Size: front 72x144 mm; depth of front 9 mm; depth of back 27 mm. Remarkably Low Price. Alluminium container with frontal frame in plastic, provided with mounting clamps. Front panel with mask and keyboard in polycarbonate 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. Membrane keyboard with 12 keys and double serigraph: numbers and functions. Autorepeat and keyclick features. Personalization of device name and/or signalation LED by inserting a label. Software driven signalation LED in several modalities. Buzzer for BELL signalations, key click and software driven acoustic signalations. EEPROM for settings, messages, key codes, etc. Memorization and representation of up to 97 messages in EEPROM, also Scrolling. 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 according to configuration used ranges from 1.2 W and 2.3 W. On board logic protection through TransZorb™.

It is possible to require custom panels and programs.
IMPORTANT

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

SYMBOLS DESCRIPTION

In the manual could appear the following symbols:

- Attention: Generic danger
- Attention: High voltage

Trade Marks

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

INTRODUCTION ................................................................................................................... 1

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

GENERAL INFORMATION ................................................................................................. 2
  KEYBOARD ................................................................................................................ 4
  SERIAL COMMUNICATION ...................................................................................... 5
  CAN INTERFACE ....................................................................................................... 5
  INDICATOR LEDS ...................................................................................................... 5
  ON BOARD POWER SUPPLY .................................................................................. 5
  BUZZER .................................................................................................................. 6
  DISPLAY ................................................................................................................ 6
  EEPROM ............................................................................................................... 6

TECHNICAL FEATURES .................................................................................................... 8
  GENERAL FEATURES ............................................................................................ 8
  PHYSICAL FEATURES .......................................................................................... 8
  ELECTRIC FEATURES .......................................................................................... 9

INSTALLATION .................................................................................................................. 10
  CONNECTIONS ....................................................................................................... 10
    CN1 - POWER SUPPLY CONNECTOR .................................................................. 10
    CN2 - SERIAL LINE CONNECTOR ....................................................................... 12
    CN3 - CAN INTERFACE CONNECTOR ................................................................ 18
  CONTRAST REGULATION TRIMMER ...................................................................... 20
  JUMPERS .................................................................................................................. 20
  SERIAL COMMUNICATION SELECTION ............................................................. 22
  CAN INTERFACE .................................................................................................... 24
  ALIMENTAZIONE ................................................................................................... 26

SOFTWARE DESCRIPTION ............................................................................................... 27
  LOCAL SET UP ........................................................................................................ 27
  RECEIVE BUFFER ................................................................................................... 28
  KEYBOARD ACQUISITION ...................................................................................... 28
    KEYS CODES ...................................................................................................... 29
  COMMUNICATION MODALITIES ........................................................................... 30
    MASTER SLAVE COMMUNICATION MODE ..................................................... 30
  DATA STORED IN EEPROM .................................................................................. 32
  CHARACTERS VISUALIZATION ON THE DISPLAY .................................................. 33
  COMMANDS FOR CURSOR POSITION ................................................................. 34
    CURSOR LEFT ................................................................................................... 34
    CURSOR RIGHT ................................................................................................. 34
    CURSOR DOWN ................................................................................................. 34
    CURSOR UP ...................................................................................................... 34
    HOME ............................................................................................................... 34
    CARRIAGE RETURN ....................................................................................... 35
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>CARRIAGE RETURN+LINE FEED</td>
<td>35</td>
</tr>
<tr>
<td>ABSOLUTE CURSOR PLACEMENT</td>
<td>35</td>
</tr>
<tr>
<td>COMMANDS FOR CHARACTERS ERASURE</td>
<td>36</td>
</tr>
<tr>
<td>BACKSPACE</td>
<td>36</td>
</tr>
<tr>
<td>CLEAR PAGE</td>
<td>36</td>
</tr>
<tr>
<td>CLEAR LINE</td>
<td>36</td>
</tr>
<tr>
<td>CLEAR END OF LINE</td>
<td>36</td>
</tr>
<tr>
<td>CLEAR END OF PAGE</td>
<td>36</td>
</tr>
<tr>
<td>COMMANDS FOR CURSOR ATTRIBUTES MANAGEMENT</td>
<td>37</td>
</tr>
<tr>
<td>CURSOR OFF</td>
<td>37</td>
</tr>
<tr>
<td>STEADY STATIC CURSOR ON</td>
<td>37</td>
</tr>
<tr>
<td>BLINKING BLOCK CURSOR ON</td>
<td>37</td>
</tr>
<tr>
<td>COMMANDS FOR EEPROM</td>
<td>38</td>
</tr>
<tr>
<td>REQUEST FOR EEPROM WRITING POSSIBILITY</td>
<td>38</td>
</tr>
<tr>
<td>WRITING OF PRESENCE BYTE</td>
<td>38</td>
</tr>
<tr>
<td>READING OF PRESENCE BYTE</td>
<td>38</td>
</tr>
<tr>
<td>COMMANDS FOR GENERAL FUNCTIONS</td>
<td>39</td>
</tr>
<tr>
<td>READING OF VERSION NUMBER</td>
<td>39</td>
</tr>
<tr>
<td>BEEP</td>
<td>39</td>
</tr>
<tr>
<td>OPERATING MODE SELECTION</td>
<td>39</td>
</tr>
<tr>
<td>FLUORESCENT DISPLAY BACKLIGHT SETTING</td>
<td>40</td>
</tr>
<tr>
<td>LED AND BUZZER ACTIVATION</td>
<td>40</td>
</tr>
<tr>
<td>COMMANDS FOR MESSAGE MANAGEMENT</td>
<td>41</td>
</tr>
<tr>
<td>READING OF THE LAST STORAGED MESSAGE NUMBER</td>
<td>41</td>
</tr>
<tr>
<td>MESSAGE STORAGE</td>
<td>41</td>
</tr>
<tr>
<td>MESSAGE READING</td>
<td>42</td>
</tr>
<tr>
<td>MESSAGE VISUALIZATION</td>
<td>42</td>
</tr>
<tr>
<td>SCROLLING MESSAGES VISUALIZATION</td>
<td>43</td>
</tr>
<tr>
<td>COMMANDS FOR KEYBOARD MANAGEMENT</td>
<td>44</td>
</tr>
<tr>
<td>KEY RECONFIGURATION</td>
<td>44</td>
</tr>
<tr>
<td>KEYCLICK ON WITHOUT MEMORIZATION</td>
<td>44</td>
</tr>
<tr>
<td>KEYCLICK OFF WITHOUT MEMORIZATION</td>
<td>44</td>
</tr>
<tr>
<td>KEYCLICK ON WITH MEMORIZATION</td>
<td>45</td>
</tr>
<tr>
<td>KEYCLICK OFF WITH MEMORIZATION</td>
<td>45</td>
</tr>
<tr>
<td>COMMANDS FOR USER CHARACTERS</td>
<td>46</td>
</tr>
<tr>
<td>DEFINITION OF USER CHARACTER</td>
<td>46</td>
</tr>
<tr>
<td>DEFINITION AND MEMORIZATION OF USER CHARACTER</td>
<td>47</td>
</tr>
<tr>
<td>GRAPHIC COMMANDS</td>
<td>48</td>
</tr>
<tr>
<td>ALPHANUMERIC MODE SETTING</td>
<td>48</td>
</tr>
<tr>
<td>GRAPHIC MODE SETTING</td>
<td>50</td>
</tr>
<tr>
<td>GRAPHIC CURSOR ABSOLUTE POSITION</td>
<td>50</td>
</tr>
<tr>
<td>GRAPHIC AREA SETTING</td>
<td>50</td>
</tr>
<tr>
<td>GRAPHIC FONT SETTING</td>
<td>54</td>
</tr>
<tr>
<td>EXTERNAL CARDS</td>
<td>41</td>
</tr>
<tr>
<td>APPENDIX A: COMMAND CODES SUMMARY TABLES</td>
<td>A-1</td>
</tr>
<tr>
<td>APPENDIX B: DISPLAY CHARACTERS</td>
<td>B-1</td>
</tr>
</tbody>
</table>
APPENDIX C: MOUNTING OUTLINE DIMENSIONS .................................................... C-1
TERMINAL DIMENSIONS ........................................................................................................... C-1
SURFACE OR FLUSH PANEL MOUNTING ............................................................................ C-2
CUSTOMIZATION LABEL INSERTION ............................................................................. C-3
CUSTOMIZATION LABEL DIMENSIONS ............................................................................. C-4

APPENDIX D: ALPHABETICAL INDEX ................................................................................. D-1
FIGURES INDEX

FIGURE 1: PHOTO OF THE SEVERAL MODELS AVAILABLE ................................................................. 3
FIGURE 2: QTP 12 PHOTO IN ALL AVAILABLE MODELS ............................................................... 7
FIGURE 3: CONSUMPTIONS TABLE .............................................................................................. 9
FIGURE 4: CN1 - POWER SUPPLY CONNECTOR ........................................................................ 10
FIGURE 5: AC POWER SUPPLY 8+24 Vac .............................................................................. 10
FIGURE 6: DC POWER SUPPLY +10+40 Vdc ......................................................................... 11
FIGURE 7: STABILIZED POWER SUPPLY +5 Vdc ................................................................. 11
FIGURE 8: CN2 - SERIAL LINE CONNECTOR ............................................................... 12
FIGURE 9: RS 232 point-to-point connection example ............................................................. 13
FIGURE 10: RS 422 point-to-point connection example ............................................................ 13
FIGURE 11: RS 485 point-to-point connection example ............................................................ 13
FIGURE 12: RS 485 network connection example .................................................................... 14
FIGURE 13: CURRENT LOOP 4 WIRES POINT-TO-POINT CONNECTION EXAMPLE ......... 15
FIGURE 14: CURRENT LOOP 2 WIRES POINT-TO-POINT CONNECTION EXAMPLE .............. 15
FIGURE 15: CURRENT LOOP NETWORK CONNECTION EXAMPLE ....................................... 16
FIGURE 16: REAR VIEW ........................................................................................................... 17
FIGURE 17: JUMPERS, CONNECTORS, TRIMMER, ETC LOCATION ......................................... 17
FIGURE 18: CN3 - CAN INTERFACE CONNECTOR ............................................................. 18
FIGURE 19: CAN LINE CONNECTION .................................................................................. 18
FIGURE 20: CAN INTERFACE CONNECTION EXAMPLE .................................................... 19
FIGURE 21: JUMPERS TABLE ............................................................................................... 20
FIGURE 22: COMPONENTS MAP SOLDER SIDE ............................................................... 21
FIGURE 23: COMPONENTS MAP COMPONENTS SIDE ..................................................... 21
FIGURE 24: DRIVER FOR SERIAL COMMUNICATION SELECTION ...................................... 23
FIGURE 25: PHOTO OF QTP 12-C2 .................................................................................. 25
FIGURE 26: PHOTO OF QTP 12-F2 .................................................................................. 25
FIGURE 27: PHOTO OF QTP 12-GF2 .................................................................................. 25
FIGURE 28: KEYS ENUMERATION AND LOCATION .......................................................... 29
FIGURE 29: DEFAULT KEY CODES .................................................................................. 29
FIGURE 30: EXAMPLE OF MASTER SLAVE COMMUNICATION .............................................. 31
FIGURE 31: PHOTO OF CHARACTERS AVAILABLE ON QTP 12-GF2 .................................. 33
FIGURE 32: PHOTO OF FRONTAL AND KEYBOARD .......................................................... 45
FIGURE 33: USER CHARACTERS MODEL ........................................................................... 46
FIGURE 34: COORDINATES OF GRAPHIC DISPLAY PIXELS ................................................. 48
FIGURE 35: FIRST GRAPHIC EXAMPLE .............................................................................. 49
FIGURE 36: SECOND GRAPHIC EXAMPLE ............................................................................. 49
FIGURE 37: EXAMPLE OF GRAPHIC DRAWING .................................................................... 51
FIGURE 38: HORIZONTAL DATA AND HORIZONTAL SHIFT .............................................. 52
FIGURE 39: HORIZONTAL DATA AND VERTICAL SHIFT ................................................... 52
FIGURE 40: VERTICAL DATA AND HORIZONTAL SHIFT ..................................................... 53
FIGURE 41: VERTICAL DATA AND VERTICAL SHIFT ............................................................ 53
FIGURE 42: AVAILABLE CONNECTIONS DIAGRAM ......................................................... 57
FIGURE A1: COMMAND CODES SUMMARY TABLE (1 OF 3) .................................................. A-1
FIGURE A2: COMMAND CODES SUMMARY TABLE (2 OF 3) ................................................ A-2
FIGURE A3: COMMAND CODES SUMMARY TABLE (3 OF 3) ................................................ A-2
FIGURE B1: QTP 12-F2, GF2 IN ALPHANUMERIC MODE CHARACTERS TABLE ................ B-1
FIGURE B2: QTP 12-C2 CHARACTERS TABLE ............................................................... B-2
FIGURE B3: QTP 12-GF2 IN GRAPHIC MODE WITH MINIFONT CHARACTERS TABLE .............. B-1
FIGURE B4: QTP 12-GF2 IN GRAPHIC MODE WITH KATAKANA FONT CHARACTERS TABLE .......... B-1
FIGURE B5: QTP 12-GF2 IN GRAPHIC MODE WITH EUROPEAN FONT CHARACTERS TABLE .......... B-1
FIGURE C1: DIMENSIONS OF QTP 12 .............................................................................. C-1
FIGURE C2: BREAKING FOR MOUNTING ............................................................................. C-2
FIGURE C3: CUSTOMIZATION LABEL DIMENSIONS ............................................................... C-3
FIGURE C4: CUSTOMIZATION LABEL INSERTION ................................................................. C-4
INTRODUCTION

