free COMx serial port of the PC. It can be easily discovered that this connection cable is reversed and it can be conveniently orderd to **grifo®** with the code CCR 9+9R.

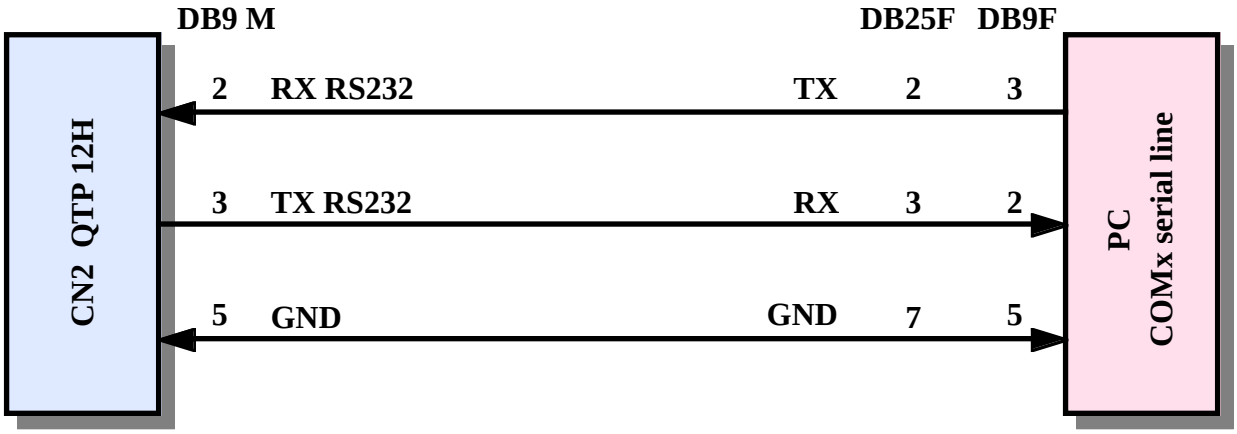


FIGURE 34: RS 232 CONNECTION WITH PC

A2) Supply power voltage on CN1 and check that buzzer is immediately disabled and a blinking block cursor is displayed in the left up corner of the display.

B) *Use of demo program:*

B1) On the floppy disks or **grifo®** CD rom received with the first purchase, it is available the file PRQTP12H.EXE, that contains the executable code for a demo program for PC that comunicate through RS 232 line with **QTP**. This file once found, must be copied in a comfortable folder on the hard disk of the used PC.

B2) Execute the program copied at point B1 and compile its start questions, by selecting the mounted display type. A this point press a key on PC to continue without execute the local setup, in fact the shown configuration coincides with the default one already set on the received **QTP 12H**.

B3) Carry on demo program execution and check that the operations described on PC monitor are correctly executed on **QTP**; when required interact with the same program in order to test all the available commands, until the end of demo program is reached.

C) *Use of terminal emulation:*

C1) Found the HYPERTERMINAL communication program on the PC, that normally is located on Windows menu: "Start | Program | Accessories | Communication", and execute it.

C2) Through the HYPERTERMINAL properties windows, setup the communication parameters to:

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

and wait the presentation of communication window.

C3) At this point type something on PC keyboard and check that pressed keys are shown on **QTP 12H** display. For compltetеness it can be tested also the effects of some commands by typing their codes sequences always on PC keyboard (this operation is simplified by contemporaneous pression of ALT key and of digits of the decimal code, on the numeric pad: for example to transmit the clear page command with decimal code 12, you can press contemporaneously the ALT key and first the keys 1 and then 2).

When during execution of the steps above described a problem or a malfunction is found, we suggest to read and repeat again all the steps carefully and ensure that default configuration values are saved, through local setup. If malfunction persists please contact directly grifo® technician.

FIGURE 35: PHOTO OF FRONT PANEL

COMMANDS

This chapter describes all the commands available in **QTP12H** firmware and each relative input and output parameters. The commands are divided in subgroups according with their functions and for each code, or codes sequence, there is a double description: the mnemonic one through the ASCII characters and the numeric one under decimal and hexadecimal format.
The commands respect the **ADDS Viewpoint** standard so all the sequences begin with **ESC** character corresponding to the **27** decimal code (**1B Hex**).
A rich list of demo programs (supplied in source and executable format) shows the pratical use modalities of commands: we suggest to add these demo programs, received during first purchase on CD or floppy disk, to this chapter documentation.
A summarized descriptions of all the available commands, their parameters and possible results answers, are reported in the table of APPENDIX A.

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, il will be moved to the Home position that is the first position in the first row.

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 that is the 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 below 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 PLACEMENT OF ALPHANUMERIC CURSOR

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** parameters.
These characters are the row and column values of the new desidered position referred to coordinate 0, 0 of the Home position, plus a constant offset of **32** (**20 Hex**). The position is expressed in alphanumeric mode so their valid values ranges respectively are 32÷33 and 32÷51. When row and/or column values are not compatible with the specified ranges, the command is ignored.
If, for example, the user wants to place the cursor on the second line, third column (row 1, column 2), then the following sequence must be sent:

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

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 same 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 EEPROM

In this paragraph are described some of the commands that directly manages some data saved on EEPROM of **QTP 12H**; there are other commands that indirectly use this memory device but they are described in next paragraphs.

REQUEST FOR EEPROM AVAILABILITY

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

This command checks if the **QTP 12H** is ready for management of its on board EEPROM. This command must be executed any time there are data to be read or write on this type of memory.
When **QTP 12H** firmware receives this command, it answers with the following codes:

6	*(06 Hex)*	*(ACK)*	*->*	***QTP 12H** ready*
21	*(15 Hex)*	*(NACK)*	*->*	***QTP* 12H** *not ready*

If firmware sends back the NACK code, it is not yet possible to memorize a new data on EEPROM or get an already saved one.

WRITE 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 desidered value, it allows for example, to verify that **QTP 12H** runs correctly, or if there are some communication problems on the serial line.

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

READ PRESENCE BYTE

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

The firmware sends back the value of its presence byte.
For example, this command can be useful to verify the presence, or the correct running, of the card and its firmware.

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

COMMANDS FOR GENERAL FUNCTIONS

In the following paragraphs are described all the general purpose commands that manage some features of **QTP 12H** firmwares. These commands do not come into the other subgroups and for this reason they are described in a proper paragraph.

READ FIRMWARE VERSION

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

The firmware returns a string of 3 characters containing the management firmware version that is resident and executed by **QTP 12H**.
For example with firmware version 1.3 the following characters will be returned:

49 46 51 or **31 2E 33 Hex** or **1.3**

FLUORESCENT DISPLAY BRIGHTNESS SETTING

Code: ***27 108 lum***
Hex code: ***1B 6C lum***
Mnemonic: ***ESC l ASCII(lum)***

Sets fluorescent display brightness to one of the four possible values, passed in **lum** parameter:

0	*(00 Hex)*	->	*Brightness at 100%*
1	*(01 Hex)*	->	*Brightness at 75%*
2	*(02 Hex)*	->	*Brightness at 50%*
3	*(03 Hex)*	->	*Brightness at 25%*

If parameter is not valid, command is ignored.

NOTE This command is available only with models **QTP 12H-F2** and **QTP 12H-GF2**. In case of **QTP 12H-C2** with LCD display, command must not be sent because it produces the visualization of an undesired character and a shift in all the next received data.

BEEP

Code: ***7***
Hex code: ***7***
Menomonic: ***BEL***

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

BUZZER ACTIVATION

Code: ***27 50 255 attr***
Hex code: ***1B 32 FF attr***
Mnemonic: ***ESC 2 ASCII(255) ASCII(attr)***

The on board buzzer is driven using attribute specified in **attr** parameter, that can assume the following values:

0 (00 Hex) -> buzzer OFF
255 (FF Hex) -> buzzer ON
85 (55 Hex) -> buzzer intermittent

If parameters is not valid, command is ignored.
The intermittent function is completely autonomous and it doesn't requires any intervent from user side.
For example, to activate the buzzer with intermittent attribute, the following sequence must be sent:

27 50 255 85 or **1B 32 FF 55 Hex** or **ESC 2 ASCII(255) U**

OPERATING MODE SELECTION

Code: ***27 65 mode***
Hex code: ***1B 41 mode***
Mnemonic: ***ESC A ASCII(mode)***

It defines the operating mode for the special characters (those provided of code less than 32 = 20H) and the single character commands. The selected modality is defined by value of **mode** parameter, with the following correspondence:

0 (00 Hex) -> Command mode
255 (FF Hex) -> Representation mode

If **mode** value is not one of the above described, the command is ignored. Further information about operating mode are reported in CHARACTER VISUALIZATION ON DISPLAY paragraph.

COMMANDS FOR MESSAGE MANAGEMENT

In the following paragraphs are described all the commands that manage messages, available in **QTP 12H** firmwares. The messages are 20 characters sequence that can be saved on board EEPROM and then reloaded or represented on display, simply by suppling 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 command. The **QTP 12H** firmware manages the scrolling messages visualization, too; with this feature on a single line of display can be shown more text that continuosly shift from right to left.
Moreover 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 **QTP** serially connected to PC.
QTP 12H features one EEPROM with a size of 2048 bytes that can storeup to 97 messages identified by a number from 0 to 96.

READING OF MAX MESSAGE NUMBER

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

This comand returns the number of the last messages that can be saved on EEPROM. It is always **96** (**60 Hex**) as previsously described. This command is important for other **QTP** models that has a variable messages number and it has been implemented on **QTP 12H** for compatibility and interchangeability with all grifo® operator panels.

