RKD LT
Remote Keyboard Display controller

TECHNICAL MANUAL
RKD LT
Remote Keyboard Display controller

TECHNICAL MANUAL

Intelligent video terminal for ABACO® intelligent bus; size 100x160 mm single EUROPE format; management of FUTABA 20x2 up to 40x4 characters alphanumeric displays; management of graphic LCD displays mounting HITACHI or TOSHIBA controller, 120x64 up to 240x128 pixels; on board LCD contrast regulation trimmer; management of a 56 keys BG keyboard or generic a 7x8 matrix keyboard, featuring Shift, Control, Caps Lock, Autorepeat and Keyclick functions; management of 8 external LEDs; on board buzzer for acoustic indications and keyclick; RS 232, 422 or Current Loop serial communication line; auxiliary RS 232 serial line to echo the received data; Baud Rate from 2400 up to 19200 baud; ABACO® BUS addressing space as low as 2 bytes; 8 pins Dip Switch to set the I/O address; up to 2KBytes of E’PROM to store messages; capability to store alphanumeric screens in EPROM for later visualization; one 8 pins Dip Switch and one 4 pins Dip Switch for set-up and configuration; only one + 5Vdc power supply; 130 mA (without display and keyboard); optionally available with GDOS 80 operating system already installed to let the User develop his/her own management programs.
No part of this document may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, either electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written consent of grifo®.

IMPORTANT

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

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

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

SYMBOLS DESCRIPTION

In the manual could appear the following symbols:

- Attention: Generic danger
- Attention: High voltage

Trade Marks

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

INTRODUCTION ........................................................................................................................ 1
CARD VERSION AND FIRMWARE ........................................................................................ 1

GENERAL FEATURES...............................................................................................................2
COMMUNICATION TO CONTROL UNIT ........................................................................ 4
ABACO® BUS PARALLEL INTERFACE ............................................................................ 4
MAIN SERIAL LINE .............................................................................................................. 4
AUXILIARY SERIAL LINE .................................................................................................. 4
KEYBOARD ............................................................................................................................ 4
BUZZER ................................................................................................................................... 5
FLUORESCENT ALPHANUMERIC DISPLAYS ............................................................. 5
LCD GRAPHIC DISPLAYS .................................................................................................. 5
EXTERNAL LEDS .................................................................................................................. 5
EEPROM.................................................................................................................................. 5
FIRMWARE TRKDGL .......................................................................................................... 5
FIRMWARE TRKDAF ........................................................................................................... 6

TECHNICAL FEATURES .......................................................................................................... 8
GENERAL FEATURES .......................................................................................................... 8
PHISICAL FEATURES .......................................................................................................... 9
ELETCTRIC FEATURES ...................................................................................................... 9

INSTALLATION ........................................................................................................................ 10
CONNECTIONS ................................................................................................................... 10
CN1 - EXTERNAL LEDS CONNECTOR ..................................................................... 10
CN2 - CONNECTOR FOR BG KEYBOARD ............................................................... 12
CN3 - SERIAL LINES CONNECTOR ........................................................................ 16
J5 - COMPLEMENTAR CONNECTOR FOR SERIAL LINES ........................................ 19
J1 - CONNECTOR FOR FLUORESCENT ALPHNUMERIC DISPLAY .................... 20
J2 - CONNECTOR FOR GRAPHIC LCD DISPLAYS ................................................ 21
K1 - CONNECTOR FOR ABACO® BUS ................................................................. 22
JUMPERS .............................................................................................................................. 24
5 PINS JUMPERS ............................................................................................................ 24
3 PINS JUMPERS ............................................................................................................ 26
2 PINS JUMPERS ............................................................................................................ 27
GRAPHIC LCD CONTRAST REGULATION............................................................. 27
SERIAL COMMUNICATION ............................................................................................. 28

PARALLEL INTERFACE DESCRIPTION ............................................................................ 30
INTRODUCTION ................................................................................................................... 30
BOARD MAPPING ........................................................................................................... 30
PARALLEL COMMUNICATION REGISTERS .................................................................... 31
PARALLEL COMMUNICATION MANAGEMENT ......................................................... 32
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOFTWARE DESCRIPTION</td>
<td>33</td>
</tr>
<tr>
<td>BOARD CONFIGURATION</td>
<td>33</td>
</tr>
<tr>
<td>CONFIGURATION USING TRKDGL FIRMWARE</td>
<td>33</td>
</tr>
<tr>
<td>CONFIGURATION USING TRKDAF FIRMWARE</td>
<td>36</td>
</tr>
<tr>
<td>RECEPTION BUFFER</td>
<td>38</td>
</tr>
<tr>
<td>KEYBOARD ACQUISITION</td>
<td>38</td>
</tr>
<tr>
<td>CHARACTERS REPRESENTATION</td>
<td>40</td>
</tr>
<tr>
<td>CURSOR POSITIONING COMMANDS</td>
<td>40</td>
</tr>
<tr>
<td>CURSOR LEFT</td>
<td>40</td>
</tr>
<tr>
<td>CURSOR RIGHT</td>
<td>41</td>
</tr>
<tr>
<td>CURSOR DOWN</td>
<td>41</td>
</tr>
<tr>
<td>CURSOR UP</td>
<td>41</td>
</tr>
<tr>
<td>CARRIAGE RETURN</td>
<td>41</td>
</tr>
<tr>
<td>CARRIAGE RETURN + LINE FEED</td>
<td>42</td>
</tr>
<tr>
<td>HOME</td>
<td>42</td>
</tr>
<tr>
<td>CURSOR ABSOLUT POSITIONING</td>
<td>42</td>
</tr>
<tr>
<td>CHARACTERS DELETION COMMANDS</td>
<td>43</td>
</tr>
<tr>
<td>CLEAR PAGE</td>
<td>43</td>
</tr>
<tr>
<td>CLEAR END OF LINE</td>
<td>43</td>
</tr>
<tr>
<td>CURSOR ATTRIBUTES MANAGEMENT COMMANDS</td>
<td>44</td>
</tr>
<tr>
<td>CURSOR DISABLED</td>
<td>44</td>
</tr>
<tr>
<td>CURSOR ENABLED AND FIXED</td>
<td>44</td>
</tr>
<tr>
<td>CURSOR ENABLED AND FLASHING</td>
<td>44</td>
</tr>
<tr>
<td>FLASHING BLOCK CURSOR</td>
<td>44</td>
</tr>
<tr>
<td>AUXILIARY SERIAL PORT MANAGEMENT COMMANDS</td>
<td>45</td>
</tr>
<tr>
<td>ECHO ACTIVATION ON AUXILIARY SERIAL PORT</td>
<td>45</td>
</tr>
<tr>
<td>ECHO DEACTIVATION ON AUXILIARY SERIAL PORT</td>
<td>45</td>
</tr>
<tr>
<td>CHARACTERS ATTRIBUTES MANAGEMENT COMMANDS</td>
<td>46</td>
</tr>
<tr>
<td>REVERSE ATTRIBUTE ACTIVATION</td>
<td>46</td>
</tr>
<tr>
<td>REVERSE ATTRIBUTE DEACTIVATION</td>
<td>46</td>
</tr>
<tr>
<td>KEYBOARD MANAGEMENT COMMANDS</td>
<td>47</td>
</tr>
<tr>
<td>KEYCLICK ON LOCAL BUZZER ACTIVATION</td>
<td>47</td>
</tr>
<tr>
<td>KEYCLICK ON LOCAL BUZZER DEACTIVATION</td>
<td>47</td>
</tr>
<tr>
<td>KEYCLICK ON BG KEYBOARD ACTIVATION</td>
<td>47</td>
</tr>
<tr>
<td>KEYCLICK ON BG KEYBOARD DEACTIVATION</td>
<td>47</td>
</tr>
<tr>
<td>EXTERNAL LEDS MANAGEMENT COMMANDS</td>
<td>48</td>
</tr>
<tr>
<td>EXTERNAL LEDS ACTIVATION</td>
<td>48</td>
</tr>
<tr>
<td>EXTERNAL LEDS REVERSED ACTIVATION</td>
<td>48</td>
</tr>
<tr>
<td>ALPHANUMERIC SCREENSHOTS MANAGEMENT COMMANDS</td>
<td>49</td>
</tr>
<tr>
<td>SCREENSHOT VISUALIZATION</td>
<td>49</td>
</tr>
<tr>
<td>SCREENSHOT MEMORIZATION IN EEPROM</td>
<td>50</td>
</tr>
<tr>
<td>SCREENSHOT VISUALIZATION FROM EEPROM</td>
<td>50</td>
</tr>
<tr>
<td>SCREENSHOT READ FROM EEPROM</td>
<td>51</td>
</tr>
<tr>
<td>GRAPHIC MODE COMMANDS</td>
<td>51</td>
</tr>
<tr>
<td>REVERSE ATTRIBUTE SETTING</td>
<td>51</td>
</tr>
<tr>
<td>CHARACTERS ZOOM SETTING</td>
<td>51</td>
</tr>
<tr>
<td>CHARACTERS ELONGATION SETTING</td>
<td>52</td>
</tr>
<tr>
<td>CHARACTER CURSOR PLACEMENT</td>
<td>52</td>
</tr>
<tr>
<td>ABSOLUTE CURSOR PLACEMENT</td>
<td>52</td>
</tr>
</tbody>
</table>
GENERAL INDEX