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

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

The reported data are destined- IN EXCLUSIVE WAY- to specialized users, that can interact with the devices in safety conditions for the persons, for the machine and for the 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 respect the actual safety normative. The protection of this container is not threshold to the only atmospheric agents, but specially to mechanic, electric, magnetic, etc. ones.

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

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

FIRMWARE VERSION

This handbook make reference to printed circuit version 101202 and to firmware version 1.3 and following ones. The validity of the information contained in this manual is subordinated to the firmware release number, so the user must always verify the correct correspondence between the notations. Inside the device, the firmware release number is written on the label stuck on the CPU or it can be obtained by a proper command sent through the serial line. Hardware version is printed in several places, for example on the top of center of printed circuit.
QTP 12 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, command, visualization operations which could be necessary in many civil and/or industrial applications.

QTP 12 is available with alphanumeric Fluorescent or LEDs back lite LCD displays, with 20 characters for 2 lines or with Fluorescent graphic display 140x16 pixels.

QTP 12 is directly connected with the display and the 12 keys membrane keyboard embedded in the front panel, in addition there are a pocket to inset a label for customization and a signalation LED. A practical and resistant 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 12 allow to mount/unmount it by a simple rectangular break in the support rack. Also a table, with no further element, can be used.

QTP 12 is the best choice whenever the user needs to show messages and 12 keys are enough to interact with the hardware.

QTP 12 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. This way, the amount of work for the main CPU is lowered, also the messages to show must not be sent through the serial line to the panel. Showing messages in sliding mode allows to represent up to 200 characters on the first row of the display.

It is also possible to read, through the serial line, the messages stored in the EEPROM. This allows the user to employ QTP 12 as a small storage device where special data like hardware Set-Up, passwords, ID codes, etc. can be stored.

QTP 12 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 units.

Also, interconnection is warranted by an optional CAN serial line, that increases the utilisation ranges and impovers overall network performances.

The QTP 12 is able to execute an entire range of display commands, including Clear Screen, Position cursor, EEPROM reading or writing, etc., with code compatibility to ADDS ViewPoint standard video terminal.

Features of QTP 12, including options, are as follows:

- size: front 72x144 mm; depth of fron 9 mm; depth of back 27 mm.
- remarkably low price.
- alluminium container with frontal frame in plastic, provided with mounting clamps.
- front panel with mask and keyboard in polycarbonate 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.: 

- QTP 12-C2: alphanumeric LCD backlit, 2 rows by 20 characters
- QTP 12-F2: alphanumeric Fluorescent, 2 rows by 20 characters
- QTP 12-GF2: graphic Fluorescent, 140 by 16 pixels
**Figure 1: Photo of the several models available**
- maximum size of visible area: 82x18 mm.
- membrane keyboard with 12 keys and double serigraph: numbers and functions.
- autorepeat and keyclick features.
- personalization of device name and/or signalation LED by inserting a label.
- software driven signalation LED in several modalities.
- buzzer for BELL signalations, key click and software driven acoustic signalations.
- EEPROM for settings, messages, key codes, etc.
- memorization and representation of up to 97 messages in EEPROM, also scrolling.
- 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 according to configuration used ranges from 1.2 W and 2.3 W.
- on board logic protection through TransZorb™.
- it is possible to require custom panels and programs.

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

**KEYBOARD**

**QTP 12** has a membrane keyboard with 12 keys located around the display.
All the keys are metallic dome type so they provide a tactile sensation of the key pressed and they withstand the knocks and bumps of industrial life.
All the keys have a standard label (see figure 32) that satisfy the normal man-machine interface requirements.
Remarkable is the presence of numeric figures, th whole alphabet and some functions that allow to input any kind of data and to execute any kind of command.
These keys are equipped with autorepeat and they are totally software reconfigurable or on the other hand the code sent in RS 232 when a key is pressed can be changed or disabled.
It is also possible to switch on/off the keyclick function, i.e the buzzer short activation each time a key is pressed.
Please remark that the four central keys may assume a variable functionality, in case of graphic display installed. This allows to draw on screen their function names and change them according to needs of the application (e. g. START, STOP, INS, DEL, ALRM, INFO, etc.).
Four keys are used for local setup of some operating parameters as described in paragraph “LOCAL SET-UP”.
In addition, a label can be inserted in the frontal to customize the terminal as described in APPENDIX C.
SERIAL COMMUNICATION

QTP 12 has one serial communication line. By default shape it is configured as RS 232, but using a proper indication in the ordering phase, it can be configured in:

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

The physical protocol of both the serial lines is completely configurable through a dedicated set up modality that let the user select the values listed in "TECHNICAL SPECIFICATIONS" chapter, by the simple keyboard use.

Logic protocol can be point-to-point or master-slave, using the nineth bit technique; this latter allows QTP 12 to be connected in a network and to communicate with terminals of the same type or different type, easily and efficiently.

CAN INTERFACE

QTP 12 can have, as option, of a complete CAN interface that supports the BasicCAN and PeliCAN 2.0B standards protocol. 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

INDICATOR LEDS

QTP 12 has 1 indicator LEDs for different visual signals, that can be enabled, disabled or enabled with blinking attribute, through comfortable serial commands. This functionality is totally autonomous and it doesn't need any interventation on user side.

The main purpose of LED is to show a visual indication about the card's status, making so easier debug and verify operations even from a long distance from the panel, where the display is not readable.

For this reason there are three different status (off, on and blinking), to divide different operative situations as, for example, an allarm, an alert, a ready to work, etc.

ON BOARD POWER SUPPLY

One of the most important peculiarity of QTP 12 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 12 without power supply can be ordered using the code .5Vdc or .ALIM, in this case +5 Vdc stabilized power supply must be provided by an external source.
BUZZER

QTP 12 has a circuitry that generates a steady sound, based on a capacitive buzzer. This circuitry can be activated by software through a specific command for generating a simple beep or it can be linked to a key pressure, just to get the keyclick function, or it can signalize possible malfunctions. In latter case, after a power on, the card generates an intermittent sound and doesn't work correctly, there is a wrong condition that must be resolved: please contact grifo® technicians.

DISPLAY

QTP 12 is available with Fluorescent or backlit LCD alphanumeric displays 20x2 characters or graphic fluorescent 140x16 pixels. LEDs backlighting of LCD models 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 12 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 commands, QTP 12 with graphic display (QTP 12-GF2) can execute all kinds of commands (graphic and alphanumeric), while QTP 12 with alphanumeric display (QTP 12-C2 and QTP 12-F2), of course, cannot execute graphic commands. The user must choose the right display (so the right QTP 12 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 with LCD display not backlit: for detailed information about these options and their availability, please contact directly grifo® offices.

EEPROM

QTP 12 has on board EEPROM (size 2 KBytes) for storing set up, communication protocol, keys codes, messages, and so on. Many of the stored data have vital importance so a serial EEPROM has been chosen to obtain the best warranties on validity and maintenance of the saved information, even when power supply is not available. It is possible to memorize up to 97 messages of 20 characters that can be first saved on EEPROM and then read or shown on the display at any moments, just giving a proper command to the terminal, with the right message identification number. QTP 12 also manages scrolling messages, to show on an unique line more text than it could be visible without scrolling. For detailed information about messages please read “COMMANDS FOR MESSAGES MANAGEMENT” paragraph.
FIGURE 2: QTP 12 PHOTO IN ALL AVAILABLE MODELS
TECHNICAL FEATURES

GENERAL FEATURES

Resources:
- IP54 frontal
- Software manageable status LED
- 12 software reconfigurable keys
- Buzzer for beep, feedback and keyclick
- Full duplex RS 232 serial line, it can be buffered in RS 422, RS 485 or current loop (option)
- EEPROM size 2 KBytes for configuration, keys code, etc.
- Real Time Clock backed with Lithium battery (option)
- Magnetic cards reader for one track badge (option)
- CAN interface (option)
- Alphanumeric or graphic display in three different models
- Trimmer to set LCD display contrast

Displays:
- alphanumeric LCD 20x2 LED backligthing
- alphanumeric fluorescent 20x2
- graphic fluorescent 140x16

CPU:
- 89C5115 or 89C51CC02 with crystal 14.7456 MHz

Communication 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)

Receive buffer size:
- 30 characters

PHYSICAL FEATURES

Size:
- DIN 72x144: 144 x 72 x 27 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 5 x 7 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: 300 g max.