MESSAGE STORAGE

Code:	**27 33 67 mess.n. chr.0÷chr.19**
Hex code:	**1B 21 43 mess.n. chr.0÷chr.13 Hex**
Mnemonic:	**ESC ! C ASCII(mess.n.) ASCII(chr.0)÷ASCII(chr.19)**

This command stores the 20 characters message, identified by **mess.n.** parameter, 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 **0÷255** (**0÷FF Hex**). The message number must be included in the range of **0÷96** to select one of the available messages.

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

MESSAGE READING

Code: ***27 33 69 mess.n.***
Hex code: ***1B 21 45 mess.n.***
Mnemonic: ***ESC ! E ASCII(mess.n.)***

This command reads the 20 characters message identified by **mess.n.** parameter, from the EEPROM and it returns this message, beginning from the first char of the string. At the end of the string the characters **CR+LF** are returned too, unless the master slave communication has been selected.
The message number must be included in the range of **0÷96** to select one of the available messages. If this number is out of range, the command is ignored.

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

VISUALIZATION OF N MESSAGES

Code: ***27 33 68 mess.n. n***
Hex code: ***1B 21 44 mess.n. n***
Mnemonic : ***ESC ! D ASCII(mess.n.) 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.n.** while the remaining messages are those immediately subsequents in EEPROM.
The **mess.n.** value and the number of the following messages defined by **n**, must be included in the range **0÷96**, to select only the available messages. If these number are out of range the command is ignored.
The **n** quantity of messages to be visualized depends on the model of the installed display. For alphanumeric mode all displays can show at most 40 characthers so the maximum number of messages is 2. In other words the **n** parameter can be set with a value in the range **1÷2** and if it is out of this range, command is ignored.
Once the command is executed 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:

27 33 68 10 2 or **1B 21 44 0A 02 Hex** or **ESC ! D LF STX**

NOTE: This command uses the on board EEPROM, so before executing it is better to check the EEPROM availability through the proper command; in fact if it is not ready the command is delayed until the operation under execution is completed.

SCROLLING MESSAGES VISUALIZATION

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

This command visualizes a **n.chr** characters string on the display first line in sliding mode. The string is shifted from right to left and so the user can visualize on a single line (the first) many information, more than the normal 20 characters.
The string of **n.chr** characters, begins with the first character of the **mess.n.** message already stored in EEPROM and continues with next characters always saved in following EEPROM messages.
The **mess.n.** value must be included in the range **0÷96**, to select one of the available messages. If the value is out of range this command is ignored.
Instead the **n.chr** parameter must range in the following values:

0	Stops the scrolling messages visualization (the **mess.n.** value doesn't care).
20÷200	Enables sliding visualization of the specified characters.

If **n.chr.** value is out of the specified ranges or it points after the last character of the last message stored in EEPROM, the command will be ignored.
The scrolling messages visualization is always performed on the first display line and the cursor position and attributes are mantained.
For example, if you wish to visualize a 35 characters string in sliding mode, composed 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 is better to check the EEPROM availability through the proper command; in fact if it is not ready the command is delayed until the operation under execution is completed. The message visualization in sliding mode is managed in background and so there is an increased firmware execution time, that causes a subsequent slowing down of commands interpretation. This is the reason why it is necessary to wait for few **msec** between the transmission of 20÷30 bytes data blocks when many information and/or commands are sent to **QTP 12H**. In this way misunderstanding and interpreting problems of the received data, caused by receive buffer overflow, are completely avoid.

COMMANDS FOR CURSOR ATTRIBUTES MANAGEMENT

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

CURSOR OFF

Code: ***27 80***
Hex code: ***1B 50***
Mnemonic: ***ESC P***

The cursor is disabled and it is not more visible.

STEADY STATIC CURSOR ON

Code: ***27 79***
Hex code: ***1B 4F***
Mnemonic: ***ESC O***

The cursor is enabled and so it is visible as a not blinking line placed under the current position character.

NOTE: This command can't be used when **QTP 12H-GF2** model is being used, that is when a graphic display with 140x16 pixels is mounted: in this condition the command has no effects.

BLINKING BLOCK CURSOR ON

Code: ***27 81***
Hex code: ***1B 51***
Mnemonic: ***ESC Q***

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

COMMANDS FOR USER CHARACTERS

QTP 12H lets the user define and show up to 8 user characters; those characters can be used to represent on display special characters, pseudo graphic characters, special symbols, etc. that are not still available in the same display (please refer to table in appendix B).
The user characters can be defined and saved with a pattern equal to a 5 x 8 pixels matrix, so organized:

	Columns of pixels				
Rows of pixels	Pat 0.4	Pat 0.3	Pat 0.2	Pat 0.1	Pat 0.0
	Pat 1.4	Pat 1.3	Pat 1.2	Pat 1.1	Pat 1.0
	Pat 2.4	Pat 2.3	Pat 2.2	Pat 2.1	Pat 2.0
	Pat 3.4	Pat 3.3	Pat 3.2	Pat 3.1	Pat 3.0
	Pat 4.4	Pat 4.3	Pat 4.2	Pat 4.1	Pat 4.0
	Pat 5.4	Pat 5.3	Pat 5.2	Pat 5.1	Pat 5.0
	Pat 6.4	Pat 6.3	Pat 6.2	Pat 6.1	Pat 6.0
	Pat 7.4	Pat 7.3	Pat 7.2	Pat 7.1	Pat 7.0

FIGURE 36: USER CHARACTERS PATTERN

The user characters representation is really simple in fact it is sufficient to send the proper code (0 to 7 equal to 8 to 15) with a previous setting of representation mode, through OPERATING MODE SELECTION command.
When the user character are saved, their patterns are written on EEPROM and then they are reloaded on display any time the terminal is powered on or initialized.

NOTE: Please remind that on **QTP 12H** with fluorescent displays the character has a 5 x 7 pixels matrix (Pat 0÷Pat 6) and the last row of the pattern is not displayed. Furthermore on **QTP 12H-F2**, the value of Pat 7.4 pixel defines the status of all the five pixels Pat 7.4÷Pat 7.0, or in other words it defines the status of underline attribute of the defined character.

DEFINITION OF USER CHARACTER

Code: ***27 66 nchar Pat 0 ÷ Pat 7***
Hex code: ***1B 42 nchar Pat 0 ÷ Pat 7***
Mnemonic: ***ESC B ASCII(nchar) ASCII(Pat 0) ÷ ASCII(Pat 7)***

After the two command identification codes, other 9 bytes must be sent with the following meaning:

nchar	*(0÷7)*	*(00÷7 Hex)*	->	*Number of user character to define*
Pat 0	*(0÷31)*	*(00÷1F Hex)*	->	*First byte of pattern equal to first high row of character.*
:	*:*		->	*:* *:* *:* *:*
Pat 7	*(0÷31)*	*(00÷1F Hex)*	->	*Seventh byte of pattern equal to last low row of character.*

This command loads on the display the pattern of the user character **nchar** with the value placed in the eight bytes **Pat 0 ÷ Pat 7**, as described in figure 36; the pattern is only defined but not saved, so when **QTP 12H** is turned off, the user character **nchar** doesn't mantain the supplied pattern.
For example to define the user character 5 as an empty rectangle with maximum size, the following sequence has to be sent:

27 66 5 31 17 17 17 17 17 17 31 or
1B 42 05 1F 11 11 11 11 11 11 1F Hex

DEFINITION AND MEMORIZATION OF USER CHARACTER

Code: ***27 33 66 nchar Pat 0 ÷ Pat 7***
Hex code: ***1B 21 42 nchar Pat 0 ÷ Pat 7***
Mnemonic: ***ESC ! B ASCII(nchar) ASCII(Pat 0) ÷ ASCII(Pat 7)***

After the three command identification codes, other 9 bytes must be sent with the following meaning:

nchar	*(0÷7)*	*(00÷7 Hex)*	->	*Number of user character to define and save*
Pat 0	*(0÷31)*	*(00÷1F Hex)*	->	*First byte of pattern equal to first high row of character.*
:	*:*		->	*:* *:* *:* *:*
Pat 7	*(0÷31)*	*(00÷1F Hex)*	->	*Seventh byte of pattern equal to last low row of character.*

This command loads on the display the pattern of the user character **nchar** with the value placed in the eight bytes Pat 0 ÷ Pat 7, as described in figure 36; moreover the pattern is also saved on EEPROM, so if **QTP 12H** is turned off and on, the user character **nchar** mantain the supplied pattern.

NOTE: This command uses the on board EEPROM, so before executing it is better to check the EEPROM availability through the proper command; in fact if it is not ready the command is ignored.
Execution time of the command is about 80 msec. When the command has been transmitted and several commands must follow, it is better to insert a delay to avoid receive buffer overflow.