FIGURE 1: CARD PHOTO ...................................................................................................................................... 3
FIGURE 2: COMPONENTS MAP ..................................................................................................................... 7
FIGURE 3: CN1 - EXTERNAL LEDS CONNECTOR .......................................................................................... 10
FIGURE 4: COMMON CATHOD EXTERNAL LEDS EXAMPLE CONNECTION DIAGRAM .................................. 11
FIGURE 5: CN2 - CONNECTOR FOR BG KEYBOARD .................................................................................. 12
FIGURE 6: BG KEYBOARD ELECTRIC DIAGRAM ....................................................................................... 13
FIGURE 7: FRONT VIEW OF BG KEYBOARD ............................................................................................ 14
FIGURE 8: CONNECTORS, DIP SWITCH, TRIMMER, ETC. LOCATION .......................................................... 15
FIGURE 9: CONNECTOR FOR SERIAL LINES ............................................................................................... 16
FIGURE 10: RS 232 EXAMPLE CONNECTION DIAGRAM ........................................................................... 17
FIGURE 11: RS 422 EXAMPLE CONNECTION DIAGRAM ........................................................................... 17
FIGURE 12: 4 WIRES CURRENT LOOP EXAMPLE CONNECTION DIAGRAM .......................................... 18
FIGURE 13: J5 - COMPLEMENTAR SERIAL LINES CONNECTOR ................................................................. 19
FIGURE 14: J1 - CONNECTOR FOR ALPHANUMERIC FLUORESCENT DISPLAY ........................................ 20
FIGURE 15: J2 - CONNECTOR FOR GRAPHIC LCD DISPLAY .................................................................... 21
FIGURE 16: J3 - CONNECTOR FOR ABACO® BUS .................................................................................. 22
FIGURE 17: JUMPERS SUMMARIZING TABLE ......................................................................................... 24
FIGURE 18: 5 PINS JUMPERS TABLE .................................................................................................... 24
FIGURE 19: JUMPERS LOCATION ............................................................................................................... 25
FIGURE 20: 3 PINS JUMPERS TABLE .................................................................................................... 26
FIGURE 21: 2 PINS JUMPERS TABLE .................................................................................................... 27
FIGURE 22: MAIN SERIAL LINE AS RS 422, AUXILIARY SERIAL LINE RS 232 ........................................... 28
FIGURE 23: MAIN SERIAL LINE AS CURRENT LOOP, AUXILIARY SERIAL LINE RS 232 ....................... 28
FIGURE 24: MAIN SERIAL LINE AS RS 232, AUXILIARY SERIAL LINE RS 232 ........................................ 29
FIGURE 25: MAIN SERIAL LINE AS RS 232, AUXILIARY SERIAL LINE CURRENT LOOP ....................... 29
FIGURE 26: MAIN SERIAL LINE AS RS 232, AUXILIARY SERIAL LINE RS 232 ........................................ 29
FIGURE 27: PARALLEL COMMUNICATION ADDRESSES TABLE ............................................................ 31
FIGURE 28: TRKDGL DIP SWITCH CONFIGURATION TABLE .................................................................. 33
FIGURE 29: GRAPHIC LCD MODEL SELECTION TABLE ......................................................................... 34
FIGURE 30: TRKDGL MAIN SERIAL LINE BAUDE RATE SELECTION TABLE .......................................... 35
FIGURE 31: TRKDGL AUXILIARY SERIAL LINE BAUDE RATE SELECTION TABLE ............................... 35
FIGURE 32: TRKDAF DIP SWITCH CONFIGURATION TABLE .................................................................... 36
FIGURE 33: FUTABA FLUORESCENT DISPLAY SELECTION TABLE ....................................................... 36
FIGURE 34: TRKDAF MAIN SERIAL LINE BAUDE RATE SELECTION TABLE ........................................... 37
FIGURE 35: TRKDAF AUXILIARY SERIAL LINE BAUDE RATE SELECTION TABLE ............................... 37
FIGURE 36: SPECIAL KEYS CODE TABLE ............................................................................................... 38
FIGURE 37: ALPHANUMERIC KEYS CODE TABLE .................................................................................. 39
FIGURE 38: NUMBER OF SCREENSHOTS STORABLE IN EEPROM ....................................................... 50
FIGURE 39: ARCS NUMERATION AND DISPOSITION ............................................................................... 55
FIGURE 40: TOP ORIENTED ARROW ....................................................................................................... 56
FIGURE 41: BOTTOM ORIENTED ARROW .............................................................................................. 57
FIGURE 42: RIGHT ORIENTED ARROW .................................................................................................. 57
FIGURE 43: LEFT ORIENTED ARROW ..................................................................................................... 58
FIGURE 44: TOP ORIENTED FILLED ARROW .......................................................................................... 58
FIGURE 45: BOTTOM ORIENTED FILLED ARROW ................................................................................ 59
FIGURE 46: RIGHT ORIENTED FILLED ARROW ................................................................. 59
FIGURE 47: LEFT ORIENTED FILLED ARROW ............................................................... 60
FIGURE 48: POSSIBLE CONNECTIONS DIAGRAM ....................................................... 65
FIGURE A1: TRKDAF COMMANDS (Adds Viewpoint) Table .................................... A-1
FIGURE A2: TRKDAF COMMANDS (TVI 950) Table .................................................. A-2
FIGURE A3: TRKDG1 COMMANDS FOR ALPHANUMERIC MODE (Adds Viewpoint) Table .... A-3
FIGURE A4: TRKDG1 COMMANDS FOR ALPHANUMERIC MODE (TVI 950) Table .......... A-4
FIGURE A5: TRKDG1 COMMANDS FOR GRAPHIC MODE Table 1 .............................. A-5
FIGURE A6: TRKDG1 COMMANDS FOR GRAPHIC MODE Table 2 .............................. A-6
INTRODUCTION

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

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

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

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

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

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

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

CARD VERSION AND FIRMWARE

The present handbook is reported:

- Card RKD LT: 181093 and later
- Firmware TRKDGL: 2.1 and later
- Firmware TRKDAF: 1.1 and later

The validity of the bring informations is subordinate to the number of the card release. The user must always verify the correct correspondence among the two denotations. On the card the release number is present in more points both board printed diagram (serigraph) and printed circuit (for example along the border on the component side, near trimmer R27); the firmware version is reported in the label on the EPROM.
GENERAL FEATURES

RKD LT (Remote Keyboard Display controller) is an interesting peripheral intelligent board, size 100x160 mm standard EUROPE format, designed to solve efficiently and economically the problem of user interfacing in the industrial sector.

The board is complete video terminal, capable to manage in total autonomy both the acquisition and the visualization of informations.

Communication between RKD LT and its master unit can be serial or parallel, in order to satisfy every need for speed and suitability. These features allow any card provided with a serial interface and any ABACO® BUS control card (GPC® serie) to interface to the RKD LT intelligent controller producing the remarkable advantage not to have to develop specific management software, reducing so code space occupation and development time, and increasing the execution speed.

RKD LT can interface to FUTABA fluorescent alphanumeric displays capable of resolutions from 20x2 up to 40x4 characters, or LCD graphic displays mounting HITACHI or TOSHIBA controller, capable of resolutions from 120x64 up to 240x128 pixels.

The board can manage directly the 56 keys BG keyboard, or a generic 7x8 keys matrix keyboard, featuring the Shift, Caps Lock, Control, Autorepeat and Keyclick functions, in order to make more comfortable and easy the phase of data input.

Up to 8 external LEDs can be connected to the RKD LT board to provide visual signalations such as alarms, machine states, etc.

RKD LT board is capable to solve easily all the difficulties relative to messages visualization, graphic displays, visual and acoustic signalations, keyboard acquisition, hardcopy, etc.

- Intelligent video terminal for ABACO® BUS
- Size 100x160 mm single EUROPE format
- High speed parallel communication or serial communication
- Management of FUTABA alphanumeric displays 20x2 up to 40x4 characters
- Management of graphic LCD displays mounting HITACHI or TOSHIBA controller, 120x64 up to 240x128 pixels
- On board LCD contrast regulator trimmer
- Management of a 56 keys BG keyboard or a generic 7x8 keys matrix keyboard, featuring Shift, Caps Lock, Control, Autorepeat and Keyclick functions
- Management of 8 external LEDs
- On board buzzer for acoustic indications and keyclick
- RS 232, RS 422 or Current Loop serial communication line
- Auxiliary RS 232 serial line to echo the received data
- Baud Rate from 2400 up to 19200 baud
- ABACO® BUS addressing space as low as 2 bytes
- 8 pins Dip Switch to set the I/O address
- Up to 2KBytes of EEPROM to store messages, parameters, etc.
- Capability to store alphanumeric screens in EPROM for later visualization
- One 8 pins Dip Switch and one 4 pins Dip Switch for set-up and configuration
- Only one + 5Vdc power supply; 130 mA (without display and keyboard)
- Optionally available with GDOS 80 operating system already installed to let the User develop his/her own management programs
FIGURE 1: CARD PHOTO
COMMUNICATION TO CONTROL UNIT

Communication between RKD LT and its master unit can be serial or by the ABACO® BUS, a high speed parallel data path, in order to satisfy every need for communication velocity and hardware needs.

ABACO® BUS PARALLEL INTERFACE

RKD LT board is provided with an ABACO® BUS parallel interface capable to speed up all the communication operations. This section manages the communication between the on board CPU and the master control card (GPC® serie); in detail it manages the I/O mapping of the board in the addressing space by simply acting on a 8 pins Dip Switch, the board takes only two bytes of space. The ABACO® BUS is designed to work in the industrial automation environment providing an 8 bits data path with an 8 bits address word.

MAIN SERIAL LINE

RKD LT board is provided with a serial communication line which can be buffered as RS 232 or it can be buffered as current loop or RS 422 provided that the auxiliary serial line is kept as RS 232. The communication parameters are: 8 bits, no parity, 1 stop bit, the BaudRate is selectable amongst 2400, 4800, 9600, 19200 Baud. The User can set this last parameter by acting on the proper Dip Switches.

AUXILIARY SERIAL LINE

RKD LT board is provided with an auxiliary serial communication line which can be buffered as RS 232 or it can be buffered as current loop or RS 422 provided that the main serial line is kept as RS 232. This line can perform received data echo functions, making possible harcopy operations on a different serial device, like a printer, that the board can drive directly. The communication parameters are: 8 bits, no parity, 1 stop bit, the baud rate is selectable amongst 2400, 4800, 9600, 19200 Baud. The User can set this last parameter by acting on the proper Dip Switches.

KEYBOARD

RKD LT board is able to manage directly a 56 keys BG keyboard or a 7x8 keys generic matrix keyboard; the keys are acquired by the board itself with AutoRepeat, Shift, Control and Caps-Lock functions performing. It is also possible to disable the KeyClick function, that is the audio feedback, made by activating the on board buzzer, when a key is pressed.
BUZZER

The RKD LT board is provided with a circuitry able to utter a constant sound employing a capacitiva buzzer. This circuitry can be software enabled to perform beeps or, using the KeyClick function, can be enabled when a key is hit.