Mounting:

- Surface or flush panel mounting through clamps
- On a bearing surface

Keys autorepeat: After 500 ms and then every 100 ms

Temperature range: From 0 to 50 °C

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

Connectors:

- CN1: 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

**ELECTRIC FEATURES**

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

RS 422-485 Termination:

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

Hereunder is listed the **QTP 12** power consumption referred to the different display types, already corrected with power supply section efficiency:

<table>
<thead>
<tr>
<th>DISPLAY Model</th>
<th>Consumption +5 Vdc</th>
<th>Consumption 10÷40 Vdc 8÷24 Vac</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCD 20x2 alphanumeric backlit: QTP 12-C2</td>
<td>225 mA</td>
<td>1.5 W</td>
</tr>
<tr>
<td>Fluorescent alphanumeric 20x2: QTP 12-F2</td>
<td>210 mA</td>
<td>1.4 W</td>
</tr>
<tr>
<td>Fluorescent grafic 140x16: QTP 12-GF2</td>
<td>330 mA</td>
<td>2.2 W</td>
</tr>
</tbody>
</table>

**FIGURE 3: CONSUMPTIONS TABLE**

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

In this chapter there are the information for a right installation and correct use of the terminal QTP 12. In detail there are the locations and functions of each connector, of the user settable jumpers and of the trimmer. For the connectors it is described the pin outs, the meaning of the connected signals and some connection examples, that simplify and speed the installation phase.

CONNECTIONS

QTP 12 terminal has 3 connectors that can be linkeded to other devices or directly to the field, according to system requirements.
In this paragraph there are connectors pin out, a short signals description (including the signals direction) and connectors location (see figure 17).
Connectors are accessible from the back of the container, through a breaking.

CN1 - POWER SUPPLY CONNECTOR

CN1 is a 2 pins quick release, screw termina connector, pitch 5 mm.
On CN1 must be connected the single power supply voltage for the terminal that can be one out of three different types:

![Figure 4: CN1 - Power Supply Connector](image)

Signals description:

\[
\begin{align*}
\text{Vac} & = 1 \quad - \text{AC power supply lines connected to on board switching section; } 8+24 \text{ Vac} \\
\text{+Vdc pow} & = 1 \quad - \text{DC power supply lines connected to on board switching section; } +10+40 \text{ Vdc or stabilized +5 Vdc voltage connected to on board logic, as ordered} \\
\text{GND} & = \quad - \text{DC Power supply ground signal.}
\end{align*}
\]

NOTE

For further information about power supply configurations, please refer to paragraph “POWER SUPPLY”.
Figure 5: AC Power Supply 8-24 Vac

Figure 6: DC Power Supply +10-40 Vdc

Figure 7: Stabilized Power Supply +5 Vdc
CN2 - SERIAL LINE CONNECTOR

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

**FIGURE 8: CN2 - SERIAL LINE CONNECTOR**

Signals description:

- **RX RS 232** = I - RS 232 Receive Data signal.
- **TX RS 232** = O - RS 232 Transmit Data signal.
- **RX- RS 422** = I - Negative signal for RS 422 serial differential receive.
- **RX+ RS 422** = I - Positive signal for RS 422 serial differential receive.
- **TX- RS 422** = O - Negative signal for RS 422 serial differential transmit.
- **TX+ RS 422** = O - Positive signal for RS 422 serial differential transmit.
- **RXTX- RS 485** = I/O - Negative signal for RS 485 serial differential receive and transmit.
- **RXTX+ RS 485** = I/O - Positive signal for RS 485 serial differential receive and transmit.
- **RX - C.L.** = I - Negative signal for current loop serial bipolar receive.
- **RX+ C.L.** = I - Positive signal for current loop serial bipolar receive.
- **TX- C.L.** = O - Negative signal for current loop serial bipolar transmit.
- **TX+ C.L.** = O - Positive signal for current loop serial bipolar transmit.
- **GND** = - Ground signal
FIGURE 9: RS 232 POINT-TO-POINT CONNECTION EXAMPLE

FIGURE 10: RS 422 POINT-TO-POINT CONNECTION EXAMPLE

FIGURE 11: RS 485 POINT-TO-POINT CONNECTION EXAMPLE
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 extremis, respectively near the Master unit and the Slave unit at the greatest distance from the Master.

Forcing and terminating circuitry is installed on QTP 12 board. It can be enabled or disabled through specific jumpers, as explained later.

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

For further information please refer to TEXAS INSTRUMENTS Data-Book, "RS 422 and RS 485 Interface Circuits", the introduction about RS 422-485.
**Figure 13: Current Loop 4 Wires Point-to-Point Connection Example**

**Figure 14: Current Loop 2 Wires Point-to-Point Connection Example**
Possible Current Loop connections are two: 2 wires and 4 wires. These connections are shown in figures 11÷13 where it is possible to see the voltage for VCL and the resistances for current limitation (R). The supply voltage varies in compliance with the number of connected devices and voltage drop on the connection cable.

The choice of the values for these components must be done considering that:
- circulation of a 20 mA current must be guaranteed;
- potential drop on each transmitter is about 2.35 V with a 20 mA current;
- potential drop on each receiver is about 2.52 V with a 20 mA current;
- in case of shortcircuit each transmitter must dissipate at most 125 mW;
- in case of shortcircuit each receiver must dissipate at most 90 mW.

For further info please refer to HEWLETT-PACKARD Data Book, (HCPL 4100 and 4200 devices).
**Figure 16: Rear View**

**Figure 17: Jumpers, Connectors, Trimmer, etc. Location.**
CN3 - CAN INTERFACE CONNECTOR

CN3 is a 3 pins, male, 90 degrees, quick release screw terminal connector with 3.54 mm pitch. Through CN3 must be connected the CAN serial communication line by following the standard rules defined by the same protocol. Signal placement has been designed to reduce interference and to easy the connection to the field, according to the standard.

**Signals description:**

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

![Figure 18: CN3 - CAN interface connector](image)

![Figure 19: CAN line connection](image)
Please remind that a CAN network must have two termination resistors (120 Ω) placed at its extremes, respectively near the master unit and the slave unit at the greatest distance from the master. On QTP 12 the terminating circuitry is already installed: it can be connected or not through specific jumper, as explained later, in paragraph "JUMPERS". Should the system to connect be at very different potentials, it is possible to connect also the grounds of the systems, that is pin 1 of CN3, to solve eventual problems of communication and/or correct working.
CONTRAST REGULATION TRIMMER

On **QTP 12** board there is a trimmer that defines the contrast on LCD displays.
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 17.

JUMPERS

On **QTP 12** there are 4 jumpers and one dip switch for card configuration.
Connecting these jumpers, the user can define some parameters of its working modes. Here below is the jumpers list, location and function:

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>CONNECTION</th>
<th>PURPOSE</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1</td>
<td>position 1-3 (jumper a filo)</td>
<td>Configures serial line for standard electric protocol RS 485 (2 wires half duplex). Configures serial line for standard electric protocol RS 422 (4 wires half duplex or full duplex).</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>J2</td>
<td>not connected</td>
<td>Does not connect termination and forcing circuitry to serial line RS 422, RS 485. Connects termination and forcing circuitry to serial line RS 422, RS 485.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td></td>
<td></td>
</tr>
<tr>
<td>J3</td>
<td>not connected</td>
<td>Does not connect 120 Ω CAN line termination resistor. Connects CAN 120 Ω line termination resistor.</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connected</td>
<td></td>
<td></td>
</tr>
<tr>
<td>J4</td>
<td>-</td>
<td>Reserved.</td>
<td>-</td>
</tr>
</tbody>
</table>

**Figure 21: JUMPERS table**

The following tables describe all the right connections of **QTP 12** jumpers with their relative functions.
To recognize these valid connections, please refer to the board printed diagram (serigraph) or to figure 23 of this manual, where the pins numeration is listed; for recognizing jumpers location, please refer to figure 23 again.
The "*" denotes the default connection, or on the other hand the connection set up at the end of testing phase, that is the configuration the user receives.
Further information about purpose of jumpers are reported in the following paragraphs.
**Figure 22:** Components map solder side

**Figure 23:** Components map components side
SERIAL COMMUNICATION SELECTION

Serial line of **QTP 12** can be buffered in RS 232, RS 422, RS 485 or current loop.
By hardware can be selected which one of these electric standards is used, through jumpers connection (as described in the previous tables) and drivers installation.
By software the serial line can be programmed to operate with all the standard physical protocols, in fact the bits per character, parity, stop bits and baud rates can be decided by an opportune setup procedure.
In the following paragraphs there are all the information on serial communication configurations. Some devices needed for RS 422, RS 485 and current loop configurations are not mounted on the board in standard configuration; this is why each fist non-standard (non-RS 232) serial configuration must be always performed by grifo® technicians.
This far the User can change in autonomy the configuration following the informations below:

- **SERIAL LINE IN RS 232** (default configuration)
  
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IC4</td>
<td>= driver MAX 202</td>
</tr>
<tr>
<td>IC2</td>
<td>= no device</td>
</tr>
<tr>
<td>J1</td>
<td>= indifferent</td>
</tr>
<tr>
<td>J2</td>
<td>= not connected</td>
</tr>
<tr>
<td>IC5</td>
<td>= no device</td>
</tr>
<tr>
<td>IC3</td>
<td>= no device</td>
</tr>
<tr>
<td>IC6</td>
<td>= no device</td>
</tr>
</tbody>
</table>

- **SERIAL LINE IN CURRENT LOOP** (option .CLOOP)
  
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IC4</td>
<td>= no device</td>
</tr>
<tr>
<td>IC2</td>
<td>= no device</td>
</tr>
<tr>
<td>J1</td>
<td>= indifferent</td>
</tr>
<tr>
<td>J2</td>
<td>= not connected</td>
</tr>
<tr>
<td>IC5</td>
<td>= no device</td>
</tr>
<tr>
<td>IC3</td>
<td>= driver HP 4200</td>
</tr>
<tr>
<td>IC6</td>
<td>= driver HP 4100</td>
</tr>
</tbody>
</table>

Please remark that current loop serial interface is passive, so it must be connected an active current loop serial line, that is a line provided with its own power supply, like described in figures 11+13. Current Loop Interface can be employed to make both point-to-point and multi-point connections through a 2-wires or a 4-wires connection.

- **SERIAL LINE IN RS 422** (option .RS 422)
  
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IC4</td>
<td>= no device</td>
</tr>
<tr>
<td>IC2</td>
<td>= driver SN 75176 or MAX 483</td>
</tr>
<tr>
<td>J1</td>
<td>= position 2-3</td>
</tr>
<tr>
<td>J2</td>
<td>= (*)</td>
</tr>
<tr>
<td>IC5</td>
<td>= driver SN 75176 or MAX 483</td>
</tr>
<tr>
<td>IC3</td>
<td>= no device</td>
</tr>
<tr>
<td>IC6</td>
<td>= no device</td>
</tr>
</tbody>
</table>

RS 422 electric protocol can be used to make 4-wires full duplex connections both multi-point and point -to-point.
Transmitter abilitation, essential in networks, is managed directly by **QTP 12** selecting the master-slave logic protocol.
FIGURE 24: DRIVER FOR SERIAL COMMUNICATION SELECTION

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

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>IC4</td>
<td>= no device</td>
</tr>
<tr>
<td>IC2</td>
<td>= driver SN 75176 or MAX 483</td>
</tr>
<tr>
<td>IC5</td>
<td>= no device</td>
</tr>
<tr>
<td>IC3</td>
<td>= no device</td>
</tr>
<tr>
<td>IC6</td>
<td>= no device</td>
</tr>
</tbody>
</table>

J1 = position 1-3
J2 = (*)

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 managed by firmware, configured with logic protocol master-slave.