COMMANDS FOR GRAPHICS

QTP 12H-GF2, featuring a 140 x 16 pixels graphic display, allows the possibility to show graphic images, histograms, characters with different font and size, diagrams, etc., through a short group of simple graphic commands.
All graphic commands are based on the smallest visible entity of display, that are points or pixels that are orgranized in the coordinates system described in the following figure:

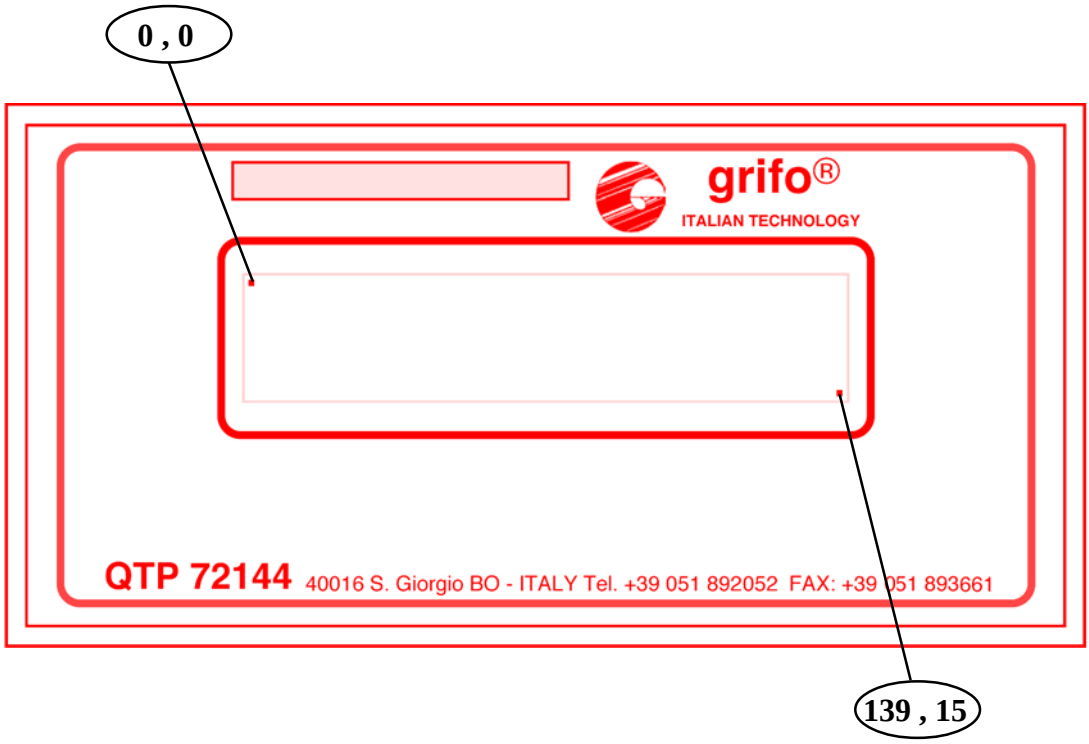


FIGURE 37: COORDINATES OF GRAPHIC DISPLAY PIXELS

NOTE: Please remind that following graphic commands can be used only on model **QTP 12H-GF2** while the remaining models do not recognize them as commands so they show the character of the command sequence on display.
In addition, **QTP 12H-GF2** recognizes both graphic and alphanumeric commands already described in previous paragraphs.

ALPHANUMERIC MODE SETTING

Code: ***27 208***
Hex code: ***1B D0***
Mnemonic: ***ESC ASCII(208)***

This command sets alphanumeric representation mode, which allows to use all alphanumeric commands, described in previous paragraphs. When command is executed, the cursor (if enabled) is shown in the last position decided by previous commands.
After power on it is immediately selected alphanumeric mode to make all functionalities available.

FIGURE 38: FIRST GRAPHIC EXAMPLE

FIGURE 39: SECOND GRPHIC EXAMPLE

GRAPHIC MODE SETTING

Code: ***27 209***
Hex code: ***1B D1***
Mnemonic: ***ESC ASCII(209)***

This command sets graphic mode, that enables the interpretation of caharacters sent to firmware as graphic data and not as commands. When this command is executed there is no effect on display, but characters received are no more checked for one characters commands and they go directly to graphic display. So one character commands are not executed until alphanumeric mode is restored. On the other hand, two or more characters commands, starting with ESC = 27 = 1BH, are always checked and executed, indipendently from selected mode.
After power on, alphanumeric mode is automatically selected by firmwares to allow the use of all commands.

GRAPHIC CURSOR ABSOLUTE POSITION

Code: ***27 206 y x 0***
Hex code: ***1B CE y x 00***
Mnemonic: ***ESC ASCII(206) ASCII(y) ASCII(x) NUL***

Moves the cursor to the point of coordinates **x** and **y**; the position indicated by these two parameters is absolute, so it is not affected by all other settings and it is indipendent from normal alphanumeric cursor placement. Characters to show, received after this command are displayed from indicated point, and they are drawn to the right and to the top.
Values of coordinates **y** and **x** must be in the range **0÷15** and **0÷139**, that are the size of used display.
If, for example, thegraphic cursor must be placed on pixel with coordinates (10 , 100), then the following sequence must be sent:

27 206 100 10 0 or **1B CE 64 0A 00 Hex** or **ESC ASCII(206) d LF NUL**

NOTE: Code **0** (**NUL**) described at the end of command sequence, is present for compatibility with future expansions and for compatibility with other terminals: it must be always transmitted anyway to ensure correct command execution.

GRAPHIC AREA SETTING

Code: ***27 241 x1 y1 x2 y2 cmd***
Hex code: ***1B F1 x1 y1 x2 y2 cmd***
Mnemonic: ***ESCASCII(241) ASCII(x1) ASCII(y1) ASCII(x2) ASCII(y2) ASCII(cmd)***

Defines graphic work area and the action to make on it. The graphic area has a top left corner equal to pixel with coordinates **x1** , **y1** and bottom right corner placed on coordinates **x2 , y2**. The values of **y1**, **y2** and **x1**, **x2** must be respectively in the ranges **0÷15** and **0÷139**, that are the size of used display.
Byte **cmd** selcts the action to perform on the defined graphic area and thus the function of the next bytes the **QTP 12H** receive, as described in the following list:

cmd = *67 (43 Hex) C -> Clears selected area.*

70 (46 Hex) F -> Fills selected area.

72 (48 Hex) H -> Draws the selected area with following horizontal graphic data, with horizontal shift.

73 (49 Hex) I -> Inverts selected area.

79 (4F Hex) O -> Draws a frame around selected area.

86 (56 Hex) V -> Draws the selected area with following vertical graphic data, with horizontal shift.

104 (68 Hex) h -> Draws the selected area with following horizontal graphic data, with vertical shift.

111 (6F Hex) o -> Deletes a frame around selected area.

118 (76 Hex) v -> Draws the selected area with following vertical graphic data, with vertical shift.

About commands that draw data in graphc area (H,h,V,v) next bytes sent to the terminal are used as graphic data that decide pixels activation of display. The correspondance between display pixels and bits of these bytes is explained in figures 38÷41 where all the four organization and shift modes are described.
Logic status **1** of a bit correspond to **activation** of corresponding pixel and viceversa logic status **0** of bit correspond to **deactivation** of pixel.
The draw selected area commands can be completed in two ways: by filling all the selected area or by interruption caused from another command; naturally this latter condition stops execution of the first command and so only pixels already received at that moment will be visualized.
For example, to draw an arrow like the one in the following figure, placed on the top left corner of display:

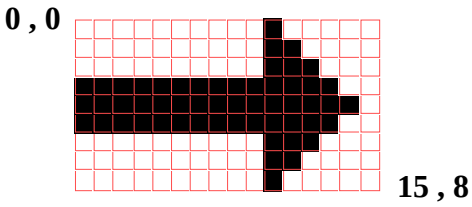


FIGURE 40: EXAMPLE OF GRAPHIC DRAWING

first send the command sequence:

27 241 0 0 15 8 72 or **1B F1 00 00 0F 08 48 Hex**

and then graphic data sequence:

0	**0**	**0**	**255**	**255**	**255**	**0**	**0**	**0**	**32**	**48**	**56**	**252**	**254**	**252**	**56**	**48**	**32**	or
00	**00**	**00**	**FF**	**FF**	**FF**	**00**	**00**	**00**	**20**	**30**	**38**	**FC**	**FE**	**FC**	**38**	**30**	**20**	**Hex**