FLUORESCENT ALPHANUMERIC DISPLAYS

The RKD LT board can manage FUTABA fluorescent alphanumeric displays. In detail it can manage displays with rows made of 20 or 40 characters: that is the 20x2, 20x4, 40x1, 40x2 and 40x4 models.

LCD GRAPHIC DISPLAYS

The RKD LT board can manage graphic displays mounting a HITACHI or TOSHIBA controller, both in alphanumeric and in graphic mode. In detail these models can be connected: 120x64 pixels, corresponding to 15x8 characters; 128x128 pixels (16x16 characters); 240x64 pixels (40x8 characters); 160x128 pixels (20x16 characters); 240x128 pixels (40x16 characters). Contrast regulation is made by an on board circuit provided with a trimmer to perform this operation.

EXTERNAL LEDS

The RKD LT board can manage 8 external LEDs, in the common cathod connection. The LEDs can be used to make visual signalations like alarms, showing machine status, etc. The same lines may also be used for a low-level management of a seven segments display.

EEPROM

The RKD LT board is provided with an EEPROM, whose size ranges from 512 Bytes to 2K Bytes, to store messages. The main purpose of this section is to give a non volatile memory device where to store informations even when the master unit can't perform this feature.

FIRMWARE TRKDGL

It is the terminal emulation firmware, able to run both in graphic and alphanumeric mode. It performs the scansion and the debouncing on the BG keyboard; every key pressed is recognized, coded and transmitted to the control system by the desired communication mode (serial or parallel). In addition the special keys are managed in autonomy, assigning them their functions, also the AutoRepeat, Shift, Control, Caps-Lock and Keyclick functions are managed.
Command and/or data received by the control system are interpreted and executed interacting with the display; remarkable commands are: cursor positioning, lines, rectangles and circumferences drawing, clear display, graphic zoom of characters, etc.

In alphanumeric mode, the **RKD LT** board can emulate the **TVI 950** or **Adds-ViewPoint** standard terminal protocols.

A 64KBytes EPROM area is reserved by the firmware to store alphanumeric screens that need to be displayed very quickly, this allows also a remarkable memory saving for the master control unit.

To create, correct and generate the alphanumeric screens to be displayed by **RKD LT**, **grifo®** has realized a powerful text editor running on P. C., called **RKD_EDIT**, available both in Italian and in English. The software is also able to program directly the EPROM on **RKD LT**.

The User interface is based on scroll-down menus and the employ of the mouse, making possible to write the text quickly and comfortably.

For further informations please refer to the next chapters.

---

**FIRMWARE TRKDAF**

The alphanumeric terminal emulatore designed to work with fluorescent displays.

It performs the scansion and the debouncing on the BG keyboard; every key pressed is recognized, coded and transmitted to the control system by the desired communication mode (serial or parallel).

In addition the special keys are managed in autonomy, assigning them their functions, also the AutoRepeat, Shift, Control, Caps-Lock and KeyClick functions are managed.

Command and/or data received by the control system are interpreted and executed interacting with the display; remarkable commands are: cursor positioning, clear line, turn off or on cursor, etc.

The **RKD LT** board can emulate the **TVI 950** or **Adds-ViewPoint** standard terminal protocols.

Also this firmware reserves a 64KBytes EPROM area to store alphanumeric screens that need to be displayed very quickly, this allows also a remarkable memory saving for the master control unit.

Also for this firmware it is possible to create quickly the text screenshots using **RKD_EDIT**, as described in the previous paragraph.

For further informations please refer to the next chapters.
FIGURE 2: COMPONENTS MAP
TECHNICAL FEATURES

GENERAL FEATURES

Devices:
- Interface for 56 keys BG Keyboard, or generic 7x8 matrix keyboard.
- Interface for graphic LCD display mounting TOSHIBA or HITACHI controller, with trimmer to regulate the contrast.
- Interface for FUTABA fluorescent alphanumeric display.
- Interface to drive directly 8 external LEDs.
- Buzzer for BELL signalations or audio feedback of a key pressed.
- 2 Full Duplex RS 232, RS 422 or Current-Loop serial lines.
- EEPROM to store messages, parameters, etc. Max 2 Kbytes.
- 2 dip switch (11 total pins) for the board configuration.
- Parallel interface for ABACO® BUS.
- 8 pins Dip switch to set the I/O address.

On board CPU: Z180 with 18.432 MHz quartz.

Parallel Interface:
8 bit BUS for data and address.
256 byte total addressing space.
2 byte of I/O space taken.

Serial Communication:
BAUD RATE: 2400, 4800, 9600 or 19200 Baud.
1 Stop Bit.
No Parity.
8 Bit.

Managed Displays: Alphanumeric fluorescent displays
Futaba M202SD01BA and compatible - 20x2 characters “small”
Futaba M202SD08GK and compatible - 20x2 characters “wide”
Futaba M204SD01AA and compatible - 20x4 characters
Futaba M405SD04GJ and compatible - 40x1 characters
Futaba M402SD07GK and compatible - 40x2 characters
Futaba M404SD01BA and compatible - 40x4 characters (+IAF 404)

LCD graphic displays mounting Toshiba T6963C controller
Toshiba TLX-1021 and compatible - 120x64 pixels
Toshiba TLX-711A and compatible - 240x64 pixels
Toshiba TLX-1391 and compatible - 128x128 pixels
Toshiba TLX-1013 and compatible - 160x128 pixels
Toshiba TLX-1301 and compatible - 240x128 pixels
Toshiba TLX-1091 and compatible - 240x128 pixels

LCD graphic displays mounting Hitachi HD61830B controller
Varitronix MGLS240128V2 and compatible - 240x128 pixels
PHISICAL FEATURES

Size: 100x160 mm standard EUROPE format.

Weight: 185 g

Temperature range: From 0 to 70 centigrad degrees

Relative Humidity: 20% up to 90% (without condens).

Connectors:
- J1: 20 pins, male, vertical, low profile connector
- J2: 20 pins, male, vertical, low profile connector
- J3: DIN 41612 64 pins M 90 degrees A+C type C
- J5: 16 pins, male, vertical, low profile connector
- CN1: 10 pins, male, 90 degrees, low profile connector
- CN2: 20 pins, male, 90 degrees, low profile connector
- CN3: 16 pins, male, 90 degrees, low profile connector

ELETCTRIC FEATURES

Power Supply Voltage: +5 Vdc.

Consumption: 130 mA without keyboard and display.

RS 422 termination circuit:
- Line termination resistance = 120 Ω
- Positive pin pull-up resistance = not present
- Negative pin pull-up resistance = not present

Serial Resistance for LEDs: 330 Ω

Voltage range for LCD contrast: -6 ÷ -12 Vdc or -7.5 ÷ -15 Vdc
INSTALLATION

In this chapter there are all information for a right installation and correct use of the card. The User can find the location and functions of each connectors, LEDs, jumpers and some explanatory diagram.

CONNECTIONS

The **RKD LT** module has seven connectors that can be linked to other devices or directly to the field, according to system requirements. In this paragraph there are connectors pin out, a brief signals description (including the signal direction), connectors location and some electrical diagrams that show the on board circuit of each connector.

CN1 - EXTERNAL LEDS CONNECTOR

Connector CN1, used to connect 8 external LEDs to the board is a 2.54 mm pitch, 10 pins, male, 90 degrees, low profile connector. Here follows its pinout and an example of connection for external LEDs.

The connection mode must be **common cathod** as shown in figure 4.

**Figure 3: CN1 - External LEDs connector**

Symbols description:

<table>
<thead>
<tr>
<th>LED n</th>
<th>=</th>
<th>O</th>
</tr>
</thead>
<tbody>
<tr>
<td>GND</td>
<td>=</td>
<td>- LEDs common cathod ground signal</td>
</tr>
<tr>
<td>+5 Vdc</td>
<td>=</td>
<td>O</td>
</tr>
</tbody>
</table>
FIGURE 4: COMMON CATHOD external LEDs example connection diagram
CN2 - CONNECTOR FOR BG KEYBOARD

The CN2 connector, dedicated to connect the BG keyboard, is a 2.54 mm pitch, 20 pins, male, 90 degrees, low profile connector. Here follows it pin-out, a frontal view of the keyboard and the keyboard’s electric diagram, to ease the User in designing his/her own 7x8 matrix keyboard.

**FIGURE 5: CN2 - CONNECTOR FOR BG KEYBOARD**

Symbols description:

- **ROW n** = I - Input signal to connect the n-th row of 7x8 matrix
- **COLUMN n** = O - Output signal to connect the n-th column of 7x8 matrix
- **LED** = O - Output signal to manage the LED of BG keyboard
- **BUZZER** = O - Output signal to manage the buzzer of BG keyboard
- **Reserved** = - Reserved signal, must be **not connected**
- **GND** = - Ground signal
FIGURE 6: BG KEYBOARD ELECTRIC DIAGRAM
**FIGURE 7: FRONT VIEW OF BG KEYBOARD**
FIGURE 8: CONNECTORS, DIP SWITCH, TRIMMER, ETC. LOCATION
CN3 - SERIAL LINES CONNECTOR

The connector used for serial lines, called CN3, is a 2.54 mm pitch, 16 pins, male, 90 degrees, low profile connector. Here follows the pin out of signals, that has been designed to minimize interferences and to easy the connection to the field, while the signals respect the CCITT rules relative to each of the communication standard used.