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

(*) If using the RS 422 or RS 485 serial line, it is possible to connect the terminating and forcing circuit on the line by using J2.

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

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

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

---

CAN INTERFACE

Jumper J3 connects or does not connect CAN termination resistor, as described in figure 21.

CAN bus must be a differential line with 60 Ω of impedance so termination resistors must be connected to obtain this value.

This connection, in specific, must be always made in case of point-to-point communication, while in multi-point communication it must be connected only in the cards at the greatest distance, that is at the ends of the lines (please see example of figure 20).

Correct CAN termination contributes remarkably to correct communication; in fact on board interface can suppress transients and is immune against radio frequency and electromagnetic disturbs only if connection to the field is made correctly.

CAN line is not galvanically isolated (as described in paragraph “POWER SUPPLY”) from board supply voltage. Ground of CAN line connected to on board logic GND and is available on a pin of connector CN3.

This latter can be used to equilibrate difference of potentials amongst several CAN systems, but also to shield physical connection, using CAN shielded cable, to obtain the greatest protection against external noise.
**Figure 25:** Photo of QTP 12-C2

**Figure 26:** Photo of QTP 12-F2

**Figure 27:** Photo of QTP 12-GF2
ALIMENTAZIONE

Terminal QTP 12 is provided with a power supply section that solves in an efficient and comfortable way the problem to supply the boards in any situation. It generates energy for all sections of the board: control logic, display, backlighting, serial interfaces, CAN interface, LED, buzzer and keyboard. Here follow voltages required according to the configuration:

Default: This configuration features a switching power supply that requires 10÷40 Vdc or 8÷24 Vac provided through CN1 (polarity MUST be respected in case of DC supply). This allows to supply the terminal using standard inexpensive power sources like transformers, batteries, solar cells, etc. Power supply EXPS-1 can be connected directly to the terminal starting from mains. Please remark that on board switching supply is provided with 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 requests.

Version .5Vdc or .ALIM: This configuration is not provided of any supply section, so 5 Vdc ± 5% stabilized supply voltage must be provided by an external source through CN1 (polarity MUST be respected also in this case). This allows to provide energy to the terminal through power supply, other cards, etc. This configuration is OEM only, please contact 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. QTP 12 is provided with a TransZorb™ protection circuit to avoid damages from malfunctioning or broken supply section. It is also provided with a distributed filtering circuitry that saves the terminal from disturbs or noise from the field, improving the overall system performances. For further information please refer to paragraph “ELECTRIC FEATURES”.
SOFTWARE DESCRIPTION

As already said QTP 12 terminal is a complete video terminal and for this reason any characters received from serial line, if it is not a command, is shown on the display and codes of any pressed external keys, are transmitted to the control master unit.

These operations are automatically performed by on board firmware that is programmed and executed by the QTP 12 CPU.

The on board firmware also allows a local set up which allows to set the physic communication protocol by using the keys and the display.

This manual contains, in addition to the description of the different functions, a complete list of the recognized command sequences, to be used to benefit of the main features of QTP 12.

For each code or code sequence, there is a double description i.e: the mnemonic one through the ASCII characters and the numerical one under decimal and hexadecimal form.

The said commands respect the ADDS View Point standard so all the sequences begin with ESC character corresponding to the 27 decimal code (1B Hex).

LOCAL SET UP

Thanks to a proper local set up mode, some parameters of communication protocol and the key click mode can be set by the user with the simple use of 4 keys.

To enter set up mode the user must power on the QTP 12 and simultaneously he must press the keys number 0 and number 1 (serigraph * and 0) for at least half of a second.

When the set up mode is entered, on the display appears the “** Local Setup **” string and with keys number 10 and number 11 (serigraph 9 and #) the configuration parameters shall be changed as below described:

Number 10 (#): Changes 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 figura of hexadecimal identification name
"NAME (Hex)" second figura of hexadecimal identification name
"EEPROM DATA" initializes data in EEPROM
"SAVE and EXIT" to exit from set up mode

Number 10 (9): Changes current value of 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 figure enclosed in "<>" from 0 to F (def.=80H)
EEPROM DATA: NOINI or INIT (def.=NOINI)
SAVE and EXIT: exits set up and configures QTP 12 with selected parameters
Once exited from set up mode, the selected parameters are saved on EEPROM and they are maintained until another local set up is executed and then 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.

Available options for menus BAUD RATE and STOP BIT define the physical communication protocol with next two parameters unchangeable and set to 8 bits per character and no parity.

Option of remaining menus are described in the following parameters.

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

The local set up is normally executed only one time after the first installation, so the required four external keys can be connected only during this phase and thereafter the **QTP 12** can be used without keys as a simple display unit.

**RECEIVE BUFFER**

**QTP 12** 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 operation.

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

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

The master unit must stop the transmission until the **QTP 12** 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.

**KEYBOARD ACQUISITION**

When **QTP 12** recognizes an external key pressure, if normal communication is used it translates it and then transmits the determined code on serial line, by using the rules described in the following paragraph.

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

Moreover an auto repeat function of the stroked key is implemented so when **QTP 12** recognizes the pressure on a key for a time greater than 0.5 sec, it will start the serial transmission of its code about each 0.1 sec, and it lasts until that specific key is released.

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

Here are the table which shows the codes that QTP 12 sends on serial line when a key is pressed; the code here is shown in decimal, hexadecimal and ASCII mnemonic format, like for command sequences:

![Figure 28: Keys enumeration and location](image)

<table>
<thead>
<tr>
<th>KEY NUMBER</th>
<th>SERIGRAPHY</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>* ESC</td>
<td>42</td>
<td>2A</td>
<td>*</td>
</tr>
<tr>
<td>1</td>
<td>0 +.</td>
<td>48</td>
<td>30</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1 FUNZ</td>
<td>49</td>
<td>31</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2 ABC</td>
<td>50</td>
<td>32</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>3 DEF</td>
<td>51</td>
<td>33</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>4 GHI</td>
<td>52</td>
<td>34</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>5 JKL</td>
<td>53</td>
<td>35</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>6 MNO</td>
<td>54</td>
<td>36</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>7 PQRS</td>
<td>55</td>
<td>37</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>8 TUV</td>
<td>56</td>
<td>38</td>
<td>8</td>
</tr>
<tr>
<td>10</td>
<td>9 WXYZ</td>
<td>57</td>
<td>39</td>
<td>9</td>
</tr>
<tr>
<td>11</td>
<td># ENTER</td>
<td>35</td>
<td>23</td>
<td>#</td>
</tr>
</tbody>
</table>

These codes are those transmitted under default condition, i.e. the configuration the user receives, but they can be comfortably reconfigured by using a specific command. This feature really simplifies the management software development in fact the master unit can change the codes according with his requirements and it can also disable the keys.
COMMUNICATION MODALITIES

QTP 12 features two different serial communication modalities:

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

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

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

MASTER SLAVE COMMUNICATION MODE

The Master Slave mode uses the 9 bits communication technique. In addition to the 8 data bit also a 9th bit is managed as it is needed for recognizing between a call coming from the "Master" to any of the "Slave" structures and a simple info transmission between Master and the selected device.

When 9th bit is placed at 1, the data byte has to contain the name, or identifying code, of the device towards it needs to communicate, while by placing this particular bit at 0, it is possible to take out or supply info at this device.

When QTP 12 is used, the identifying code must be that one set by the local set up programm on the "NAME (Hex)" entry. When this byte is sent (with 9th bit set to 1) the QTP 12 recognizes itself and it waits the string containing chars, data or commands. In this string there must only be a command that involves the return of an information sent via serial line from QTP 12 part; if there is more than one command the results of the remaining ones are ignored.

Between the transmission of a char. and the next one there must be an interval of time shorter than the Time Out, as elapsed this delay, the QTP 12 will consider the data string ended and it will begin the answering phase. The Time Out value for each baud rate is below described:

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

Master unit must wait for:

"character transmission time"+"Time out"
before reaching the first character of the answering string returned by the QTP 12.
The answer consists in a byte containing the code of the pressed key (FF Hex, no key is pressed) or a data string related to a reading command sent in the previous request.
Please remark that answer is provided also in case the only identification name is requested, simplify the check for keys pressed or invalid commands.

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

<table>
<thead>
<tr>
<th>Master</th>
<th>QTP 12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sends “Reading of version number” command, that is the characters sequence:</td>
<td>Receives character of the command and verifies the Time Out of 550 µ sec</td>
</tr>
<tr>
<td>80H with ninth bit set to 1</td>
<td></td>
</tr>
<tr>
<td>56H with ninth bit set to 0</td>
<td>Recognizes command sequence, executes the command and stores answer for next command</td>
</tr>
<tr>
<td>delay between characters lower than 550 µsec</td>
<td></td>
</tr>
<tr>
<td>Waits for 837 µsec</td>
<td>Sends the answer, which is the code of the eventual key pressed with ninth bit set to 0</td>
</tr>
<tr>
<td>Receives one character of answer</td>
<td></td>
</tr>
<tr>
<td>Sends a string to show on the display, that is the character sequence:</td>
<td>Receives character of the command and verifies the Time Out of 550 µ sec</td>
</tr>
<tr>
<td>1° character of string with ninth bit set to 0</td>
<td></td>
</tr>
<tr>
<td>2° character of string with ninth bit set to 0</td>
<td>Recognizes command sequence and shows on the display the characters of the string</td>
</tr>
<tr>
<td>: : : : : : :</td>
<td></td>
</tr>
<tr>
<td>delay between characters lower than 550 µsec</td>
<td>Transmits saved response which is the version number required in previous command, with ninth bit set 0</td>
</tr>
<tr>
<td>Waits for 837 µsec</td>
<td></td>
</tr>
<tr>
<td>Receives three characters of answer with the version number previously requested</td>
<td></td>
</tr>
<tr>
<td>Sends key pressed check command, that is the character sequence:</td>
<td>Receives character of the command and verifies the Time Out of 550 µ sec</td>
</tr>
<tr>
<td>80H with ninth bit set to 1</td>
<td></td>
</tr>
<tr>
<td>Waits for 837 µsec</td>
<td>Recognizes sequence without commands so performs no operation</td>
</tr>
<tr>
<td>Receives one character of answer corresponding to code of eventual key pressed</td>
<td>Sends the answer, which is the code of the eventual key pressed with ninth bit set to 0</td>
</tr>
</tbody>
</table>

Figure 30: Example of master slave communication
Several demo programs, written in different languages, are provided with QTP 12. They implement master slave communication and can be used directly by the user or modified according to the specific needs. When the master unit is a PC, the user can also take advantage of comfortable DLL libraries that allow to manage high level master slave communication, this means without having to worry about management of ninth bit, timings, eventual electric protocol converters, etc. Also these libraries are provided with the first purchase, complete of user manual, on a CD.