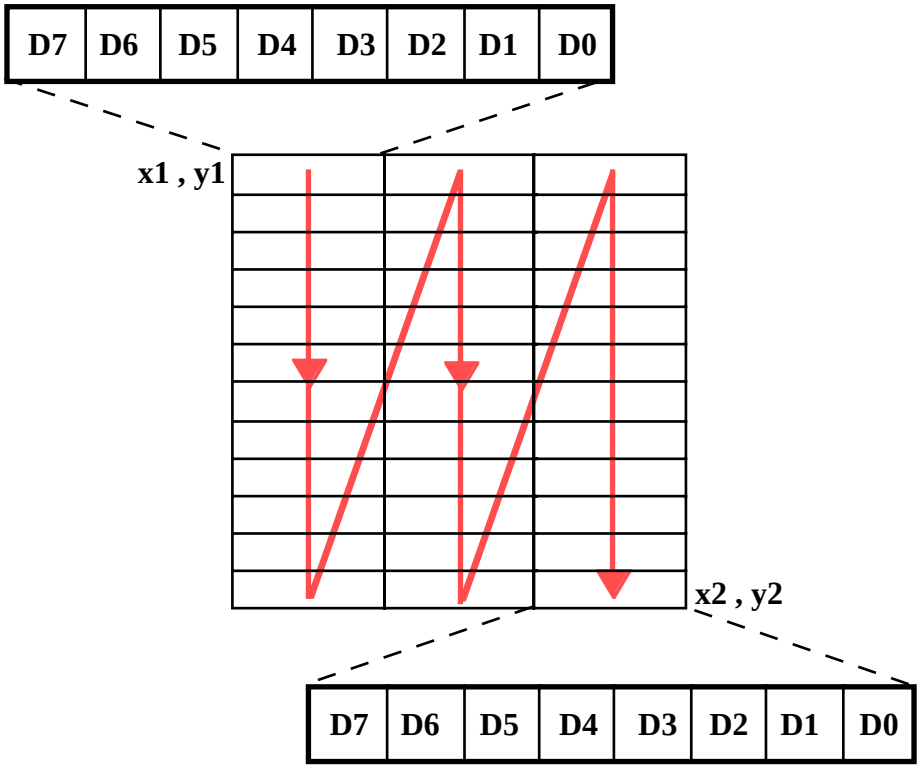


FIGURE 41: HORIZONTAL GRAPHIC DATA AND HORIZONTAL SHIFT

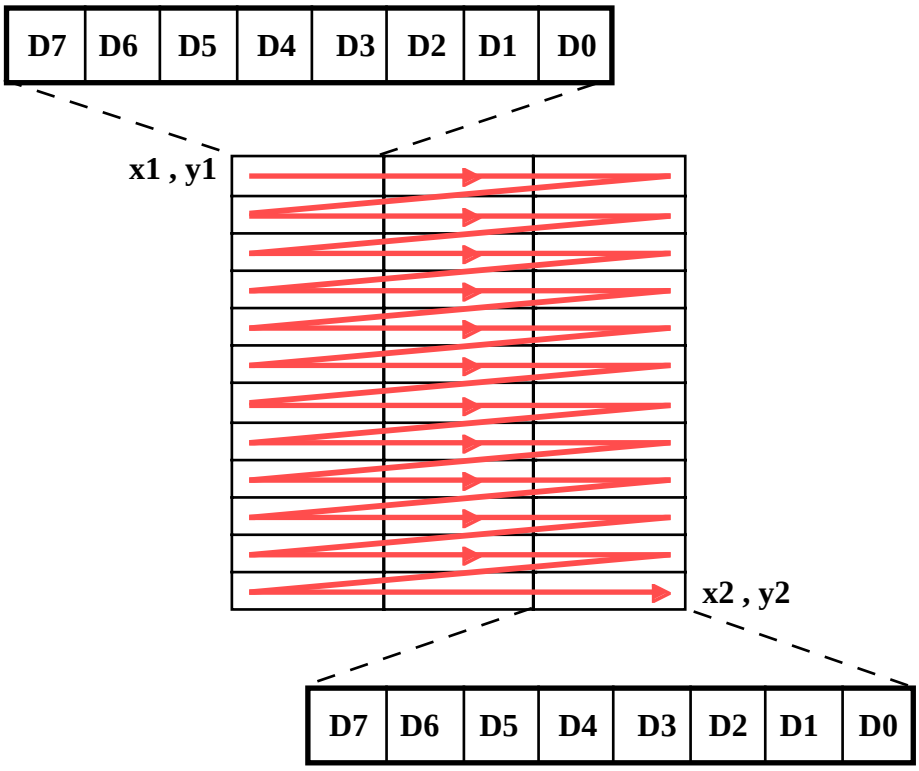


FIGURE 42: HORIZONTAL GRAPHIC DATA AND VERTICAL SHIFT

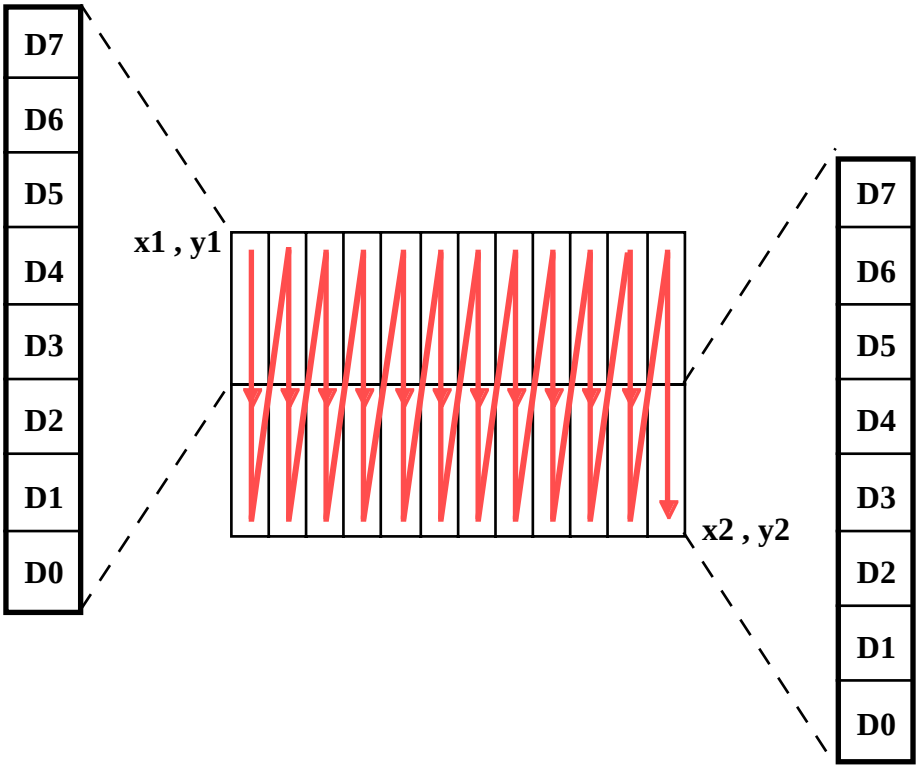


FIGURE 43: VERTICAL GRAPHIC DATA AND HORIZONTAL SHIFT

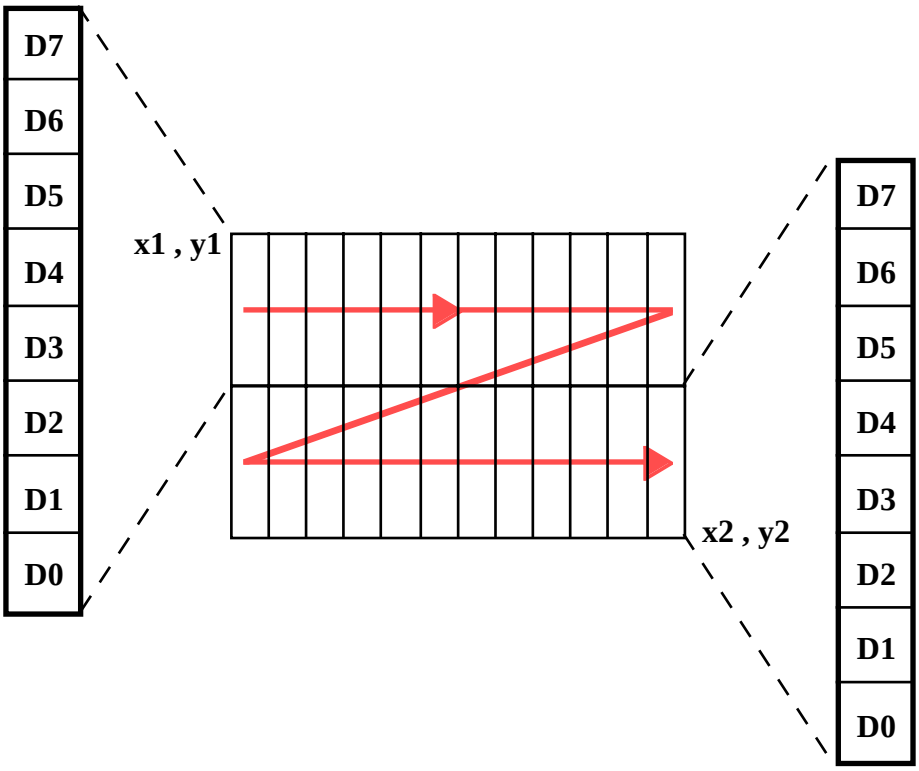


FIGURE 44: VERTICAL GRAPHIC DATA AND VERTICAL SHIFT

GRAPHIC FONT SETTING

Code: ***27 242 font***
Hex code: ***1B F2 font***
Mnemonic: ***ESC ASCII(242) ASCII(font)***

Selects the font used for next alphanumeric characters visualization, in graphic mode. If graphic mode is selected and a drawing area command is not under exectution (as already stated this command uses received bytes as graphic data). then the received bytes are shown on display as characters, anyway. In this last condition a font can be selected that is different from the one used in alphanumeric mode.
The available graphic fonts can be selected with parameter **font**:

font	=	*65*	*(41 Hex)*	*A*	*-> Proportional spacing minifont, 3x5÷5x5 pixels.*
		97	*(61 Hex)*	*a*	*-> Proportional spacing minifont, 3x5÷5x5 pixels.*
		66	*(42 Hex)*	*B*	*-> Katakana font, 5x7 pixels.*
		67	*(43 Hex)*	*C*	*-> Katakana font, 10x14 pixels.*
		98	*(62 Hex)*	*b*	*-> European font, 5x7 pixels.*
		99	*(63 Hex)*	*c*	*-> European font, 10x14 pixels.*
		49	*(31 Hex)*	*1*	*-> 1 pixel interspacing.*
		50	*(32 Hex)*	*2*	*-> 2 pixels interspacing.*

The first six font selection commands are mutually exclusive, while the inerspacing selection commands add their effect to the first ones. So, each of the five fonts can be set with interspacing of 1 or 2 pixels, obtaining 10 different fonts. The font selected is used only in graphic mode, while in alphanumeric mode only the classic font shown in figure B1 with 1 pixel interspacing, is used.
After a power on or initialization, alphanumeric mode is automatically enalbed and the Katakana font, 5x7 , 1 pixel interspacing is automatically selected for graphic mode.
For further information about available characters with described fonts, please refer to APPENDIX B, while figure 31 shows a photo with three of the ten avaialble fonts.

APPENDIX A: COMMANDS SUMMARY TABLES

The following tables list a summary of all the commands recognized by **QTP 12H** firmware. Please remind that these command are compatible with ADDS Viewpoint standard. As in all the other descriptions of the manual, the codes are reported in three formats: decimal, hexadecimal and mnemonic, while the last column reports the number of data returned by command.