**Symbols description:**

- **TxDA - TxDB** = O - Transmit Data: transmit data signal for main (A) and auxiliary (B) RS 232 serial line
- **RxDA - RxDB** = I - Transmit Data: transmit data signal for main (A) and auxiliary (B) RS 232 serial line
- **CTSA - CTSB** = I - Clear To Send: clear to send signal for main (A) and auxiliary (B) RS 232 serial line
- **RTSA** = O - Request To Send: request to send signal for main (A) RS 232 serial line (not present on auxiliary serial line)
- **RX- RS422 / RX- C.L.** = I - Receive Data Negative: Main and auxiliary, RS 422 or Current Loop serial line, reception negative bipolar signal
- **RX+ RS422 / RX+ C.L.** = I - Receive Data Positive: Main and auxiliary, RS 422 or Current Loop serial line, transmission positive bipolar signal
- **TX- RS422 / TX- C.L.** = O - Transmit Data Negative: Main and auxiliary, RS 422 or Current Loop serial line, transmission negative bipolar signal
- **TX+ RS422 / TX+ C.L.** = O - Transmit Data Positive: Main and auxiliary, RS 422 or Current Loop serial line, transmission positive bipolar signal
- **GND** = - Ground signal
- **N.C.** = - Not connected
FIGURE 10: RS 232 EXAMPLE CONNECTION DIAGRAM

N.B.
In this diagram the reported RS 232 connection employs the RTSA handshake signal. This signal is managed by the RKD LT board using the policy shown forward. However it is possible to communicate to the board without problems also by a master control unit that doesn't support such signal.

FIGURE 11: RS 422 EXAMPLE CONNECTION DIAGRAM
This diagram shows the voltage to supply the ring (VCL) and the current limitation resistors (R).
The value of power supply voltage changes according to the number of connected devices, this
because the maximum dissipated power for each device must be warranted to be at most \( 125 \text{ mW} \)
for the transmitter and \( 90 \text{ mW} \) for the receiver when the maximum current allowed (20 mA)
circulates.
The resistor R is needed to limit the current when the line is short-circuited; typically for \( VCL=5\text{Vdc} \)
the value of R is \( 220 \Omega \).
For further informations please refer to the HEWLETT-PACKARD Data-Book, in the sections
about \texttt{HCPL 4100} and \texttt{HCPL 4200} Current Loop opto-couplers.
J5 - COMPLEMENTAR CONNECTOR FOR SERIAL LINES

The complementar connector for serial lines, called J5, is a 2.54 mm pitch, 16 pins, male, vertical, low profile connector. Its pin out is complementar respect to CN3 standard connector; this is made to allow a straight connection to all the control boards that follow this pin out (such as, for example, GPC® 011, GPC® 15A, GPC® 150, GPC® 188F, etc.) by simply using a 16 pins flat cable.

Symbols description:

- **TxDA - TxDB** = O - Transmit Data: transmit data signal for main (A) and auxiliary (B) RS 232 serial line
- **RxDA - RxDB** = I - Transmit Data: transmit data signal for main (A) and auxiliary (B) RS 232 serial line
- **CTSA - CTSB** = I - Clear To Send: clear to send signal for main (A) and auxiliary (B) RS 232 serial line
- **RTSA** = O - Request To Send: request to send signal for main (A) RS 232 serial line (not present on auxiliary serial line)
- **RX- RS422 / RX- C.L.** = I - Receive Data Negative: Main and auxiliary, RS 422 or Current Loop serial line, reception negative bipolar signal
- **RX+ RS422 / RX+ C.L.** = I - Receive Data Positive: Main and auxiliary, RS 422 or Current Loop serial line, reception positive bipolar signal
- **TX- RS422 / TX- C.L.** = O - Transmit Data Negative: Main and auxiliary, RS 422 or Current Loop serial line, transmission negative bipolar signal
- **TX+ RS422 / TX+ C.L.** = O - Transmit Data Positive: Main and auxiliary, RS 422 or Current Loop serial line, transmission positive bipolar signal
- **GND** = - Ground signal
- **N.C.** = - Not connected

[FIGURE 13: J5 - COMPLEMENTAR SERIAL LINES CONNECTOR]
J1 - CONNECTOR FOR FLUORESCENT ALPHANUMERIC DISPLAY

The connector for FUTABA alphanumeric fluorescent displays, called J1, is a 2.54 mm pitch, 20 pins, male, vertical, low profile connector. It allows a quick connection to the display by using a common 20 pins flat cable. Here follows the pin out.

Please remark that the power for the display is supplied directly by the RKD LT board.

**FIGURE 14: J1 - CONNECTOR FOR ALPHANUMERIC FLUORESCENT DISPLAY**

Symbols description:

- **D0 - D7** = O - D0+D7 data signals
- **/WR** = O - Write signal
- **/RD** = O - Read signal
- **/SEL** = O - Device selection signal
- **/TEST** = O - Test mode activation signal (must be always connected to +5 Vdc)
- **BUSY** = I - Busy signal
- **+5 Vdc** = O - Power supply for the display
- **GND** = - Ground signal
- **N.C.** = - Not Connected

**N.B.**
Connection between RKD LT and Futaba M404SD01BA (40x4 characters) must be performed using the proper adaptor, made by grifo®, called IAF 404.
The connector for LCD graphic displays, called J2, is a 2.54 mm pitch, 20 pins, male, vertical, low profile connector. It allows a quick connection to the display by using a common 20 pins flat cable, if present. Here follows the pin out.

Please remark that all the supply voltages for the display are given directly by the RKD LT board.

**FIGURE 15: J2 - CONNECTOR FOR GRAPHIC LCD DISPLAY**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>D0 - D7</td>
<td>I/O - D0+D7 data signals</td>
</tr>
<tr>
<td>/WR</td>
<td>O - Write signal</td>
</tr>
<tr>
<td>/RD</td>
<td>O - Read signal</td>
</tr>
<tr>
<td>/CE</td>
<td>O - Device selection signal</td>
</tr>
<tr>
<td>/RES</td>
<td>O - Reset signal</td>
</tr>
<tr>
<td>C/D</td>
<td>O - Command/Data selection signal</td>
</tr>
<tr>
<td>E</td>
<td>O - Enable signal (only for displays mounting an Hitachi controller)</td>
</tr>
<tr>
<td>FS</td>
<td>O - Font selection in 6x8 or 8x8 alphanumeric mode (only for some displays)</td>
</tr>
<tr>
<td>VEE</td>
<td>O - Negative voltage for the display contrast</td>
</tr>
<tr>
<td>+5 Vdc</td>
<td>O - Power supply for the display</td>
</tr>
<tr>
<td>GND</td>
<td>O - Ground signal</td>
</tr>
<tr>
<td>N.C.</td>
<td>Not Connected</td>
</tr>
</tbody>
</table>

**N.B.**

RKD LT board must be configured according to the display controller used (Toshiba or Hitachi). This operation can be performed only by grifo®, so the controller type must be indicated in the order.
The connector for **BUS ABACO®**, called J3, is a DIN 41612 64 pins M 90 degrees A+C type C connector. Here follows the pin out for the one installed on **RKD LT**, and the standard pin out of 8 and 16 bits **BUS ABACO®**.

![Figure 16: J3 - Connector for ABACO® BUS](image)

**Figure 16: J3 - Connector for ABACO® BUS**
Symbols description:

### 8 bit CPU

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A0-A15</td>
<td>O - Address BUS</td>
</tr>
<tr>
<td>D0-D7</td>
<td>I/O - Data BUS</td>
</tr>
<tr>
<td>INT</td>
<td>I - Interrupt request</td>
</tr>
<tr>
<td>NMI</td>
<td>I - Non Mascherable Interrupt</td>
</tr>
<tr>
<td>HALT</td>
<td>O - Halt state</td>
</tr>
<tr>
<td>MREQ</td>
<td>O - Memory Request</td>
</tr>
<tr>
<td>IORQ</td>
<td>O - Input Output Request</td>
</tr>
<tr>
<td>RD</td>
<td>O - Read cycle status</td>
</tr>
<tr>
<td>WR</td>
<td>O - Write cycle status</td>
</tr>
<tr>
<td>BUSAK</td>
<td>O - BUS Acknowledge</td>
</tr>
<tr>
<td>WAIT</td>
<td>I - Wait</td>
</tr>
<tr>
<td>BUSRQ</td>
<td>I - BUS Request</td>
</tr>
<tr>
<td>RESET</td>
<td>O - Reset</td>
</tr>
<tr>
<td>M1</td>
<td>O - Machine cycle one</td>
</tr>
<tr>
<td>RFSH</td>
<td>O - Refresh (for dynamic memories)</td>
</tr>
<tr>
<td>MEMDIS</td>
<td>I - Memory Display (signal output by a memory mapped peripheral device)</td>
</tr>
<tr>
<td>VDUSEL</td>
<td>I - VDU Selection (abilitation to be memory mapped for the peripheral device)</td>
</tr>
<tr>
<td>IEI</td>
<td>I - Interrupt Enable Input</td>
</tr>
<tr>
<td>CLK</td>
<td>O - Clock:</td>
</tr>
<tr>
<td>R.T.</td>
<td>I - Reset</td>
</tr>
<tr>
<td>+5 Vdc</td>
<td>I - +5 Vdc power supply</td>
</tr>
<tr>
<td>+12 Vdc</td>
<td>I - +12 Vdc power supply</td>
</tr>
<tr>
<td>-12 Vdc</td>
<td>I - -12 Vdc power supply</td>
</tr>
<tr>
<td>GND</td>
<td>Ground for all the signals of the BUS</td>
</tr>
</tbody>
</table>

### 16 bit CPU

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A0-A22</td>
<td>O - Address BUS</td>
</tr>
<tr>
<td>D0-D15</td>
<td>I/O - Data BUS</td>
</tr>
<tr>
<td>RD UDS</td>
<td>O - Read Upper Data Strobe</td>
</tr>
<tr>
<td>WR UDS</td>
<td>O - Write Upper Data Strobe</td>
</tr>
<tr>
<td>IACK</td>
<td>O - Interrupt Acknowledge</td>
</tr>
<tr>
<td>RD LDS</td>
<td>O - Read Lower Data Strobe</td>
</tr>
<tr>
<td>WR LDS</td>
<td>O - Write Lower Data Strobe</td>
</tr>
</tbody>
</table>

N.B.
The above mentioned indications of directionality are relative to a GPC® command board. These have been kept untouched to avoid ambiguities or misinterpretation in case of multi-boards systems.
JUMPERS

On RKD LT there are 11 jumpers for card configuration. Connecting these jumpers, the User can define for example the peripheral devices functionality, the serial communication interface and so on. To easily locate the jumpers please refer to figure 19. Here below is the jumpers list, location and function:

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>PIN N.</th>
<th>USE</th>
</tr>
</thead>
<tbody>
<tr>
<td>JP1</td>
<td>3</td>
<td>RESERVED</td>
</tr>
<tr>
<td>JP2</td>
<td>5</td>
<td>Enables the circuitry for RS 422 serial communication</td>
</tr>
<tr>
<td>JP3</td>
<td>3</td>
<td>In conjunction with JP4, JP5 e JP10, selects the type of communication for the two serial lines</td>
</tr>
<tr>
<td>JP4</td>
<td>3</td>
<td>In conjunction with JP3, JP5 e JP10, selects the type of communication for the two serial lines</td>
</tr>
<tr>
<td>JP5</td>
<td>3</td>
<td>In conjunction with JP3, JP4 e JP10, selects the type of communication for the two serial lines</td>
</tr>
<tr>
<td>JP6</td>
<td>3</td>
<td>Enables the circuitry for RS 422 serial communication</td>
</tr>
<tr>
<td>JP7</td>
<td>2</td>
<td>It connects the termination resistor to the RS 422 reception line</td>
</tr>
<tr>
<td>JP8</td>
<td>2</td>
<td>It connects the BUS ABACO® /M1 signal</td>
</tr>
<tr>
<td>JP10</td>
<td>3</td>
<td>In conjunction with JP3, JP4 e JP5, selects the type of communication for the two serial lines</td>
</tr>
<tr>
<td>JP11</td>
<td>2</td>
<td>It selects the voltage range for the LCD contrast</td>
</tr>
<tr>
<td>S3.1</td>
<td>2</td>
<td>It connects the BUS ABACO® /RESET signal</td>
</tr>
</tbody>
</table>

* The "*" denotes the default connection, or on the other hand the connection set up at the end of testing phase, that is the configuration the User receives.

5 PINS JUMPERS

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>CONNECTION</th>
<th>USE</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>JP2</td>
<td>position 2-3 not connected</td>
<td>Enables the RS 422 communication circuitry</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Disables the RS 422 communication circuitry</td>
<td></td>
</tr>
</tbody>
</table>

* The "*" denotes the default connection, or on the other hand the connection set up at the end of testing phase, that is the configuration the User receives.
Figure 19: Jumpers Location
### Figure 20: 3 Pins Jumper Table

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>CONNECTION</th>
<th>USE</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>JP3</td>
<td>position 1-2</td>
<td>In conjunction with JP4, JP5 and JP10, it configures the main serial line (A) for RS 422 or Current Loop and the auxiliary serial line (B) for RS 232</td>
<td></td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>In conjunction with JP4, JP5 and JP10, it configures the main serial line (A) for RS 232 and the auxiliary serial line (B) for RS 422 or Current Loop</td>
<td></td>
</tr>
<tr>
<td></td>
<td>not connected</td>
<td>In conjunction with JP4, JP5 and JP10, it configures both serial lines for RS 232</td>
<td></td>
</tr>
<tr>
<td>JP4</td>
<td>position 1-2</td>
<td>In conjunction with JP3, JP5 and JP10, it configures the main serial line (A) for RS 422 or Current Loop and the auxiliary serial line (B) for RS 232</td>
<td></td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>In conjunction with JP3, JP5 and JP10, it configures the main serial line (A) for RS 232 and the auxiliary serial line (B) for RS 422 or Current Loop</td>
<td></td>
</tr>
<tr>
<td></td>
<td>not connected</td>
<td>In conjunction with JP3, JP5 and JP10, it configures both serial lines for RS 232</td>
<td></td>
</tr>
<tr>
<td>JP5</td>
<td>position 1-2</td>
<td>In conjunction with JP3, JP4 and JP10, it configures the main serial line (A) for RS 422 or Current Loop and the auxiliary serial line (B) for RS 232</td>
<td></td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>In conjunction with JP3, JP4 and JP10, it configures the main serial line (A) for RS 232 and the auxiliary serial line (B) for RS 422 or Current Loop</td>
<td></td>
</tr>
<tr>
<td></td>
<td>not connected</td>
<td>In conjunction with JP3, JP4 and JP10, it configures both serial lines for RS 232</td>
<td></td>
</tr>
<tr>
<td>JP6</td>
<td>position 2-3</td>
<td>Enables the RS 422 communication circuitry</td>
<td></td>
</tr>
<tr>
<td></td>
<td>not connected</td>
<td>Disables the RS 422 communication circuitry</td>
<td></td>
</tr>
<tr>
<td>JP10</td>
<td>position 1-2</td>
<td>In conjunction with JP3, JP4 and JP5, it configures the main serial line (A) for RS 422 or Current Loop and the auxiliary serial line (B) for RS 232</td>
<td></td>
</tr>
<tr>
<td></td>
<td>position 2-3</td>
<td>In conjunction with JP3, JP4 and JP5, it configures the main serial line (A) for RS 232 and the auxiliary serial line (B) for RS 422 or Current Loop</td>
<td></td>
</tr>
<tr>
<td></td>
<td>not connected</td>
<td>In conjunction with JP3, JP4 and JP5, it configures both serial lines for RS 232</td>
<td></td>
</tr>
</tbody>
</table>

**N.B.**

For a correct use of the serial communication section, jumpers JP3, JP4, JP5 and JP10 must be always connected in the same positions; other combinations are meaningless.
2 PINS JUMPERS

<table>
<thead>
<tr>
<th>JUMPER</th>
<th>CONNECTION</th>
<th>USE</th>
<th>DEF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>JP7</td>
<td>not connected</td>
<td>It doesn't connect the termination resistor to the RS 422 communication circuitry</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connesso</td>
<td>It connects the termination resistor to the RS 422 communication circuitry</td>
<td></td>
</tr>
<tr>
<td>JP8</td>
<td>not connected</td>
<td>Parallel interface doesn't manage the BUS ABACO® /M1 signal</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connesso</td>
<td>Parallel interface manages the BUS ABACO® /M1 signal</td>
<td></td>
</tr>
<tr>
<td>JP11</td>
<td>not connected</td>
<td>It selects the -7.5 ÷ -15 Vdc voltage range for the LCD contrast regulation</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>connesso</td>
<td>It selects the -6 ÷ -12 Vdc voltage range for the LCD contrast regulation</td>
<td></td>
</tr>
<tr>
<td>S3.1</td>
<td>OFF</td>
<td>It doesn't connect the BUS ABACO® /RESET signal to its proper on board circuitry</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>ON</td>
<td>It connects the BUS ABACO® /RESET signal to its proper on board circuitry</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 21: 2 pins jumpers table**

**GRAPHIC LCD CONTRAST REGULATION**

**RKD LT** mounts a circuitry capable to give out the negative voltage needed to regulate the contrast of the LCD graphic displays.

According to the needs of the display being used, it is possible to select, by acting on JP11 (please see previous tables), a -7.5 ÷ -15 Vdc or a -6 ÷ -12 Vdc voltage range.

The regulation of this voltage inside its range and so the regulation of the contrast intensity is made by acting on a trimmer called **R27**.

To easily locate this component please refer to figure 8.

Optimal contrast regulation betters the display visibility also when environmental light is variable.
SERIAL COMMUNICATION

The communication lines of RKD LT board can be buffered as RS 232, RS 422 or Current Loop. The selection of which communication mode to employ is made by acting on proper jumpers, whose location and connection has been shown and explained in the previous paragraphs. Here follows the description of the possible configurations; please remark that jumpers not mentioned have no influence for the communication whatever their connection is.

- **MAIN SERIAL LINE (A) AS RS 422, AUXILIARY SERIAL LINE (B) AS RS 232**
  MAX 202 serial driver must be installed on U15, driver SN 75176 must be installed on U17 and U18 while no driver must be installed on U17A and U18A. No jumper must be connected.

![](image1)

**Figure 22: Main serial line as RS 422, auxiliary serial line RS 232**

- **MAIN SERIAL LINE (A) AS CURRENT LOOP, AUXILIARY SERIAL LINE (B) AS RS 232**
  MAX202 serial driver must be installed on U15, driver HCPL4200 must be installed on U17A, driver HCPL4100 must be installed on U18A while no driver must be installed on U16, U17 and U18. Jumpers JP3, JP4, JP5 and JP10 must be connected in position 1-2, while the remaining jumpers must be disconnected.

![](image2)

**Figure 23: Main serial line as Current Loop, auxiliary serial line RS 232**
- MAIN SERIAL LINE (A) AS RS 232, AUXILIARY SERIAL LINE (B) AS RS 422
MAX 202 serial driver must be installed on U16, driver SN 75176 must be installed on U17 and U18 while no driver must be installed on U17A and U18A.

**Figure 24: Main serial line as RS 232, auxiliary serial line RS 232**

- MAIN SERIAL LINE (A) AS RS 232, AUXILIARY SERIAL LINE (B) AS CURRENT LOOP
MAX202 serial driver must be installed on U16, driver HCPL4200 must be installed on U17A, driver HCPL4100 must be installed on U18A while no driver must be installed on U16, U17 and U18. Jumpers JP3, JP4, JP5 and JP10 must be connected in position 2-3, while the remaining jumpers must be disconnected.

**Figure 25: Main serial line as RS 232, auxiliary serial line Current Loop**

- MAIN SERIAL LINE (A) AS RS 232, AUXILIARY SERIAL LINE (B) AS RS 422
MAX 202 serial driver must be installed on U15 and U16, while no driver must be installed on U17, U18, U17A and U18A. No jumper must be connected.

**Figure 26: Main serial line as RS 232, auxiliary serial line RS 232**
PARALLEL INTERFACE DESCRIPTION

INTRODUCTION

This chapter contains an exhaustive hardware and software informations about how to use the parallel interface installed on RKD LT board. Part of these informations consist of the occupation of ABACO® BUS I/O mapping by the RKD LT board and the communication modalities to the master control unit. Of course, if the RKD LT board is configured to communicate by the serial line, the informations contained in this chapter are needless.

BOARD MAPPING

The RKD LT board takes only two consecutive bytes in the I/O addressing space, these bytes can be allocated from a different base address according to how the board is mapped. This feature allows to install more than one RKD LT board across the same BUS ABACO®, or to install the board across a bus where other boards are present, obtaining a structure easy to expand without any modification to the software already written.