NOTES:

1) To ensure right command execution, between a call and the next one it is necessary to wait for a time that is related to the number of commands sent and type of operations these latter ones 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 opportunally, before any characters transmission, 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.

DATA STORED IN EEPROM

On board EEPROM of QTP 12 stores data used and/or changed through the specific commands. The menu "EEPROM DATA" of local setup allows the user to decide whether to leave these data unchanged (NOINI option) or to set them to their default value (INIT option) to restore the configuration of the board when delivered, as follows:

- presence byte -> 255 (FFH)
- keys codes -> reported in table of figure 29
- models of user defineable characters -> 255 (FFH)
- messages -> 255 (FFH)

exiting from local set up a string is shown on the display with a progress bar; * inform about the status of the operation. Initialization phase duration, and so the number of * printed, depends on EEPROM size and can be up to 20 seconds. User must be very careful with EEPROM initialization, in fact all previously saved data are lost.
CHARACTERS VISUALIZATION ON THE DISPLAY

**QTP 12** shows on its display all the received characters having a code included in the range $0 \leq 255$ ($00 \leq FF$ Hex) but the one that identifies a command sequence ($27 = 1BH$).

The character is visualized on the current cursor position and this latter will go to the next position; if it is placed on the last character of the display (right down corner), it will be placed on home position (left up corner).

The correspondence between codes and displayed characters is defined by the following rules:

<table>
<thead>
<tr>
<th>Codes</th>
<th>Characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0 \leq 15$ ($00 \leq 0F$ Hex)</td>
<td>User defineable</td>
</tr>
<tr>
<td>$16 \leq 31$ ($10 \leq 1F$ Hex)</td>
<td>Special and different according with installed display</td>
</tr>
<tr>
<td>$32 \leq 127$ ($20 \leq 7F$ Hex)</td>
<td>Standard ASCII</td>
</tr>
<tr>
<td>$128 \leq 255$ ($80 \leq FF$ Hex)</td>
<td>Special and different according with installed display</td>
</tr>
</tbody>
</table>

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 12** 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 model of **QTP 12** 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”.

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

*FIGURE 31: PHOTO OF CHARACTERS AVAILABLE ON QTP 12-GF2*
COMMANDS FOR CURSOR POSITION

Here follows the list of the cursor positioning commands.

CURSOR LEFT

Code: 21
Hex code: 15
Mnemonic: NACK

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

CURSOR RIGHT

Code: 6
Hex code: 6
Mnemonic: ACK

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

CURSOR DOWN

Code: 10
Hex code: A
Mnemonic: LF

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

CURSOR UP

Code: 26
Hex code: IA
Mnemonic: SUB

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

HOME

Code: 1
Hex code: 1
Mnemonic: SOH

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

<table>
<thead>
<tr>
<th>Code:</th>
<th>13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>D</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>CR</td>
</tr>
</tbody>
</table>

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

CARRIAGE RETURN+LINE FEED

<table>
<thead>
<tr>
<th>Code:</th>
<th>29</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>1D</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>GS</td>
</tr>
</tbody>
</table>

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

ABSOLUTE CURSOR PLACEMENT

<table>
<thead>
<tr>
<th>Code:</th>
<th>27 89 \texttt{r c}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>1B 59 \texttt{r c}</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>\texttt{ESC Y ASCII(r) ASCII(c)}</td>
</tr>
</tbody>
</table>

The cursor is moved to the absolute position indicated by \texttt{r} and \texttt{c}.

These characters are the row and column values of the new desired position referred to coordinate 0, 0 of the Home position, plus a constant offset of 32 (20 Hex).

If, for example, the user wants to place the cursor on the second line, third column (row 1, column 2), the following byte sequence must be sent:

\texttt{27 89 33 34} or \texttt{1B 59 21 22 Hex} or \texttt{ESC Y ! "}

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

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

BACKSPACE

- **Code:** 8
- **Hex code:** 8
- **Mnemonic:** BS

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

CLEAR PAGE

- **Code:** 12
- **Hex code:** C
- **Mnemonic:** FF

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

CLEAR LINE

- **Code:** 25
- **Hex code:** 19
- **Mnemonic:** EM

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

CLEAR END OF LINE

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

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

CLEAR END OF PAGE

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

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

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

CURSOR OFF

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

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

STEADY STATIC CURSOR ON

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

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

BLINKING BLOCK CURSOR ON

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

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

In the following paragraphs are described all the commands that manage the data saved on QTP 12 on board EEPROM; there are other commands that indirectly use this memory device but they are described in next paragraphs.

REQUEST FOR EEPROM WRITING POSSIBILITY

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

This command checks if the QTP 12 is ready for writing data on its on board EEPROM. This command must be executed any time there are data to be saved on this type of memory.

When QTP 12 receives this command, it answers with the following codes:

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

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

WRITING OF PRESENCE BYTE

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

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

This byte has a reserved allocation on the on board EEPROM that, once it is set with the desired value, it allows for example, to verify that QTP 12 runs correctly, or if there are some communication problems on the serial line.

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

READING OF PRESENCE BYTE

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

The QTP 12 sends back on the serial line the value of its presence byte.

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

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

In the following paragraphs are described all the general purpose commands that manage some of the QTP 12 features.

READING OF VERSION NUMBER

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

On the serial line is returned a string of 3 characters containing the program managing version that is resident and executed by QTP 12.

For example with firmware version 1.3 the following characters will be transmitted:

49  46  51 or 31  2E  33 Hex or 1.3

BEEP

Code: 7
Hex code: 7
Mnemonic: BEL

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

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 (provided of code less than 32 = 20H) and the single character commands.

The selected modality is defined by mode value, 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 available inside “CHARACTER VISUALIZATION ON THE DISPLAY” paragraph.
FLUORESCENT DISPLAY BACKLIGHT SETTING

<table>
<thead>
<tr>
<th>Code</th>
<th>27 108 lum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code Hex</td>
<td>1B 6C lum</td>
</tr>
<tr>
<td>Mnemonic</td>
<td>ESC 1 ASCII(lum)</td>
</tr>
</tbody>
</table>

Sets fluorescent display backlight setting to one of the four possible values stored in `lum`:

- 0 (00 Hex) -> 100% of backlight
- 1 (01 Hex) -> 75% of backlight
- 2 (02 Hex) -> 50% of backlight
- 3 (03 Hex) -> 25% of backlight

If parameter is not valid, command is ignored.

**N.B.** This command is available only with models **QTP 12-F2** and **QTP 12-GF2**.
In case of **QTP 12-C2** with display LCD, command must not be sent because it produces the writing of undesired characters.

LED AND BUZZER ACTIVATION

<table>
<thead>
<tr>
<th>Code</th>
<th>27 50 device attr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code Hex</td>
<td>1B 32 device attr</td>
</tr>
<tr>
<td>Mnemonic</td>
<td>ESC 2 ASCII(device) ASCII(attr)</td>
</tr>
</tbody>
</table>

Device indicated in `device` is activated using attributed specified in `attr`.
In detail:

- 0 (00 Hex) -> status LED
- 255 (FF Hex) -> buzzer

while available attributes are:

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

If parameter is not valid, command is ignored.

For example, to activate LED with attribute of device intermittent, send to the terminal the sequence

27 50 0 85 or 1B 32 00 55 Hex or ESC 2 NUL U
COMMANDS FOR MESSAGE MANAGEMENT

In the following paragraphs are described all the commands that manage messages on QTP 12. 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 commands. Furthermore a comfortable program for PC, named QTP EDIT, allows any user to edit the messages, save and load them on PC disks and transmit/receive them directly to/from QTP serially connected to PC. QTP 12 features one EEPROM sized 2048 bytes that can store up to 97 messages from 0 to 96.

READING OF THE LAST STORAGED MESSAGE NUMBER

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

This command returns on the serial line the number of the last messages that can be saved on EEPROM. It is always 96 (60 Hex) as previously described. This command is important for other QTP models that perform variable messages number and has been implemented for compatibility with those models.

MESSAGE STORAGE

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

This command stores the 20 characters message, with number indicated as mess. num., on the on board EEPROM. The 20 chars which form the message must be visualizable on the display so they must be included in the range 0÷255 (0÷FF Hex).

The message number must be included in the range of 0÷96.

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

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

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

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

The message number must be included in the range of 0÷69, if this number is out of range, the command is ignored.

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

MESSAGE VISUALIZATION

- **Code:** 27 33 68 mess. num. n
- **Hex code:** 1B 21 44 mess. num. n
- **Mnemonic:** ESC ! D ASCII(mess. num.) ASCII(n)

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

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

The message number must be included in the range 0÷69, if this number is out of range, the command is ignored.

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

For alphanumeric mode all the three displays can show at most 40 characters so in this case maximum number is 2, so n ranges from 1 to 2, if its value is out of this range, command is ignored.

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

For example; to visualize the messages number 10 and 11, it will be necessary to send the following sequence:

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 check the EEPROM ready; in fact if it is not ready the command is ignored and 21 (15 Hex) = NAK is returned.
SCROLLING MESSAGES VISUALIZATION

**Code:**
27 33 83 mess.num. n.char

**Hex code:**
1B 21 53 mess.num. n.char

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

This command visualizes a `n.char` characters message on the **display first line** in sliding mode.
The message is shifted from right to left and so the user can visualize a very long string on the display.
The string of "`n.char`" characters, begins with the first character of the "`mess.num.`" message already stored in EEPROM and continues with next messages.
The "`mess.num.`" value must be included in the range 0÷69; if the value is out of range this command is ignored.
The "`n.char`" parameter is used as follow:

- `0` -> Stops the message visualization in sliding mode.
- `20÷200` -> Activates sliding.

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

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

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

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

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

Below are described the commands that can be used to manage the external keys, connected to QTP 12. Detailed information about keys management and codes transmitted by the terminal, are available in “KEYBOARD ACQUISITION” paragraph.

KEY RECONFIGURATION

<table>
<thead>
<tr>
<th>Code:</th>
<th>27 55 key n. code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>1B 37 key n. code</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>ESC 7 ASCII(key n.) ASCII(code)</td>
</tr>
</tbody>
</table>

When the selected key n. is reconfigured, each time it is pressed, the card will send the new specified code on serial line.

The value of key n. to be reconfigured must be included in the range 0÷11 (0÷B Hex) otherwise the command is ignored, and it will substitute the codes described in figure 28.

The code value can vary in the range 0÷254 (00÷FE Hex) but value 255 (FF Hex) indicates that the key is disabled and when it will be pressed the QTP will not send any code.

Figure 20 reports the default key codes and the paragraph “DATA STORED IN EEPROM” indicates how to restore these codes in case of unwanted changes.

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

KEYCLICK ON WITHOUT MEMORIZATION

<table>
<thead>
<tr>
<th>Code:</th>
<th>27 53</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>1B 35</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>ESC 5</td>
</tr>
</tbody>
</table>

The keyclick function is switched on so there is a sound feedback when a key is pressed.

This setting is not saved inside the on board EEPROM so if the terminal is powered off and on it goes back to the previous condition, defined and saved in local set up mode.