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

FIGURE A1: COMMAND CODES SUMMARY TABLE (1 OF 3)

COMMAND	CODE	HEX CODE	MNEMONIC	Ret.
Beep	07	07	BEL	0
Reading of version number	27 86	1B 56	ESC V	3
Operating mode selection	27 65 mode	1B 41 mode	ESC A ASCII(mode)	0
BUZZER activation	27 50 255 attr	1B 32 FF attr	ESC 2 ASCII(255) ASCII(attr)	0
Fluorescent display brightness setting	27 108 lum	1B 6C lum	ESC l ASCII(lum)	0
Request of EEPROM availability	27 51	1B 33	ESC 3	1
Writing presence byte	27 33 78 byte	1B 21 4E byte	ESC ! N ASCII(byte)	0
Reading presence byte	27 33 110	1B 21 6E	ESC ! n	1
Definition of user character	27 66 nchar Pat0÷Pat7	1B 42 nchar Pat0÷Pat7	ESC B ASCII(nchar) ASCII(Pat0)÷ASCII(Pat7)	0
Definition and memorization of user character	27 33 66 nchar Pat0÷Pat7	1B 21 42 nchar Pat0÷Pat7	ESC ! B ASCII(nchar) ASCII(Pat0)÷ASCII(Pat7)	0
Reading of max message number	27 110	1B 6E	ESC n	1
Message storage	27 33 67 mess.n. chr.0÷chr.19	1B 21 43 mess.n. chr.0÷chr.13	ESC ! C ASCII(mess.n.) ASCII(chr.0)÷ASCII(chr.19)	0
Message reading	27 33 69 mess.n.	1B 21 45 mess.n.	ESC ! E ASCII(mess.n.)	0
Visualization of n messaggi	27 33 68 mess.n. n	1B 21 44 mess.n. n	ESC ! D ASCII(mess.n.) ASCII(n)	0
Scrolling message visualization	27 33 83 mess.n. n.chr	1B 21 53 mess.n. n.chr	ESC ! S ASCII(mess.n.) ASCII(n.chr)	0

FIGURE A2: COMMAND CODES SUMMARY TABLE (2 OF 3)

COMMAND	CODE	HEX CODE	MNEMONIC	Ret.
Graphic cursor absolute position	27 206 y x 00	1B CE y x 00	ESC ASCII(206) ASCII(y) ASCII(x) NUL	0
Alphanumeric mode setting	27 208	1B D0	ESC ASCII(208)	0
Graphic mode setting	27 209	1B D1	ESC ASCII(209)	0
Graphic area setting	27 241 x1 y1 x2 y2 cmd	1B F1 x1 y1 x2 y2 cmd	ESC ASCII(241) ASCII(x1) ASCII(y1) ASCII(x2) ASCII(y2) ASCII(cmd)	0
Graphic font setting	27 242 font	1B F2 font	ESC ASCII(242) ASCII(font)	0

FIGURE A3: COMMAND CODES SUMMARY TABLE (3 OF 3)

APPENDIX B: DISPLAY CHARACTERS

The following tables shows the characters sets displayed on **QTP 12H** for all the possible received characters, according with ordered display and model and according with functionality mode preselected through proper commands.
Even the not ASCII characters (or special characters) change when the display type changes and if the user requires a characters set different from those described in the following figures, he can directly contact **grifo®**.