The BUS interface circuitry is installed on the board itself, to select the base address it uses an 8 pins Dip Switch, called S3, where the User can directly set the base address. Here follows the correspondance between the Dip Switch disposition and the selected address space management mode:

- **S3.1** -> See paragraph “JUMPER”
- **S3.2** -> A1 address signal
- **S3.3** -> A2 address signal
- **S3.4** -> A3 address signal
- **S3.5** -> A4 address signal
- **S3.6** -> A5 address signal
- **S3.7** -> A6 address signal
- **S3.8** -> A7 address signal

The Dip Switch works in complemented logic, this means that an ON position generates a logic zero, while on OFF position generates a logic one.

Also jumper JP8, described in the previous chapter, influences the addressing logic and must be set according to the type of control board (GPC® serie) used. In detail, if the control board is provided with /M1 signal on the BUS ABACO® connector then JP8 must be connected and viceversa.

Please be careful not to install more than one board inside the same addressing space, considering also the amount of bytes occupied by every board in the addressing space. In case this condition should not be respected a BUS conflict happens, compromising the regular work of the whole system.
Here follows an example.
If the User wants to map the RKD LT board on address 04AH in a BUS where also a control board provided with /M1 signal works, the configuration must be:

- JP8 -> Connected
- S3.2 -> OFF
- S3.3 -> ON
- S3.4 -> OFF
- S3.5 -> ON
- S3.6 -> ON
- S3.7 -> OFF
- S3.8 -> ON

To locate on the board the devices here mentioned, please refer to figure 8 and 19, printed in the previous pages.

**PARALLEL COMMUNICATION REGISTERS**

The parallel communication registers of RKD LT board are mapped at the below mentioned addresses, having indicated with <base add> the base address for the board mapping, that is the address set by S3 as explained in the previous paragraph.

<table>
<thead>
<tr>
<th>REGISTER</th>
<th>ADDRESS</th>
<th>R/W</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATUS</td>
<td>&lt;base add&gt;+00</td>
<td>R</td>
<td>Status register of the parallel interface</td>
</tr>
<tr>
<td>DATA</td>
<td>&lt;base add&gt;+01</td>
<td>R/W</td>
<td>Data register of the parallel interface</td>
</tr>
</tbody>
</table>

*Figure 27: Parallel communication addresses table*
PARALLEL COMMUNICATION MANAGEMENT

This paragraph explains the software management modalities of the parallel communication between RKD LT board and a master control unit. The read/write DATA register, described in the previous table, allows to receive from and to transmit informations to the board, while the read only STATUS registers must be interpreted as follows:

\[
\begin{array}{cccccccc}
\text{bit7} & \text{bit6} & \text{bit5} & \text{bit4} & \text{bit3} & \text{bit2} & \text{bit1} & \text{bit0} \\
\text{STATUS} &=& \text{OBF} & \text{IBF} & \text{NU} & \text{NU} & \text{NU} & \text{NU} & \text{NU} \\
\end{array}
\]

where:

- **NU** = Not Used
- **IBF** = If active (1) indicates that the RKD LT board is ready to receive a new data; the master control unit can write it to the DATA register.
- **OBF** = If active (1) indicates that the RKD LT board has written a data into its transmission buffer; the master control unit can read the data performing a read operation of the DATA register.

After a Reset or a Power-On the master control unit must verify that the RKD LT board has completed its initialization phase and is ready to receive data. This condition is indicated by the logic state of the flags IBF and OBF, which must show the values: IBF=1 and OBF=0.

The program running on the control board must start by performing a cycle (eventually timed out for more safety) to test the status of these signals and wait for the RKD LT board to be ready for the communication.

To show the above described modalities, here follow two examples subroutines written in CBZ 80:

"sendtorkd"
REM Transmits to RKD LT the value contained in the variable var%
REM Begin
DO : REM Waiting for bit IBF to activate
ibf%=INP(status%)
UNTIL ((ibf% AND &040)=&040)
OUT data%, var%
REM End
RETURN

"recfromrkd"
REM Tests whether RKD LT has sent a char; in such case returns it in the variable var%,
REM Otherwise returns -1
REM Begin
obf%=INP(status%) : REM Tests bit OBF
IF ((obf% AND &080)=&080) THEN var%=INP(data%) ELSE var%=-1
REM End
RETURN
SOFTWARE DESCRIPTION

The RKD LT board is a full functional video terminal; so whatever is received, except the command sequences, is displayed and the codes of the keys pressed on the BG Keyboard are sent to the master control unit.

To properly use the board, the User must first configure it by selecting: the kind of display used, the communication protocol, etc. This operation can be performed simply by acting on the configuration Dip Switches, as will be explained in the following pages.

In addition to the description of the many commands recognized by the board, there is a complete description of the command sequences accepted by the on board firmware.

Every code or code sequence is reported with a double description: the mnemonic one, employing ASCII characters, and the numeric one, expressed both in decimal and hexadecimal format.

BOARD CONFIGURATION

To configure the RKD LT board the User must act on the S1 and S2 Dip Switches, to easily locate these components on the board please refer to figure 8.

Here follow the configuration modalities, accepted by the TRKDGL and TRKDAF firmware. Please remark that the below described settings must be performed when the board is not supplied, because the acquisition of the Dip Switch configuration is made only after a Reset or a Power-On.

CONFIGURATION USING TRKDGL FIRMWARE

The configuration using the LCD graphic display manager TRKDGL firmware must be performed as described in the following table.

<table>
<thead>
<tr>
<th>N. DIP</th>
<th>FUNCTION</th>
<th>ON</th>
<th>OFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2.1</td>
<td>NOT USED</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>S2.2</td>
<td>Select the BAUD RATE of the main serial line</td>
<td>See next table</td>
<td></td>
</tr>
<tr>
<td>S2.3</td>
<td>Select the communication interface</td>
<td>ABACO® BUS</td>
<td>Main serial line</td>
</tr>
<tr>
<td>S1.1</td>
<td>Select the display model</td>
<td>See next table</td>
<td></td>
</tr>
<tr>
<td>S1.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1.4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1.5</td>
<td>Selects the LCD display working mode</td>
<td>Graphic</td>
<td>Alphanumeric</td>
</tr>
<tr>
<td>S1.7</td>
<td>Select the BAUD RATE of the auxiliary serial line</td>
<td>See next table</td>
<td></td>
</tr>
<tr>
<td>S1.8</td>
<td>Selects the standard of the commands</td>
<td>ADDS VIEWPOINT</td>
<td>TVI950</td>
</tr>
</tbody>
</table>

**FIGURE 28: TRKDGL DIP SWITCH CONFIGURATION TABLE**
Display model selection:

The selection of the LCD graphic display is performed by the first 4 dips of S1, as described in the following table.

<table>
<thead>
<tr>
<th>S1.1</th>
<th>S1.2</th>
<th>S1.3</th>
<th>S1.4</th>
<th>DISPLAY (format - controller)</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
<td>TLX-711A and compatibles with char matrix <strong>6x8</strong> (240 x 64 pixels - TOSHIBA T6963C)</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
<td>TLX-711A and compatibles with char matrix <strong>8x8</strong> (240 x 64 pixels - TOSHIBA T6963C)</td>
</tr>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>ON</td>
<td>OFF</td>
<td>TLX-1391 and compatibles with char matrix <strong>6x8</strong> (128 x 128 pixels - TOSHIBA T6963C)</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>ON</td>
<td>OFF</td>
<td>TLX-1391 and compatibles with char matrix <strong>8x8</strong> (128 x 128 pixels - TOSHIBA T6963C)</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>OFF</td>
<td>OFF</td>
<td>TLX-1021 and compatibles (120 x 64 pixels - TOSHIBA T6963C)</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>OFF</td>
<td>OFF</td>
<td>TLX-1013 or compatibles (160 x 128 pixels - TOSHIBA T6963C)</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
<td>OFF</td>
<td>TLX-1301 and compatibles with char matrix <strong>6x8</strong> (240 x 128 pixels - TOSHIBA T6963C)</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>ON</td>
<td>OFF</td>
<td>TLX-1301 and compatibles with char matrix <strong>8x8</strong> (240 x 128 pixels - TOSHIBA T6963C)</td>
</tr>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
<td>ON</td>
<td>TLC-1091 and compatibles (240 x 128 pixels - TOSHIBA T6963C)</td>
</tr>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
<td>MGLS240128V2 and compatibles (240 x 128 pixels - HITACHI HD61830B)</td>
</tr>
</tbody>
</table>

**Figure 29: Graphic LCD model selection table**

**N. B.:**
The character matrix format is managed only in alphanumeric mode, when graphic mode is selected (dip S1.5 ON) the minimum character size is 8x8 pixels and the S1 dips must be configured opportune.
The selection of **6x8 character format in graphic mode is not acceptable** and causes visualization problems.
Main serial line Baud Rate selection:

If the parallel communication mode is selected (dip S2.4 ON) the configuration of dips S1.2 and S1.3 has no influence to the Baud Rate selection.
Otherwise, to set this parameter the User must act on S2 as follows.

<table>
<thead>
<tr>
<th>S2.2</th>
<th>S2.3</th>
<th>BAUD RATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>19200 Baud</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>9600 Baud</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>4800 Baud</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>2400 Baud</td>
</tr>
</tbody>
</table>

**Figure 30: TRKDGL Main serial line Baud Rate selection table**

Auxiliary serial line Baud Rate selection:

The auxiliary serial line Baud Rate selection must be performed by acting on dips S1.6 and S1.7, according to the following table.

<table>
<thead>
<tr>
<th>S1.6</th>
<th>S1.7</th>
<th>BAUD RATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>19200 Baud</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>9600 Baud</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>4800 Baud</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>2400 Baud</td>
</tr>
</tbody>
</table>

**Figure 31: TRKDGL Auxiliary serial line Baud Rate selection table**
CONFIGURATION USING TRKDAF FIRMWARE

The TRKDAF firmware is used to manage the Futaba fluorescent alphanumeric displays. To perform the configuration of RKD LT board using this firmware please see the following table.