KEYCLICK OFF WITHOUT MEMORIZATION

<table>
<thead>
<tr>
<th>Code:</th>
<th>27 54</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hex code:</td>
<td>1B 36</td>
</tr>
<tr>
<td>Mnemonic:</td>
<td>ESC 6</td>
</tr>
</tbody>
</table>

The keyclick function is disabled so there is not sound feedback when a key is pressed.

This setting is not saved inside the on board EEPROM so if the terminal is powered off and on it goes back to the previous condition, defined and saved in local set up mode.
KEYCLICK ON WITH MEMORIZATION

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

This command enables KeyClick function, so there is an audible feedback when a key is pressed. This setting is stored on the on board EEPROM so if the card is turned off and on, it keeps the current condition.

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

KEYCLICK OFF WITH MEMORIZATION

Code: 27 33 54  
Hex code: 1B 21 36  
Mnemonic: ESC ! 6

This command disables KeyClick function, so there is not audible feedback when a key is pressed. This setting is stored on the on board EEPROM so if the card is turned off and on, it keeps the current condition.

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

F I G U R E  3 2 :  P H O T O  O F  F R O N T A L  A N D  K E Y B O A R D
**COMMANDS FOR USER CHARACTERS**

**QTP 12** 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:

![User Characters Model](image)

**Figure 33: User Characters Model**

The user characters representation is really simple in fact it is sufficient to send the proper code (0 to 7 or 8 to 15) with a possible previous setting of representation mode, through OPERATING MODE SELECTION command.

When the user character are saved their pixels patterns are written on EEPROM and then they are reloaded on display any time the terminal is powered on.

**NOTE:** On **QTP 12-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 to QTP 12 with the following meaning:

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

This command loads on the display the pattern of the user character nchr with the value placed in the eight byte byte Pat 0 + Pat 7, as described in figure 33; the pattern is only defined but not saved, so if QTP 12 is turned off and on the user character nchr doesn't maintain 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 31
```

or

```
1B 42 05 1F 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 to QTP 12 with the following meaning:

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

This command loads on the display the pattern of the user character nchr with the value placed in the eight byte byte Pat 0 + Pat 7, as described in figure 33; moreover the pattern is also saved on EEPROM, so if QTP 12 is turned off and on the user character nchr mantain the supplied pattern.

**NOTE:**

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

Execution time is about 80 msec: if several commands must follow this, it is better to insert a delay to avoid input buffer overflow.
GRAPHIC COMMANDS

**QTP 12-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.

A set of graphic commands is available; such commands are based on pixels (smallest visible entity) organized in the following coordinates system:

N.B. Please remark that following graphic commands can be used only on model **QTP 12-GF2** while the remaining models do not recognize them as commands so they show them on display.

In addition, **QTP 12-GF2** recognizes both graphic and alphanumeric commands.

**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.

After command execution, the cursor (if enabled) is shown in the last position decided by previous commands.

After power on, alphanumeric mode is immediately set.
**Figure 35: First graphic example**

**Figure 36: Second graphic example**
GRAPHIC MODE SETTING

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

This command sets graphic mode, it enables the interpretation of data as graphic and not as commands.

After execution of this command there is no effect on display, but characters received are not interpreted as commands any more; they go directly to the display.

One-character commands have no effect until alphanumeric mode is restored.

On other hand, commands starting with ESC = 27 = 1BH are always enabled, despite the mode selected.

After power on, alphanumeric mode is set by default.

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 numbers is absolute, so it is not affected by all other parameters and is beyond the range of normal alphanumeric positioning.

Characters received after this command are displayed from indicated point, and 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 is included in size of display used.

**NOTE:**
Code 0 (NUL) is present for compatibility with future versions of firmware: it must be always transmitted anyway.

GRAPHIC AREA SETTING

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

Sets graphic work area and the action to make on it.

Top left corner of area is set with coordinates x1 and y1, bottom right corner is x2 and y2.

Values of y1, y2 and x1, x2 must be in the ranges 0+15 and 0+139, that is included in size of display used.

Byte cmd selects the action to perform to graphic area also according to the next bytes received from serial line:
cmd = 67 (43 Hex) C -> Clears selected area.

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

72 (48 Hex) H -> Draw the 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 -> Draw the following vertical graphic data with horizontal shift.

104 (68 Hex) h -> Draw the following horizontal graphic data with vertical shift.

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

118 (76 Hex) v -> Draw the following vertical graphic data with vertical shift.

About commands that draw data in graphic area (H,h,V,v):
next bytes sent to the terminal are used as graphic data that will decide pixel activation of display. Correspondance between 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.
This command can complete execution in two ways:
- by filling/drawing the selected graphic area
- by interruption because another command is received

This latter case stops execution so only pixels already represented at that moment will be visualized.
For example, to draw an arrow like the one in the following figure on the top left corner:

\[ \begin{array}{cccccc}
0 & 0 & 0 & 255 & 255 & 255 \\
00 & 00 & 00 & FF & FF & FF
\end{array} \]

\[ \begin{array}{cccccc}
0 & 0 & 32 & 48 & 56 & 252 \\
00 & 00 & 00 & 20 & 30 & 38
\end{array} \]

\[ \begin{array}{cccccc}
15 & 8 & FC & FE & FC & 38 \\
30 & 20 & Hex
\end{array} \]

**Figure 37: Example of graphic drawing**

send the command sequence first:

\[ \begin{array}{cccccc}
27 & 241 & 0 & 0 & 15 & 8 & 72 \\
1B & F1 & 00 & 00 & 0F & 08 & 48
\end{array} \] Hex
**Figure 38: Horizontal data and horizontal shift**

**Figure 39: Horizontal data and vertical shift**
**Figure 40:** Vertical data and horizontal shift

**Figure 41:** Vertical data and vertical shift
GRAPHIC FONT SETTING

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

Sets font for next alphanumeric characters visualization in graphic mode.
In graphic mode byte are shown on display as characters anyway, if a drawing command is not under execution.
In this mode a different font from the one in alphanumeric mode can be selected.
It can be selected with parameter **font**:

\[
\begin{align*}
\text{font} & = 65 \ (41 \text{ Hex}) \ A \quad \rightarrow \ \text{Proportional spacing minifont 3x5+5x5 pixels.} \\
         & = 97 \ (61 \text{ Hex}) \ a \quad \rightarrow \ \text{Proportional spacing minifont 3x5+5x5 pixels.} \\
         & = 66 \ (42 \text{ Hex}) \ B \quad \rightarrow \ \text{Katakana font 5x7 pixels.} \\
         & = 67 \ (43 \text{ Hex}) \ C \quad \rightarrow \ \text{Font Katakana font 10x14 pixels.} \\
         & = 98 \ (62 \text{ Hex}) \ b \quad \rightarrow \ \text{Font Europeo font 5x7 pixels.} \\
         & = 99 \ (63 \text{ Hex}) \ c \quad \rightarrow \ \text{Font Europeo font 10x14 pixels.} \\
         & = 49 \ (31 \text{ Hex}) \ 1 \quad \rightarrow \ 1 \ \text{pixel line spacing.} \\
         & = 50 \ (32 \text{ Hex}) \ 2 \quad \rightarrow \ 2 \ \text{pixels line spacing.}
\end{align*}
\]

The first six font selection commands are mutually exclusive, while the line spacing selections add their effect to the font selection commands.
So, each of the five fonts can be set with line spacing of 1 or 2 pixels, obtaining 10 different fonts.
This selection is valid only in graphic mode, in alphanumeric mode only the classic font shown in figure B1 with line spacing 1 pixel is used.
After a power on alphanumeric mode with is enabled by default, while entering in graphic mode font Katakana 5x7, line spacing 1 pixel, is automatically enabled.
For further information about available characters with described fonts please refer to appendix B, while figure 31 shows three of the ten available fonts.
EXTERNAL CARDS

The typical application of QTP 12 are those that require interaction between operator and controlled machine for data, status, measures, information exchanges. The card can be connected to a wide range of programmable devices provided of one serial communication line produced by grifo®, or to many system of other companies.

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

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

**GPC® 323**  
General Purpose Controller 51 family  
80C32 µP, 14 MHz; Full CMOS; 1 RS 232 line (software); 1 RS 232 or RS 422-485 or Current Loop line; 24 TTL I/O lines; 11 A/D 12 bits lines; 3 Timers Counters; 64K EPROM; 64K RAM; 32K RAM and RTC backed; 32K DIL EEPROM; 8K serial EEPROM; Buzzer; 2 Activity LED; Watch dog; 5 readable DIPs; LCD Interface; **ABACO® I/O BUS**.

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

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

**GPC® 154**  
“4” Type General Purpose Controller Z80  
84C15 µP, 10 MHz; full CMOS; 1 RS 232 line; 1 RS 232 or RS 422-485 line; 16 TTL I/O lines; 512K EPROM or FLASH; 512K RAM and RTC backed; 8K serial EEPROM; 2÷4 timers/counters;Watch dog; 2 readable DIPs; LCD Interface; **ABACO® I/O BUS**; 5Vdc power supply. Size100x50 mm.

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

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

GPC® AM4
General Purpose Controller ATmega103
Microprocessor ATmega103 at 5.5 MHz; CMOS implementation; 128K internal FLASH; 32K SRAM; Back-Up with Lithium battery internal or external; 4K internal EEPROM; 1 serial line RS 232, RS 422, RS 485 or current loop; 16 I/O TTL; 8 linee A/D resolution 10 bits; 3 timers/counters; Watch Dog; Real Time Clock; ABACO® I/O BUS expansion. Interface for ISP programming.

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

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

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

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

GPC® R/T94
General Purpose Relays/transistors 9 inputs 4 outputs
CMOS card, 14 MHz 89C4051 CPU; 4K FLASH; 128 byte RAM; 256 byte SRAM+RTC backed through battery; 1K serial EEPROM; 1 RS 232, RS 422, RS 485 or current loop line; 9 optocoupled NPN inputs; 4 relays outputs (5 A) or transistor (4A 45 Vdc) optocoupled; I/O lines displayed by LEDs; 1 counter timer.+5 Vdc power supply or 8÷24 Vac wide range; plastic container for Ω rails.
FIGURE 42: AVAILABLE CONNECTIONS DIAGRAM

- **BUZZER su scheda**
- **LED di attività**
- **3 diversi tipi di display:**
  - LCD 20x2 alfanumerico
  - VFD 20x2 alfanumerico
  - VFD 140x16 grafico
- **Tastiera a membrana da 12 tasti con doppia serigrafia**

**CONTENITORE DIN 72x144**

**Alimentazione:**
- +5 Vdc
- +10÷40 Vdc
- 8÷24 Vac

**Linea CAN**

**Linea seriale RS 232, RS 422, RS 485, Current Loop**

**PC o Macintosh**
**PLC**
**QTP**
**GPC®**

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

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

SBP 02-xx
Switch BLOCK Power xx version
Low cost switching power supply able to generate voltage from +5 to +40 Vdc and current up to 2.5 A; Input from 12 to 24 Vac; Connection for DIN C Type and Ω rails.
APPENDIX A: COMMAND CODES SUMMARY TABLES

The tables of this appendix list a summary of all the command sequences recognized by QTP 12.