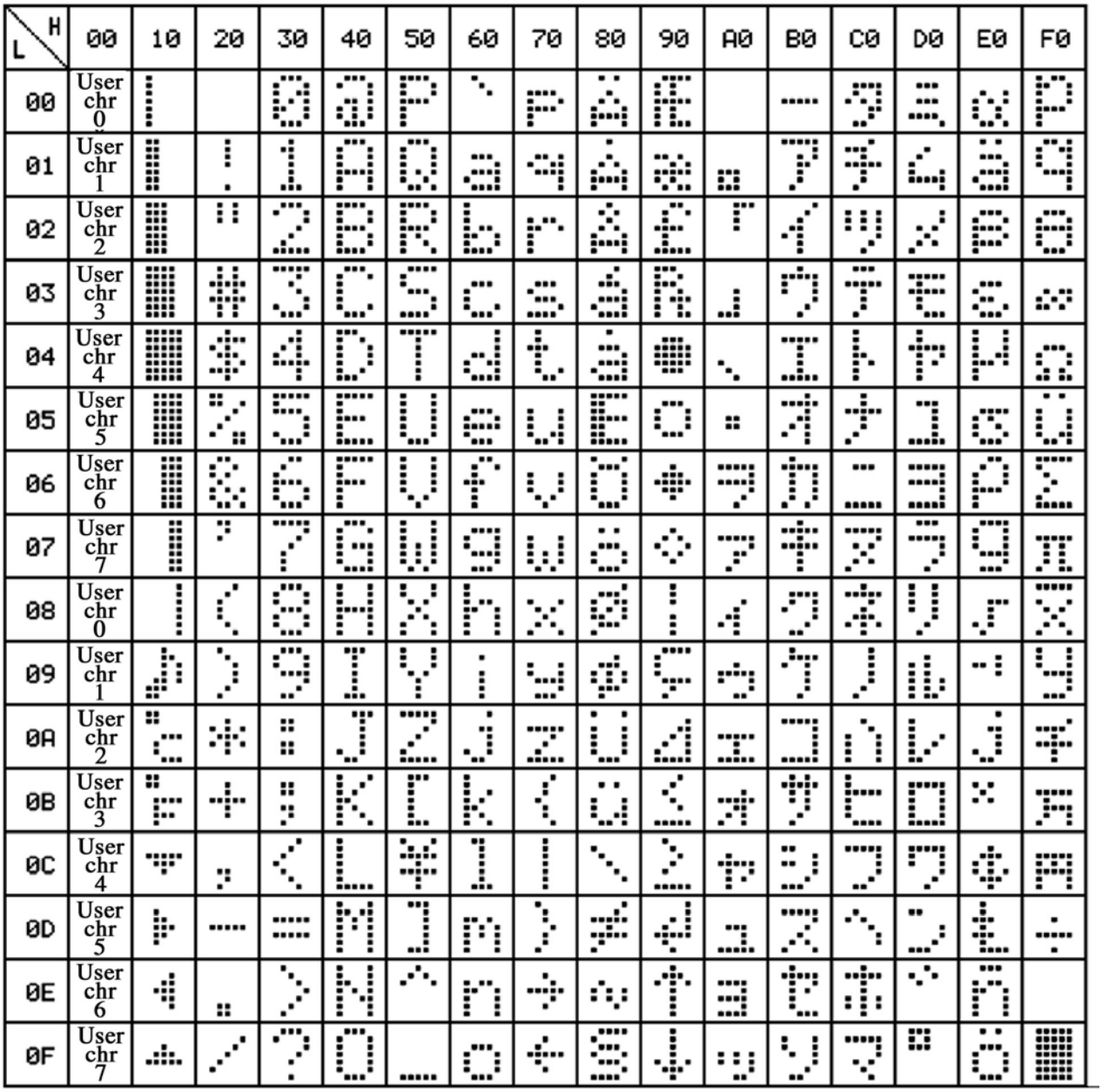


FIGURE B1: CHARACTERS TABLE OF QTP 12H-F2, GF2 IN ALPHANUMERIC MODE

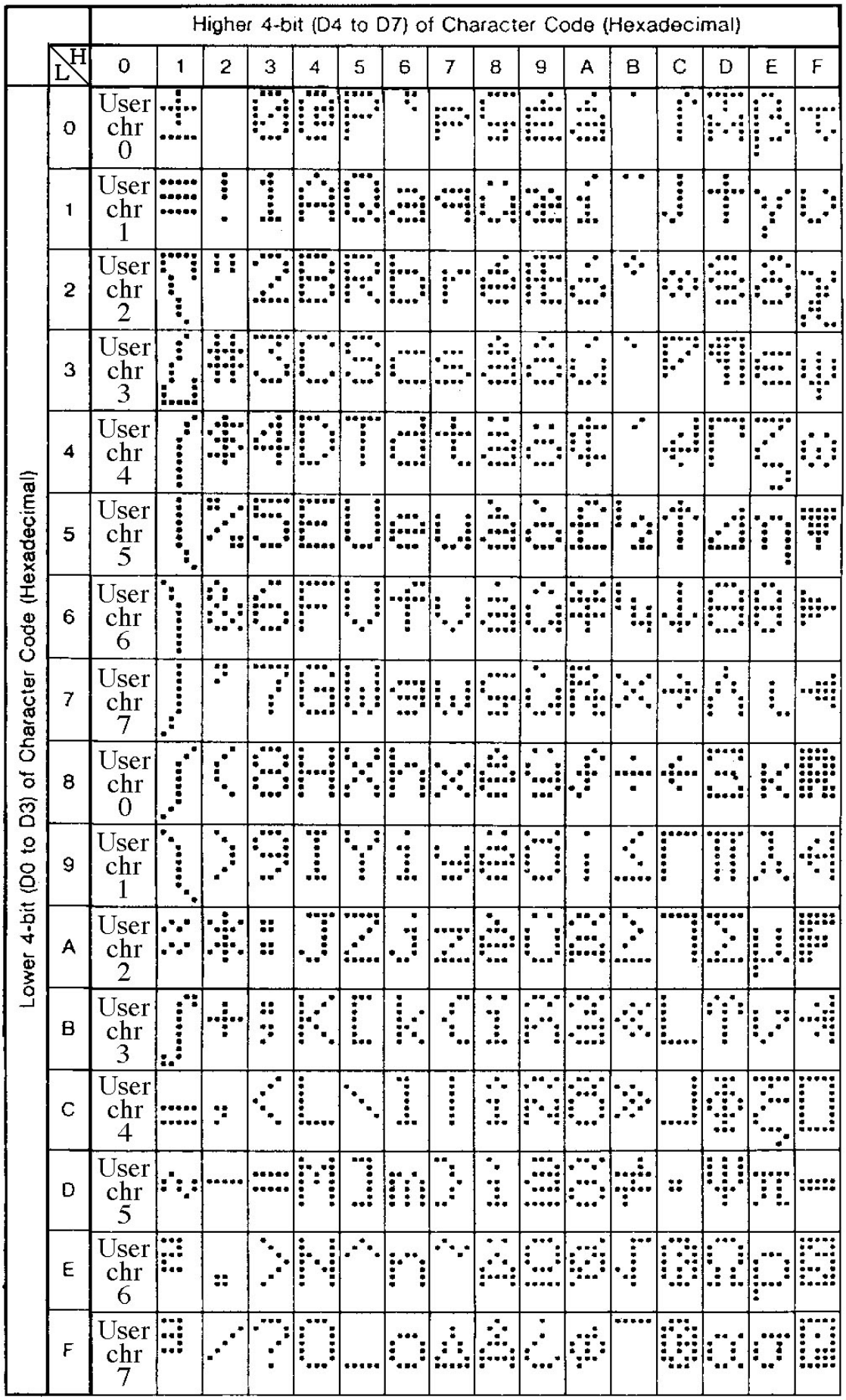


FIGURE B2: CHARACTERS TABLE OF QTP 12H-C2

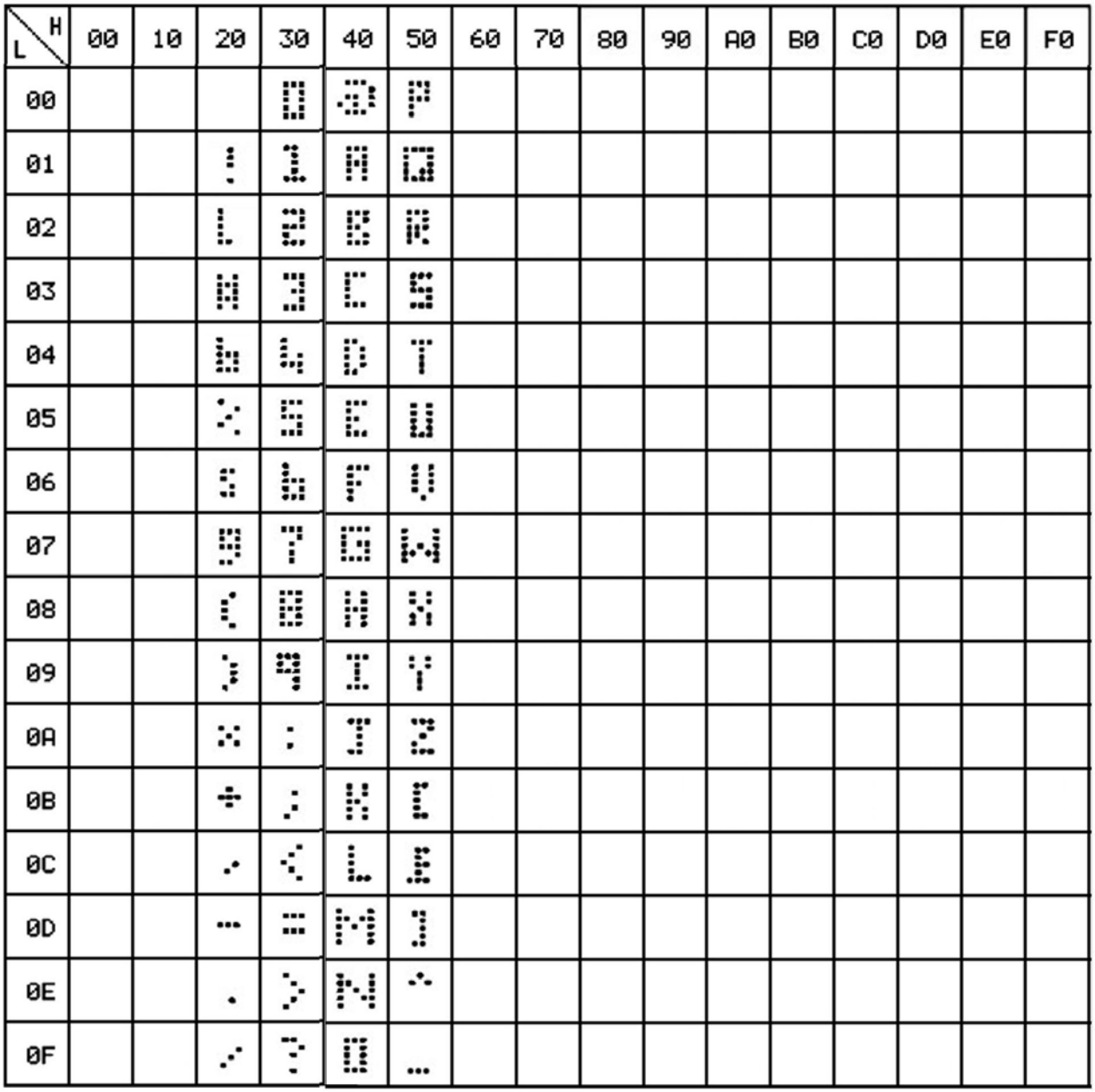


FIGURE B3: CHARACTERS TABLE OF QTP 12H-GF2, IN GRAPHIC MODE WITH MINIFONT

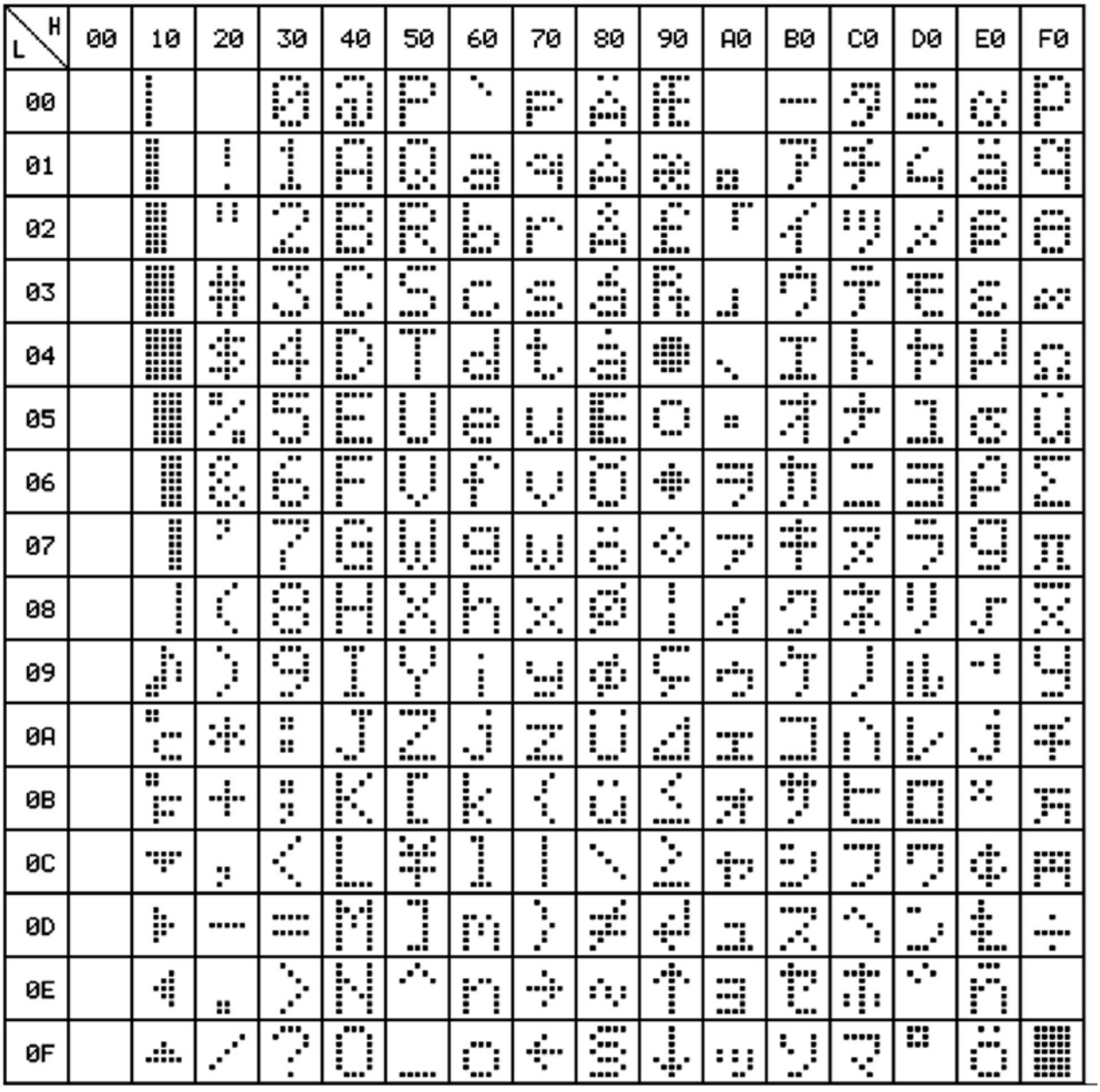


FIGURE B4: CHARACTERS TABLE OF QTP 12H-GF2, IN GRAPHIC MODE WITH KATAKANA FONT

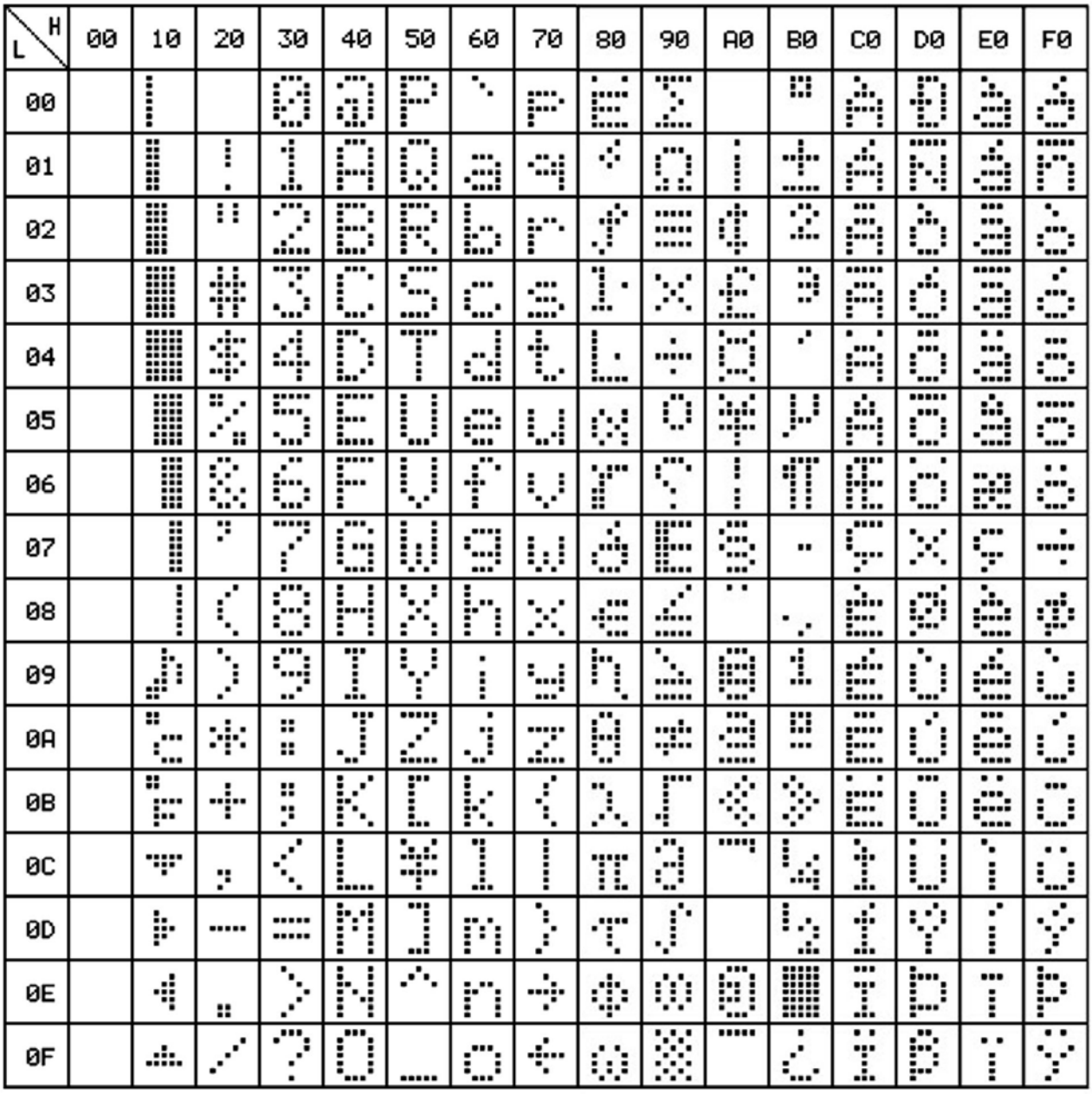


FIGURE B5: CHARACTERS TABLE OF QTP 12H-GF2, IN GRAPHIC MODE WITH EUROPEAN FONT

APPENDIX C: MOUNTING NOTES

QTP 12H is provided complete of metallic container and some accessories that simplify the mounting. Inside this appendix are described all the information concerning this operations, together with the instructions to open the container and to personalize it.

TERMINAL DIMENSIONS