<table>
<thead>
<tr>
<th>N. DIP</th>
<th>FUNCTION</th>
<th>ON</th>
<th>OFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2.1</td>
<td>NOT USED</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>S2.2</td>
<td>Select the BAUD RATE of the main serial line</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S2.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S2.4</td>
<td>Select the communication interface</td>
<td>ABACO® BUS</td>
<td>Main serial line</td>
</tr>
<tr>
<td>S1.1</td>
<td>Select the display model</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1.3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1.4</td>
<td>Selects the standard of the commands</td>
<td>ADDS VIEWPOINT</td>
<td>TVI950</td>
</tr>
<tr>
<td>S1.5</td>
<td>NOT USED</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>S1.6</td>
<td>Select the BAUD RATE of the auxiliary serial line</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1.7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S1.8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 32: TRKDAF Dip Switch configuration table

Display model selection:

The selection of the FUTABA fluorescent display is performed by the first 3 dips of S1, as described in the following table.

<table>
<thead>
<tr>
<th>S1.1</th>
<th>S1.2</th>
<th>S1.3</th>
<th>DISPLAY (format)</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
<td>Futaba M202SD01BA and compatibles (20x2 characters “small”)</td>
</tr>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>ON</td>
<td>Futaba M202SD08GK and compatibles (20x2 characters “big”)</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>OFF</td>
<td>Futaba M204SD01AA and compatibles (20x4 characters)</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
<td>Futaba M40SD04GJ and compatibles (40x1 characters)</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>OFF</td>
<td>Futaba M402SD07GK and compatibles (40x2 characters)</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>ON</td>
<td>Futaba M404SD01BA and compatibles (40x4 characters)</td>
</tr>
</tbody>
</table>

Figure 33: FUTABA fluorescent display selection table
Main serial line Baud Rate selection:

If the parallel communication mode is selected (dip S2.4 ON) the configuration of dips S1.2 and S1.3 has no influence to the Baud Rate selection.
Otherwise, to set this parameter the User must act on S2 as follows.

<table>
<thead>
<tr>
<th>S2.2</th>
<th>S2.3</th>
<th>BAUD RATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>19200 Baud</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>9600 Baud</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>4800 Baud</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>2400 Baud</td>
</tr>
</tbody>
</table>

**Figure 34: TRKDAF Main Serial Line Baude Rate Selection Table**

Auxiliary serial line Baud Rate selection:

The auxiliary serial line Baud Rate selection must be performed by acting on dips S1.6, S1.7 and S1.8, according to the following table.

<table>
<thead>
<tr>
<th>S1.6</th>
<th>S1.7</th>
<th>S1.8</th>
<th>BAUD RATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>OFF</td>
<td>19200 Baud</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>OFF</td>
<td>9600 Baud</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>OFF</td>
<td>4800 Baud</td>
</tr>
<tr>
<td>ON</td>
<td>ON</td>
<td>OFF</td>
<td>2400 Baud</td>
</tr>
<tr>
<td>OFF</td>
<td>OFF</td>
<td>ON</td>
<td>1200 Baud</td>
</tr>
<tr>
<td>ON</td>
<td>OFF</td>
<td>ON</td>
<td>600 Baud</td>
</tr>
<tr>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
<td>300 Baud</td>
</tr>
</tbody>
</table>

**Figure 35: TRKDAF Auxiliary Serial Line Baude Rate Selection Table**
RECEPTION BUFFER

**RKD LT** board is provided with a reception buffer to make it faster during the communication to the master control unit and reduce its stand by times. Every byte received, both in parallel and in serial communication mode, is immediately saved into this buffer (sized **3200 bytes**), then processed at the end of the operation in progress.

Of course, in the case of continuous transmission of command sequences that require a long execution time (such graphic mode commands, EEPROM management commands, etc.), the buffer is going to overflow. This is why the **RKD LT** board advises the other boards of the imminence of this event as hereunder described:

- **RS 232 serial communication**
  - \(\rightarrow\) /RTS deactivated (-12 Vdc)

- **Parallel communication**
  - \(\rightarrow\) IBF deactivated (logic level 0)

So the master control unit will have to manage opportunely these signals, suspending the transmission up to when the buffer of the **RKD LT** board will be empty again, and ready to receive new data. Please remark that, in case of **RS 422** or **Current Loop** serial communication, this feature is not available, because the /RTS signal is not present. The User will have to insert delays in the communication, long enough to allow the **RKD LT** board to perform the operations requested without filling completely the reception buffer.

**KEYBOARD ACQUISITION**

**RKD LT** board manages directly the 56 keys BG keyboard, or a generic 7x8 matrix keyboard. When the board detects a key hit transmits its code to the master control unit, according the figures reported in the next table.

There is also an AutoRepeat feature, in fact when the **RKD LT** board detects that the pressure of a key lasts longer than **0.8 seconds**, it will begin to send repeatedly its code every about **0.2 seconds**, up to when that key will be released.

In the following table the codes of the “special” keys (ENTER, arrows, etc.) are reported; it is remarkable that some of these codes depend on how the **RKD LT** board has been configured, that is to work under TVI 950 or ADDS Viewpoint terminal emulation.

<table>
<thead>
<tr>
<th>KEY</th>
<th>ADDS VIEWPOINT</th>
<th>TVI 950</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Dec.</td>
<td>Hex</td>
</tr>
<tr>
<td>UP Arrow “red”</td>
<td>26</td>
<td>1A</td>
</tr>
<tr>
<td>DOWN Arrow “red”</td>
<td>10</td>
<td>0A</td>
</tr>
<tr>
<td>LEFT Arrow “red”</td>
<td>21</td>
<td>15</td>
</tr>
<tr>
<td>RIGHT Arrow “red”</td>
<td>06</td>
<td>06</td>
</tr>
<tr>
<td>UP Arrow “cyan”</td>
<td>133</td>
<td>85</td>
</tr>
<tr>
<td>DOWN Arrow “cyan”</td>
<td>134</td>
<td>86</td>
</tr>
<tr>
<td>STOP / RUN</td>
<td>130</td>
<td>82</td>
</tr>
<tr>
<td>DEL</td>
<td>127</td>
<td>7F</td>
</tr>
<tr>
<td>ERASE</td>
<td>132</td>
<td>84</td>
</tr>
<tr>
<td>ENTER</td>
<td>13</td>
<td>0D</td>
</tr>
<tr>
<td>Spacebar</td>
<td>32</td>
<td>20</td>
</tr>
</tbody>
</table>

**FIGURE 36: SPECIAL KEYS CODE TABLE**
All the remaining keys are managed by the Shift, Caps Lock and Control functions, that can be obtained holding respectively the **SHIFT**, **DRAW** and **CTRL** keys.

In the following table the decimal, hexadecimal and ASCII codes of these keys in the many possible combinations, are reported.

<table>
<thead>
<tr>
<th>KEY</th>
<th>NORMAL</th>
<th>+ SHIFT</th>
<th>+ DRAW (Caps lock)</th>
<th>+ DRAW + SHIFT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ?</td>
<td>49</td>
<td>31</td>
<td>1</td>
<td>!</td>
</tr>
<tr>
<td>2 &quot;</td>
<td>50</td>
<td>32</td>
<td>2</td>
<td>&quot;</td>
</tr>
<tr>
<td>3 #</td>
<td>51</td>
<td>33</td>
<td>3</td>
<td>#</td>
</tr>
<tr>
<td>4 $</td>
<td>52</td>
<td>34</td>
<td>4</td>
<td>$</td>
</tr>
<tr>
<td>5 %</td>
<td>53</td>
<td>35</td>
<td>5</td>
<td>%</td>
</tr>
<tr>
<td>6 &amp;</td>
<td>54</td>
<td>36</td>
<td>6</td>
<td>&amp;</td>
</tr>
<tr>
<td>7 '</td>
<td>55</td>
<td>37</td>
<td>7</td>
<td>'</td>
</tr>
<tr>
<td>8 (</td>
<td>56</td>
<td>38</td>
<td>8</td>
<td>(</td>
</tr>
<tr>
<td>9 )</td>
<td>57</td>
<td>39</td>
<td>9</td>
<td>)</td>
</tr>
<tr>
<td>0 @</td>
<td>48</td>
<td>30</td>
<td>0</td>
<td>@</td>
</tr>
<tr>
<td>: *</td>
<td>58</td>
<td>3A</td>
<td>2A</td>
<td>*</td>
</tr>
<tr>
<td>- =</td>
<td>45</td>
<td>2D</td>
<td>3D</td>
<td>=</td>
</tr>
<tr>
<td>; +</td>
<td>59</td>
<td>3B</td>
<td>3B</td>
<td>+</td>
</tr>
<tr>
<td>, &lt;</td>
<td>44</td>
<td>2C</td>
<td>3C</td>
<td>&lt;</td>
</tr>
<tr>
<td>, &gt;</td>
<td>46</td>
<td>2E</td>
<td>3E</td>
<td>&gt;</td>
</tr>
<tr>
<td>A</td>
<td>97</td>
<td>61</td>
<td>a</td>
<td>A</td>
</tr>
<tr>
<td>B</td>
<td>98</td>
<td>62</td>
<td>b</td>
<td>B</td>
</tr>
<tr>
<td>C</td>
<td>99</td>
<td>63</td>
<td>c</td>
<td>C</td>
</tr>
<tr>
<td>D</td>
<td>100</td>
<td>64</td>
<td>d</td>
<td>D</td>
</tr>
<tr>
<td>E</td>
<td>101</td>
<td>65</td>
<td>e</td>
<td>E</td>
</tr>
<tr>
<td>F</td>
<td>102</td>
<td>66</td>
<td>f</td>
<td>F</td>
</tr>
<tr>
<td>G</td>
<td>103</td>
<td>67</td>
<td>g</td>
<td>G</td>
</tr>
<tr>
<td>H</td>
<td>104</td>
<td>68</td>
<td>h</td>
<td>H</td>
</tr>
<tr>
<td>I</td>
<td>105</td>
<td>69</td>
<td>i</td>
<td>I</td>
</tr>
<tr>
<td>J</td>
<td>106</td>
<td>6A</td>
<td>j</td>
<td>J</td>
</tr>
<tr>
<td>K</td>
<td>107</td>
<td>6B</td>
<td>k</td>
<td>K</td>
</tr>
<tr>
<td>L</td>
<td>108</td>
<td>6C</td>
<td>l</td>
<td>L</td>
</tr>
<tr>
<td>M</td>
<td>109</td>
<td>6D</td>
<td>m</td>
<td>M</td>
</tr>
<tr>
<td>N</td>
<td>110</td>
<td>6E</td>
<td>n</td>
<td>N</td>
</tr>
<tr>
<td>O</td>
<td>111</td>
<td>6F</td>
<td>o</td>
<td>O</td>
</tr>
<tr>
<td>P</td>
<td>112</td>
<td>70</td>
<td>p</td>
<td>P</td>
</tr>
<tr>
<td>Q</td>
<td>113</td>
<td>71</td>
<td>q</td>
<td>Q</td>
</tr>
<tr>
<td>R</td>
<td>114</td>
<td>72</td>
<td>r</td>
<td>R</td>
</tr>
<tr>
<td>S</td>
<td>115</td>
<td>73</td>
<td>s</td>
<td>S</td>
</tr>
<tr>
<td>T</td>
<td>116</td>
<td>74</td>
<td>t</td>
<td>T</td>
</tr>
<tr>
<td>U</td>
<td>117</td>
<td>75</td>
<td>u</td>
<td>U</td>
</tr>
<tr>
<td>V</td>
<td>118</td>
<td>76</td>
<td>v</td>
<td>V</td>
</tr>
<tr>
<td>W</td>
<td>119</td>
<td>77</td>
<td>w</td>
<td>W</td>
</tr>
<tr>
<td>X</td>
<td>120</td>
<td>78</td>
<td>x</td>
<td>X</td>
</tr>
<tr>
<td>Y</td>
<td>121</td>
<td>79</td>
<td>y</td>
<td>Y</td>
</tr>
<tr>
<td>Z</td>
<td>122</td>
<td>7A</td>
<td>z</td>
<td>Z</td>
</tr>
</tbody>
</table>