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Home</td>
<td>01</td>
<td>01</td>
<td>SOH</td>
</tr>
<tr>
<td>Cursor left</td>
<td>21</td>
<td>15</td>
<td>NACK</td>
</tr>
<tr>
<td>Cursor right</td>
<td>06</td>
<td>06</td>
<td>ACK</td>
</tr>
<tr>
<td>Cursor down</td>
<td>10</td>
<td>0A</td>
<td>LF</td>
</tr>
<tr>
<td>Cursor up</td>
<td>26</td>
<td>1A</td>
<td>SUB</td>
</tr>
<tr>
<td>Carriage return</td>
<td>13</td>
<td>0D</td>
<td>CR</td>
</tr>
<tr>
<td>Carriage return+line feed</td>
<td>29</td>
<td>1D</td>
<td>GS</td>
</tr>
<tr>
<td>Absolute cursor position</td>
<td>27  89 r c</td>
<td>1B 59 r c</td>
<td>ESC Y ASCII(r) ASCII(c)</td>
</tr>
<tr>
<td>Back space</td>
<td>08</td>
<td>08</td>
<td>BS</td>
</tr>
<tr>
<td>Clear page</td>
<td>12</td>
<td>0C</td>
<td>FF</td>
</tr>
<tr>
<td>Clear line</td>
<td>25</td>
<td>19</td>
<td>EM</td>
</tr>
<tr>
<td>Clear end of line</td>
<td>27  75</td>
<td>1B 4B</td>
<td>ESC K</td>
</tr>
<tr>
<td>Clear end of page</td>
<td>27  107</td>
<td>1B 6B</td>
<td>ESC k</td>
</tr>
<tr>
<td>Cursor off</td>
<td>27  80</td>
<td>1B 50</td>
<td>ESC P</td>
</tr>
<tr>
<td>Steady cursor on</td>
<td>27  79</td>
<td>1B 4F</td>
<td>ESC O</td>
</tr>
<tr>
<td>Blinkling block cursor on</td>
<td>27  81</td>
<td>1B 51</td>
<td>ESC Q</td>
</tr>
<tr>
<td>Beep</td>
<td>07</td>
<td>07</td>
<td>BEL</td>
</tr>
<tr>
<td>Reading of version number</td>
<td>27  86</td>
<td>1B 56</td>
<td>ESC V</td>
</tr>
<tr>
<td>Operating mode selection</td>
<td>27  65 mode</td>
<td>1B 41 mode</td>
<td>ESC A ASCII(mode)</td>
</tr>
<tr>
<td>LED and BUZZER activation</td>
<td>27  50 device attr</td>
<td>1B 32 device attr</td>
<td>ESC 2 ASCII(device) ASCII(attr)</td>
</tr>
</tbody>
</table>

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

**Figure A2: Command codes summary table (2 of 3)**
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fluorescent display</td>
<td>27 108</td>
<td>1B 6C lum</td>
<td>ESC 1 ASCII(lum)</td>
</tr>
<tr>
<td>backlight setting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Graphic cursor absolute</td>
<td>27 206</td>
<td>1B CE y x</td>
<td>ESC ASCII(206) ASCII(y) ASCII(x)</td>
</tr>
<tr>
<td>position</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Alphanumeric mode setting</td>
<td>27 208</td>
<td>1B D0</td>
<td>ESC ASCII(208)</td>
</tr>
<tr>
<td>Graphic mode setting</td>
<td>27 209</td>
<td>1B D1</td>
<td>ESC ASCII(209)</td>
</tr>
<tr>
<td>Graphic area setting</td>
<td>27 241</td>
<td>1B F1 x1 y1 x2 y2 cmd</td>
<td>ESC ASCII(241) ASCII(x1) ASCII(y1) ASCII(x2) ASCII(y2) ASCII(cmd)</td>
</tr>
<tr>
<td>Graphic font setting</td>
<td>27 242</td>
<td>1B F2 font</td>
<td>ESC ASCII(242) ASCII(font)</td>
</tr>
</tbody>
</table>

**Figure A3: Command codes summary table (3 of 3)**
APPENDIX B: DISPLAY CHARACTERS

The following tables show the character sets displayed on QTP 12 for all the possible received characters, according with ordered display and model. Even the not ASCII characters (or special characters) change when the display type changes and if the user requires a character set different from those described in the following figures, he can take a direct contact with grifo®.

**Table:** QTP 12-F2, GF2 in Alphanumeric Mode Characters Table

| L | H | 00 | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F |
| 00 | User chr 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 01 | User chr 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 02 | User chr 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 03 | User chr 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 04 | User chr 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 05 | User chr 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 06 | User chr 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 07 | User chr 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 08 | User chr 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 09 | User chr 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0A | User chr 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0B | User chr 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0C | User chr 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0D | User chr 5 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0E | User chr 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0F | User chr 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

**Figure B1:** QTP 12-F2, GF2 in Alphanumeric Mode Characters Table
### Higher 4-bit (D4 to D7) of Character Code (Hexadecimal)

<table>
<thead>
<tr>
<th>L</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>User chr 0</td>
<td>!</td>
<td>G</td>
<td>p</td>
<td>P</td>
<td>G</td>
<td>2</td>
<td>a</td>
<td>$</td>
<td>t</td>
<td>m</td>
<td>p</td>
<td>t</td>
<td>P</td>
<td>E</td>
<td>T</td>
</tr>
<tr>
<td>1</td>
<td>User chr 1</td>
<td>!</td>
<td>1</td>
<td>A</td>
<td>0</td>
<td>a</td>
<td>q</td>
<td>a</td>
<td>e</td>
<td>i</td>
<td>J</td>
<td>t</td>
<td>y</td>
<td>v</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>User chr 2</td>
<td>!</td>
<td>2</td>
<td>B</td>
<td>R</td>
<td>b</td>
<td>R</td>
<td>e</td>
<td>E</td>
<td>6</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>z</td>
<td>z</td>
<td>z</td>
</tr>
<tr>
<td>3</td>
<td>User chr 3</td>
<td>!</td>
<td>#</td>
<td>5</td>
<td>C</td>
<td>S</td>
<td>c</td>
<td>s</td>
<td>a</td>
<td>a</td>
<td>G</td>
<td>7</td>
<td>7</td>
<td>E</td>
<td>4</td>
<td>Y</td>
</tr>
<tr>
<td>4</td>
<td>User chr 4</td>
<td>!</td>
<td>4</td>
<td>D</td>
<td>T</td>
<td>d</td>
<td>t</td>
<td>a</td>
<td>a</td>
<td>4</td>
<td>7</td>
<td>0</td>
<td>7</td>
<td>z</td>
<td>6</td>
<td>u</td>
</tr>
<tr>
<td>5</td>
<td>User chr 5</td>
<td>!</td>
<td>5</td>
<td>E</td>
<td>U</td>
<td>e</td>
<td>u</td>
<td>a</td>
<td>a</td>
<td>2</td>
<td>7</td>
<td>7</td>
<td>4</td>
<td>h</td>
<td>m</td>
<td>n</td>
</tr>
<tr>
<td>6</td>
<td>User chr 6</td>
<td>!</td>
<td>8</td>
<td>6</td>
<td>F</td>
<td>U</td>
<td>t</td>
<td>u</td>
<td>a</td>
<td>G</td>
<td>w</td>
<td>w</td>
<td>9</td>
<td>0</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>7</td>
<td>User chr 7</td>
<td>!</td>
<td>7</td>
<td>G</td>
<td>w</td>
<td>w</td>
<td>s</td>
<td>o</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
</tr>
<tr>
<td>8</td>
<td>User chr 0</td>
<td>!</td>
<td>8</td>
<td>H</td>
<td>X</td>
<td>h</td>
<td>x</td>
<td>e</td>
<td>e</td>
<td>9</td>
<td>+</td>
<td>+</td>
<td>3</td>
<td>k</td>
<td>k</td>
<td>k</td>
</tr>
<tr>
<td>9</td>
<td>User chr 1</td>
<td>!</td>
<td>9</td>
<td>I</td>
<td>y</td>
<td>i</td>
<td>y</td>
<td>e</td>
<td>a</td>
<td>1</td>
<td>7</td>
<td>T</td>
<td>T</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>A</td>
<td>User chr 2</td>
<td>**</td>
<td>J</td>
<td>2</td>
<td>j</td>
<td>z</td>
<td>e</td>
<td>e</td>
<td>U</td>
<td>a</td>
<td>Z</td>
<td>2</td>
<td>3</td>
<td>w</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>B</td>
<td>User chr 3</td>
<td>!</td>
<td>K</td>
<td>K</td>
<td>K</td>
<td>G</td>
<td>i</td>
<td>F</td>
<td>A</td>
<td>L</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>n</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>C</td>
<td>User chr 4</td>
<td>!</td>
<td>L</td>
<td>\</td>
<td>l</td>
<td>l</td>
<td>l</td>
<td>l</td>
<td>i</td>
<td>n</td>
<td>a</td>
<td>a</td>
<td>a</td>
<td>a</td>
<td>a</td>
<td>a</td>
</tr>
<tr>
<td>D</td>
<td>User chr 5</td>
<td>!</td>
<td>M</td>
<td>m</td>
<td>d</td>
<td>1</td>
<td>3</td>
<td>8</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>E</td>
<td>User chr 6</td>
<td>!</td>
<td>N</td>
<td>n</td>
<td>h</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
<td>Z</td>
</tr>
<tr>
<td>F</td>
<td>User chr 7</td>
<td>!</td>
<td>7</td>
<td>0</td>
<td>L</td>
<td>L</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>A</td>
</tr>
</tbody>
</table>