In the following figure there are dimensions of terminal **QTP 12H** external metallic container and attached frontal plastic frame. Dimensions are in **mm** and the drawing scale is 1:1.

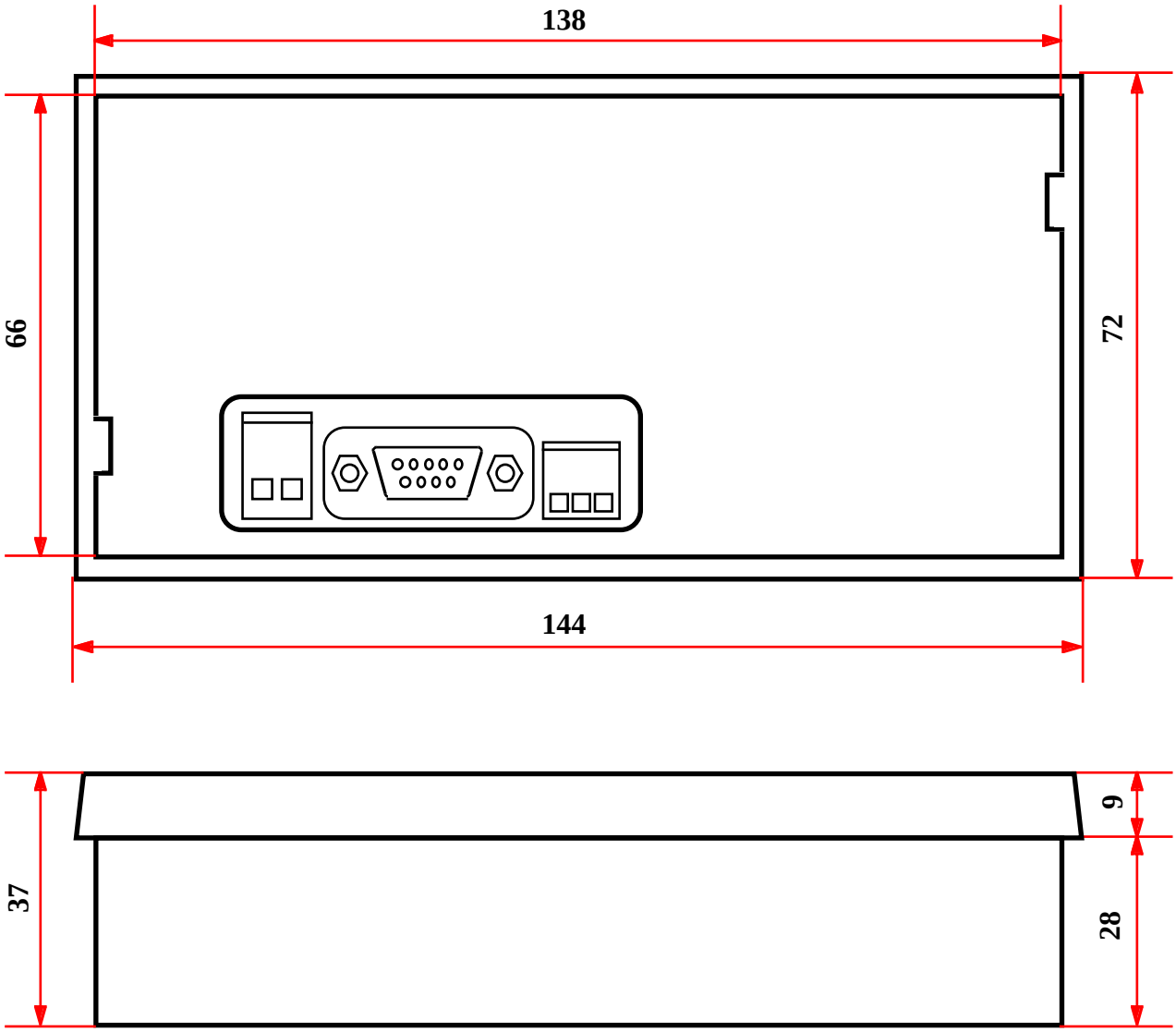


FIGURE C1: QTP 12H DIMENSIONS

The dimensions of previous figure refer to container only, but occupied area can be slightly greater by considering also mounting clamps and screws, up to a maximum of 156 x 72 x 80 mm (W x H x D).

FRONT PANEL MOUNTING

The provided mounting mode is the front panel one that is possible on any panel with 10 mm maximum thickness and fixing is done by two clamps provided with **QTP 12H**.
Installation operations are exteremly easy and they are below summarized:

1) make a rectangular breaking on mounting panel like those in the following figure;

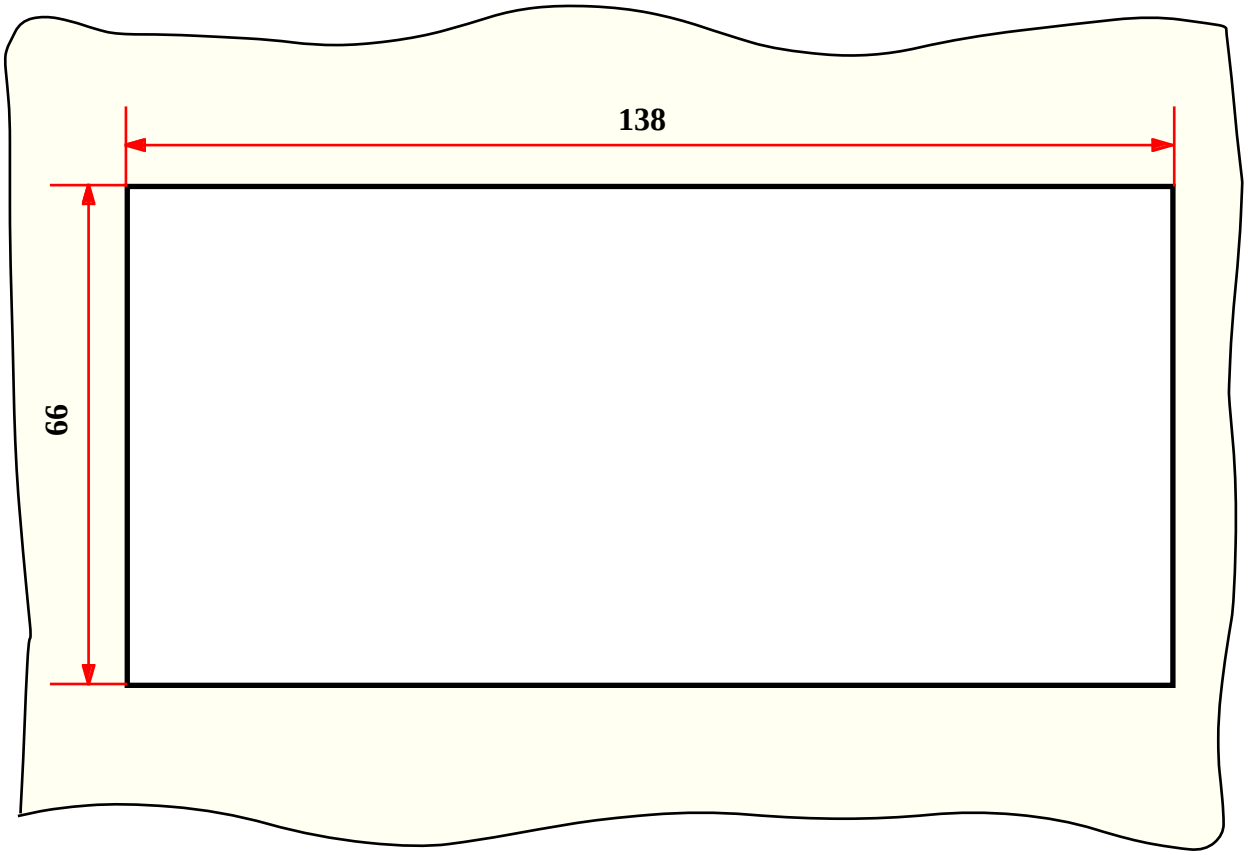


FIGURE C2: BREAKING FOR INSTALLATION

2) screw the two C clamps, keeping the sharpened part close to the screw-cut edge of clamp;