**Figure B2: QTP 12-C2 Characters table**
<table>
<thead>
<tr>
<th>L</th>
<th>00</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>A0</th>
<th>B0</th>
<th>C0</th>
<th>D0</th>
<th>E0</th>
<th>F0</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>DDF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>I A B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>L 8 R</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>B 3 C S</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>G 4 D T</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>X 5 E U</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>S 6 F V</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>R 7 T N W</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>C O N X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>09</td>
<td>V A I Y</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0A</td>
<td>X : J Z</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0B</td>
<td>+ : K E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0C</td>
<td>&lt; : L A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0D</td>
<td>- = M J</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0E</td>
<td>. &gt; N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0F</td>
<td>/ ? O</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure B3: QTP 12-GF2 Minifont in Graphic Mode Characters Table**
<table>
<thead>
<tr>
<th>L</th>
<th>H</th>
<th>00</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>A0</th>
<th>B0</th>
<th>C0</th>
<th>D0</th>
<th>E0</th>
<th>F0</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td></td>
<td>1</td>
<td>0</td>
<td>A</td>
<td>P</td>
<td>\</td>
<td>P</td>
<td>æ</td>
<td>E</td>
<td>@</td>
<td>9</td>
<td>é</td>
<td>@</td>
<td>P</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td></td>
<td>1</td>
<td>!</td>
<td>1</td>
<td>A</td>
<td>Q</td>
<td>a</td>
<td>ą</td>
<td>A</td>
<td>2</td>
<td>7</td>
<td>4</td>
<td>a</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>02</td>
<td></td>
<td>2</td>
<td>B</td>
<td>r</td>
<td>b</td>
<td>A</td>
<td>רי</td>
<td>י</td>
<td>י</td>
<td>8</td>
<td>O</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>03</td>
<td></td>
<td>#</td>
<td>3</td>
<td>C</td>
<td>S</td>
<td>c</td>
<td>ą</td>
<td>r</td>
<td>Ø</td>
<td>1</td>
<td>8</td>
<td>7</td>
<td>E</td>
<td>È</td>
<td>7</td>
<td>E</td>
<td>8</td>
</tr>
<tr>
<td>04</td>
<td></td>
<td>$</td>
<td>4</td>
<td>D</td>
<td>T</td>
<td>d</td>
<td>t</td>
<td>đ</td>
<td>đ</td>
<td>đ</td>
<td>đ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>05</td>
<td></td>
<td>%</td>
<td>5</td>
<td>E</td>
<td>u</td>
<td>e</td>
<td>u</td>
<td>E</td>
<td>O</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>O</td>
<td>0</td>
<td>1</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>06</td>
<td></td>
<td>6</td>
<td>6</td>
<td>F</td>
<td>V</td>
<td>v</td>
<td>d</td>
<td>đ</td>
<td>đ</td>
<td>đ</td>
<td>đ</td>
<td>đ</td>
<td>đ</td>
<td>đ</td>
<td>đ</td>
<td>đ</td>
<td></td>
</tr>
<tr>
<td>07</td>
<td></td>
<td>7</td>
<td>G</td>
<td>W</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
<td>e</td>
</tr>
<tr>
<td>08</td>
<td></td>
<td>1</td>
<td>8</td>
<td>H</td>
<td>k</td>
<td>h</td>
<td>δ</td>
<td>1</td>
<td>i</td>
<td>k</td>
<td>k</td>
<td>iculture</td>
<td>1</td>
<td>1</td>
<td>c</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>09</td>
<td></td>
<td>9</td>
<td>I</td>
<td>Y</td>
<td>i</td>
<td>y</td>
<td>φ</td>
<td>T</td>
<td>e</td>
<td>l</td>
<td>e</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0A</td>
<td></td>
<td>C</td>
<td>*</td>
<td>J</td>
<td>Z</td>
<td>j</td>
<td>ζ</td>
<td>ζ</td>
<td>ζ</td>
<td>ζ</td>
<td>ζ</td>
<td>ζ</td>
<td>ζ</td>
<td>ζ</td>
<td>ζ</td>
<td>ζ</td>
<td>ζ</td>
</tr>
<tr>
<td>0B</td>
<td></td>
<td>i</td>
<td>K</td>
<td>C</td>
<td>k</td>
<td>C</td>
<td>k</td>
<td>c</td>
<td>k</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
<td>c</td>
</tr>
<tr>
<td>0C</td>
<td></td>
<td>l</td>
<td>C</td>
<td>k</td>
<td>l</td>
<td>C</td>
<td>k</td>
<td>l</td>
<td>k</td>
<td>l</td>
<td>k</td>
<td>l</td>
<td>k</td>
<td>l</td>
<td>k</td>
<td>l</td>
<td>k</td>
</tr>
<tr>
<td>0D</td>
<td></td>
<td>m</td>
<td>8</td>
<td>M</td>
<td>j</td>
<td>m</td>
<td>j</td>
<td>j</td>
<td>j</td>
<td>j</td>
<td>j</td>
<td>j</td>
<td>j</td>
<td>j</td>
<td>j</td>
<td>j</td>
<td>j</td>
</tr>
<tr>
<td>0E</td>
<td></td>
<td>n</td>
<td>8</td>
<td>N</td>
<td>n</td>
<td>8</td>
<td>n</td>
<td>8</td>
<td>n</td>
<td>8</td>
<td>n</td>
<td>8</td>
<td>n</td>
<td>8</td>
<td>n</td>
<td>8</td>
<td>n</td>
</tr>
<tr>
<td>0F</td>
<td></td>
<td>?</td>
<td>?</td>
<td>O</td>
<td>L</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
</tr>
</tbody>
</table>

**Figure B4: QTP 12-GF2 Font Katakana in Graphic Mode Characters Table**
**Figure B5: QTP 12-GF2 Font European in Graphic Mode Characters Table**
APPENDIX C: MOUNTING OUTLINE DIMENSIONS

TERMINAL DIMENSIONS

Here are dimensions of terminal QTP 12 external metallic container and frontal plastic frame. Dimensions are in mm, scale is 1:1.

Dimensions of container only, area occupied can be slightly greater considering also mounting clamps and screws, up to a maximum of 156 x 72 x 80 mm (W x H x D).
SURFACE OR FLUSH PANEL MOUNTING

Surface or flush panel mounting is possible on any panel of maximum thickness 10 mm and clamping is done by two clamps provided with QTP 12.
Installation operations are extremely easy:

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

![Figure C2: Breaking for Installation](image)

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

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

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

5) screw the clamps until the QTP 12 is firmly docked to mounting panel;

6) connect the connectors.
INSERTION OF CUSTOMIZATION LABEL

Frontal of QTP 12 is provided with a pocket where the user can insert a label with one's logo, an identification code, purpose of activity LED, or more. Insert the label before mounting QTP. Label must be thin but rather rigid, for example made of 160 g/m² paper or polyester or polycarbonate. Here follow suggested dimensions, in millimeters; white zone is visible part:

![Figure C-3: Dimensions of customization label](image)

Here follow the operations to insert customization label in QTP 12.

1) Unscrew the two black screws of frontal panel (if present).
2) Separate the group back container + frame and the group keyboard + printed circuit. A simple pressure on QTP 12 connectors is enough.
3) Now keyboard is ready to insert the customization label; this latter must be inserted from the top side, using the specific window located on the back of keyboard panel, as shown in following figure. As can be seen on figure C-3, length of label must be greater than height of window to easy insertion and extraction.
4) Remount terminal QTP 12, following the previous indication in reversed order.

![Figure C-4: Customization label insertion](image)
FIXING FRONTAL PANEL TO CONTAINER

QTP 12 by default is provided with frontal panel (keyboard+printed circuit board) tongued in plastic frame of back container.
Terminal anyway allows a better mechanical docking by using two specific screws, to avoid accidental separations of frontal panel.

Here follow the operations to perform to assure such docking:

1) Separate the group back container + frame and the group keyboard + printed circuit.
   A simple pressure on QTP 12 connectors is enough.

2) Unscrew the 2 central screws to separate the frame from the rear carter.

3) On the front panel, in corrispondance to those 2 central screws, there are 2 countersink holes which are visible only in the rear view. To get these two holes accesible, the user needs to hole the polycarbonate panel covering.

4) Remount the panel by using the two screws of point "2". They will be screwed on the front panel instead of the frontal frame.
APPENDIX D: ALPHABETICAL INDEX

SYMBOLS
+VDC POW 10
.5VD 26
.ALIM 26
.CLOOP OPTION 5, 22
.RS422 OPTION 5, 22
.RS485 OPTION 5, 24
9TH BIT 30

A
ABSOLUTE CURSOR PLACEMENT, command 35
ADDS VIEW POINT 27
ALPHANUMERIC MODE SETTING, command 49
ALRM 4
AUTOREPEAT 4, 28

B
BACKLIGHT 6, 40
BACKSPACE, command 36
BASICCAN 5
BAUD RATE 27
BLINKING BLOCK CURSOR ON, command 37
BUFFER 28
BUFFER SIZE 8
BUZZER 6, 40

C
CAN 5, 18, 24
CAN TERMINATION 20, 24
CARRIAGE RETURN, command 35
CARRIAGE RETURN+LINE FEED, command 35
CCITT 12
CHARACTERS 9
CHARACTERS VISUALIZATION 33
CLEAR END OF LINE, command 36
CLEAR END OF PAGE, command 36
CLEAR LINE, command 36
CLEAR PAGE, command 36
COMMUNICATION MODE 30
COMMUNICATION TYPE 27
CONNECTORS 9
CN1 10
CN2 12
CN3 18
CONTRAST 20
CPU 8
CURRENT LOOP 5, 12, 15, 22
CURRENT LOOP NETWORK 16
CURSOR DOWN, command 34
CURSOR LEFT, command 34
CURSOR OFF, command 37
CURSOR RIGHT, command 34
CURSOR UP, command 34

D
DEFINITION AND MEMORIZATION OF USER CHARACTER, command 47
DEFINITION OF USER CHARACTER, command 47
DEL 4
DISPLAY 6, 8, 9

E
EEPROM 6, 27, 32, 38, 41, 42
EXPS-1 26
EXTERNAL CARDS 55

F
FLUORESCENT DISPLAY BACKLIGHT SETTING, command 40
FONT 54

G
GND 10
GRAPHIC AREA SETTING, command 50
GRAPHIC COMMANDS 48
GRAPHIC CURSOR ABSOLUTE POSITION, command 50
GRAPHIC FONT SETTING, command 54
GRAPHIC MODE SETTING, command 50

H
HOME, command 34

I
IDENTIFICATION 27
INFO 4
INS 4
J

JUMPERS  20

K

KEY CODES  29
KEY RECONFIGURATION, command  44
KEYBOARD  4
KEYBOARD ACQUISITION  28
KEYCLICK  4, 6, 27, 28, 44
KEYCLICK OFF WITH MEMORIZATION, command  45
KEYCLICK OFF WITHOUT MEMORIZATION, command  44
KEYCLICK ON WITH MEMORIZATION, command  45
KEYCLICK ON WITHOUT MEMORIZATION, command  44

L

LED AND BUZZER ACTIVATION, command  40
LEDS  5, 40
LOCAL SET UP  27

M

MASTER SLAVE  30
MESSAGE READING, command  42
MESSAGE STORAGE, command  41
MESSAGE VISUALIZATION, command  42
MESSAGES  6, 32, 41, 42
MOUNTING  9

O

OPERATING MODE SELECTION, command  39
OUTLINE  8

P

PELICAN  5
POLARITY  26
POWER SUPPLY  5, 9, 10, 26
PROTOCOL  8
R
READING OF PRESENCE BYTE, command 38
READING OF THE LAST STORAGED MESSAGE NUMBER, command 41
READING OF VERSION NUMBER, command 39
RELATIVE HUMIDITY 9
REQUEST FOR EEPROM WRITING POSSIBILITY, command 38
RESOURCES 8
RS 232 4, 5, 12, 22
RS 422 5, 12, 20, 22
RS 422-485 TERMINATION 9
RS 485 5, 12, 20, 24
RS 485 NETWORK 14
RV1 20
RV2 20

S
SCROLLING MESSAGES VISUALIZATION, command 43
SERIAL COMMUNICATION 5, 12
SERIAL LINE 22
SET UP 27
SIZE 8
START 4
STEADY STATIC CURSOR ON, command 37
STOP 4
STOP BIT 27
SURFACE OR FLUSH PANEL MOUNTING 9

T
TEMPERATURE RANGE 9
TRANSZORB™ 26
TRIMMER 20

U
USER CHARACTERS 33, 46

V
VAC 10

W
WEIGHT 9
WRITING OF PRESENCE BYTE, command 38