3) insert **QTP 12H** in the breaking made at point 1;

4) dock the two clamps prepared at point 2 to the specific side breakings of **QTP 12H** container, taking care that the first hook of the clamp, the one near screw-cut edge, enters correctly in the proper lateral hole of the containter;

5) screw the screws of the two clamps until the **QTP 12H** container is firmly docked to mounting panel;

6) connect the connettors on the back side.

PERSONALIZATION LABEL INSERTION

Frontal of **QTP 12H** is provided with a pocket where the user can insert a personalization label with his logo, an identification code, the terminal function, or anything else.
If the label is required please inser it before mounting **QTP**. Label must be thin but rather rigid, for example made of 160 g/m^2 paper or polyester or polycarbonate sheets. Here follow the suggested dimensions, in millimeters, of personalization label; please nothe that the white zone is the area contained in the transparent window, or in other words, the visible part:

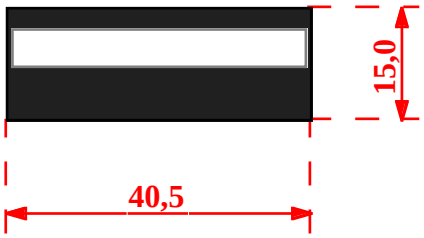


FIGURE C3: DIMENSIONS OF PERSONALIZATION LABEL

Here follow the operations required to insert personalization label inside the **QTP 12H**.

1) Unscrew the two black screws of frontal panel (if present).
2) Separate the group metallic carter + plastic frame from the group front panel + printed circuit. A simple pressure on **QTP 12H** connectors, or on the printed circuit always from the backside connectors window, is sufficient.
3) Remove the four screw nuts that couple the front panel to the printed circuit, and separate the two parts.
3) Now the front panel is ready to insert the personalization label; this latter must be inserted from the bottom side, using the specific pocket located on the back of front panel, as shown in following figure. As described on figure C3, lenght of label must be greater than height of window to simplify the insertion and extraction.
4) Remount terminal **QTP 12H**, following the previous steps in reversed order.

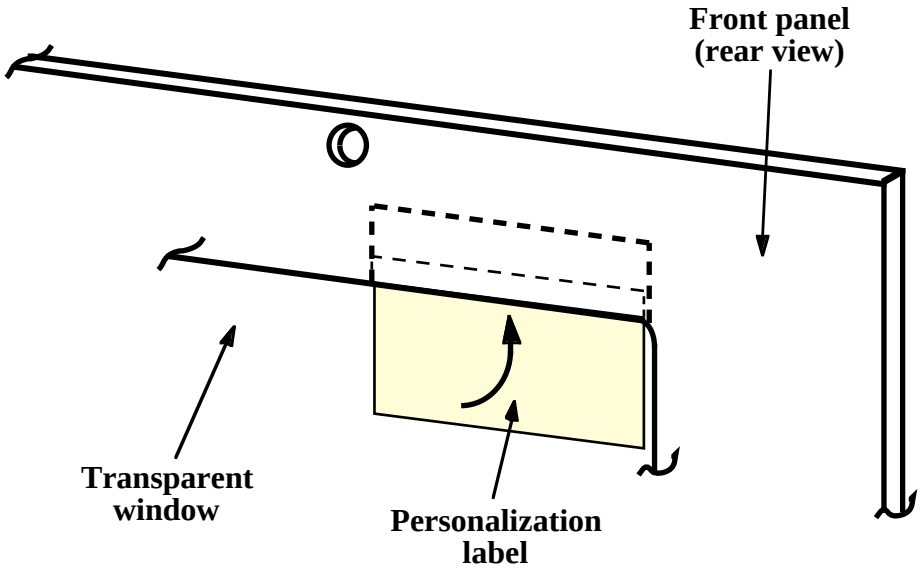


FIGURE C4: PERSONALIZATION LABEL INSERTION

FIXING FRONT PANEL TO CONTAINER

QTP 12H by default is provided with front panel (polyester panel+printed circuit board) jointed in plastic frame of the back metallic container. Terminal anyway allows a better mechanical docking between this two groups, by using two specific screws, that avoid accidental separations of front panel.

Here follows the operations that must be performed, to ensure such docking:

1) Separate the group metallic carter + plastic frame from the group front panel + printed circuit. A simple pressure on backside **QTP 12H** connectors, or on the printed circuit always from the backside connectors window, is normally sufficient.

2) Of the six screws, that dock the plastic frame to the back metallic container, unscrew the two central ones.

3) On front panel, in correspondence with these central screws, there are two holes provided with flare, visible only from the back side. It is sufficient to perforate the polyester layer that covers the frontal, to make these holes accessible.

4) Remount everything, using the same screws removed at point **2**, that will be screwed on the front panel and not on the plastic frame any more.

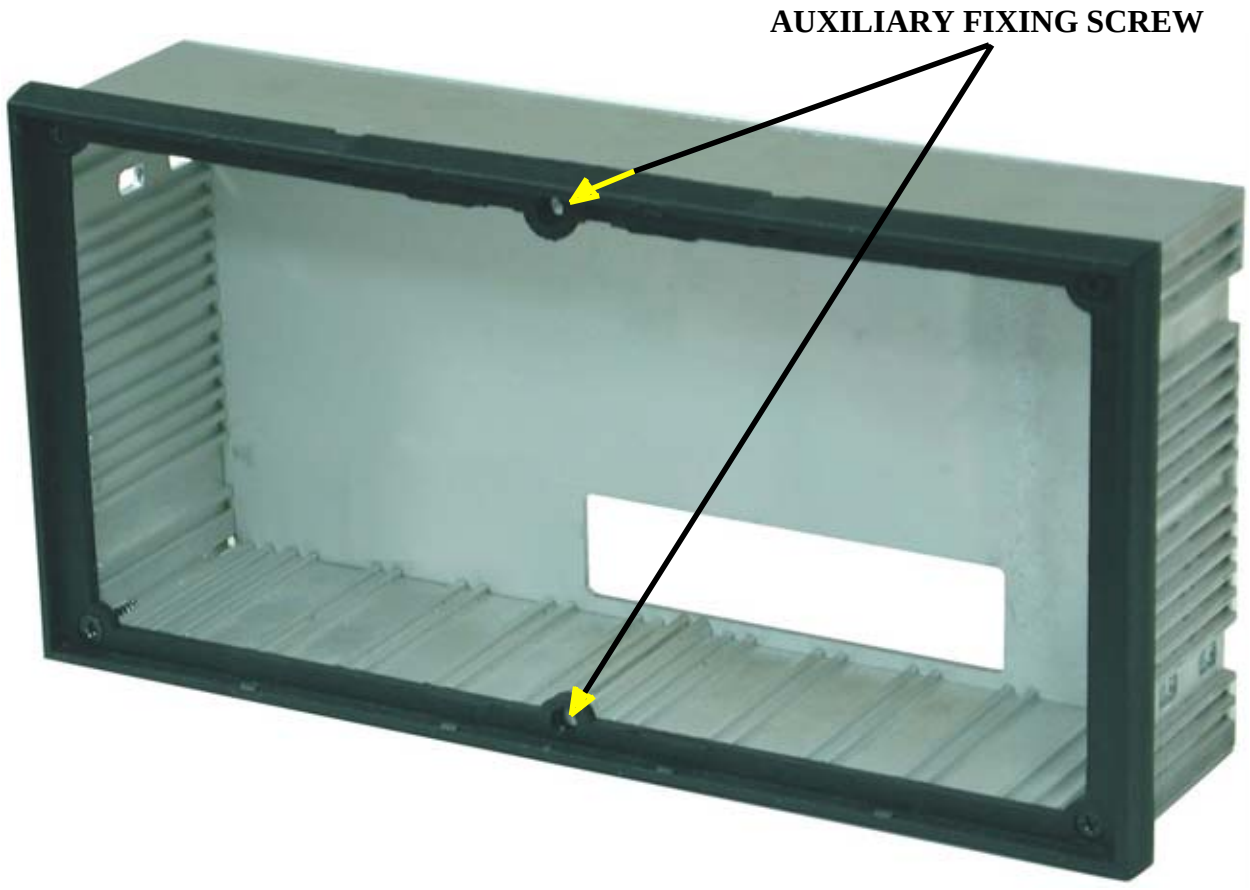


FIGURE C5: SCREW FOR FRONT PANEL FIXING

APPENDIX D: ALPHABETICAL INDEX

D

P

Q

R

S

T

U

V

W