**FIGURE 37: ALPHANUMERIC KEYS CODE TABLE**
In this table the combination obtainable pressing the CTRL key are not reported because the codes are very easy to calculate by simply subtracting 64 (40 Hex).
For example the combination CTRL+SHIFT+A generates the code: 65-64=1.
Also, we would remark that the Control function is available only for keys, or keys combinations, whose code is equal to or greater than 64 (@, A, B, C, ...).

To locate the keys in the 7x8 matrix keyboard, please refer to the previously reported BG keyboard electric diagram. In that picture also a LED and abuzzer are shown. RKD LT board manages them using the following modalities:

<table>
<thead>
<tr>
<th>LED</th>
<th>Buzzer</th>
</tr>
</thead>
<tbody>
<tr>
<td>-&gt;</td>
<td>-&gt;</td>
</tr>
</tbody>
</table>

Indicates the Caps Lock mode status: ON = capitol letters
Used for the feedback when a key is hit (KeyClick), when this feature is enabled.

CHARACTERS REPRESENTATION

RKD LT board running TRKDAF or TRKDGL in alphanumeric mode visualizes directly on the connected display all the characters having a code included in the range 32÷255 (20÷FF Hex); while when running TRKDGL in graphic mode this range reduces to 32÷126 (20÷7E Hex). Characters having code out of these ranges, if not commands, are ignored.
The characters having code included in the range 32÷127 (20÷7F Hex) are ASCII standard, while the ones having code in the range 128÷255 (80÷FF Hex), have different effects according to the model of display used. For further informations please refer to the tables reported in the specific documentation.
The character will be displayed in the current cursor position, the cursor will advance to the next position, if it was in the last position (bottom right corner) it will be placed in the Home position (top left corner).

CURSOR POSITIONING COMMANDS

Here follows a list of the several cursor positioning commands available according to the installed firmware.

CURSOR LEFT

<table>
<thead>
<tr>
<th>TVI 950</th>
<th>Code: 08</th>
<th>Mnemonic: BS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDS Viewpoint</td>
<td>Code: 21</td>
<td>(15 Hex)</td>
</tr>
<tr>
<td></td>
<td>Mnemonic: NACK</td>
<td></td>
</tr>
</tbody>
</table>

The cursor is moved one position to the left without altering the content of the display.
If the cursor is in the Home position, it will be placed in the last position (bottom right) of the display.
CURSOR RIGHT

**TVI 950**
- Code: 09
- Mnemonic: HT

**ADDS Viewpoint**
- Code: 06
- Mnemonic: ACK

The cursor is moved one position to the right without altering the content of the display.
If the cursor is in the last position (bottom right) of the display, it will be placed in the Home position.

---

CURSOR DOWN

**TVI 950 and ADDS Viewpoint**
- Code: 10 (0A Hex)
- Mnemonic: LF

The cursor is moved to the row under the one where it finds, remaining in the same column.
If the cursor finds in the last row, it will be moved to the first row.

---

CURSOR UP

**TVI 950**
- Code: 11 (0B Hex)
- Mnemonic: VT

**ADDS Viewpoint**
- Code: 26 (1A Hex)
- Mnemonic: SUB

The cursor is moved to the row above the one where it finds, remaining in the same column.
If the cursor finds in the first row, it will be moved to the last row.

---

CARRIAGE RETURN

**TVI 950 and ADDS Viewpoint**
- Code: 13 (0D Hex)
- Mnemonic: CR

The cursor is moved to the beginning of the row where it finds.
CARRIAGE RETURN + LINE FEED

TVI 950 and ADDS Viewpoint

- **Code**: 29 (1D Hex)
- **Mnemonic**: GS

The cursor is moved to the beginning of the row above the one where it finds. If the cursor finds in the last row, it will be moved to the beginning of the first line, that is in the Home position.

HOME

TVI 950

- **Code**: 28 (1C Hex)
- **Mnemonic**: FS

ADDS Viewpoint

- **Code**: 01
- **Mnemonic**: SOH

The cursor is moved to the Home position, that is the first row of the first column of the display.

CURSOR ABSOLUT POSITIONING

TVI 950

- **Code**: 27 61 r c (1B 3D r c Hex)
- **Mnemonic**: ESC = ASCII(r) ASCII(c)

ADDS Viewpoint

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

The cursor is moved to the absolut position indicated by “r” and “c”.

These codes indicate the numbers of row and column of the display, plus an offset of **32 (20 Hex)**. If, for example, the User wants to place the cursor in the Home position (row 0, column 0) under the ADDS Viewpoint terminal emulation, he/she will have to send the sequence: **27 89 32 32**.

If the values of row and column are not compatible to the type of display installed, this command is ignored.
CHARACTERS DELETION COMMANDS

Here follows a list of the several characters deletion commands available according to the installed firmware.

CLEAR PAGE

TVI 950 ed ADDS Viewpoint

| Code: 12 (0C Hex) | Mnemonic: FF |

The whole display is cleared and the cursor is placed in the Home position.

CLEAR END OF LINE

TVI 950

| Code: 23 (17 Hex) | Mnemonic: ETB |

ADDs Viewpoint

| Code: 27 75 (1B 4B Hex) | Mnemonic: ESC K |

All characters between the cursor and the end of the line where it finds are deleted, included the one under the cursor.

The cursor remains in the same position.

If, for example, the cursor finds at the beginning of a line, this command will delete the whole line.
CURSOR ATTRIBUTES MANAGEMENT COMMANDS

Here follows a list of the several cursor attributes management commands available according to the installed firmware.

CURSOR DISABLED

TVI 950 e ADDS Viewpoint

Code: 27 80 (1B 50 Hex)

Mnemonic: ESC P

The cursor is disabled and no more visible.

CURSOR ENABLED AND FIXED

TVI 950 e ADDS Viewpoint

Code: 27 79 (1B 4F Hex)

Mnemonic: ESC O

The cursor is enabled and visible as a fixed line placed under the character.

NOTE

This command is not available when Futaba 20x4 and 40x4 fluorescent displays are connected.

CURSOR ENABLED AND FLASHING

TVI 950 e ADDS Viewpoint

Code: 27 77 (1B 4D Hex)

Mnemonic: ESC M

The cursor is enabled and visible as a flashing line placed under the character.

NOTE

This command is not available when Futaba 20x4 and 40x4 fluorescent displays are connected.

FLASHING BLOCK CURSOR

TVI 950 e ADDS Viewpoint

Code: 27 81 (1B 51 Hex)

Mnemonic: ESC Q

The cursor is enabled and visible as a flashing block.

NOTE

This command is not available for displays mounting an Hitachi HD61830B controller.
AUXILIARY SERIAL PORT MANAGEMENT COMMANDS

Here follows a list of the several auxiliary serial port management commands available according to the installed firmware.

ECHO ACTIVATION ON AUXILIARY SERIAL PORT

TVI 950
Code: 27 33 65 (1B 21 41 Hex)
Mnemonic: ESC ! A

ADDS Viewpoint
Code: 27 51 (1B 33 Hex)
Mnemonic: ESC 3

This command activates the echo (re-transmission) on auxiliary serial port of all the characters received from the master control unit.

ECHO DEACTIVATION ON AUXILIARY SERIAL PORT

TVI 950
Code: 27 33 65 (1B 21 41 Hex)
Mnemonic: ESC ! A

ADDS Viewpoint
Code: 27 52 (1B 34 Hex)
Mnemonic: ESC 4

The echo (re-transmission, see previous command) on the auxiliary serial port is disabled.

NOTE
In TVI 950 terminal emulation mode, the echo is managed by the same command. In that case the command has a bistable behaviour, in fact it activates and deactivates this feature according to its previous status (echo is disabled by default).
CHARACTERS ATTRIBUTES MANAGEMENT COMMANDS

Here follows a list of the several characters attributes management commands available only when running TRKDGL firmware.

REVERSE ATTRIBUTE ACTIVATION

<table>
<thead>
<tr>
<th>TVI 950</th>
<th>Code: 27 33 50  (1B \text{ 21  32} \text{ Hex})</th>
<th>Mnemonic: ESC ! 2</th>
</tr>
</thead>
</table>

**ADDS Viewpoint**

| Code: 27 48 80 14 \(1B \text{ 30  50  0E} \text{ Hex}\) | Mnemonic: ESC 0 P SO |

This command activates the reverse representation mode for alphanumeric characters. After this command, the representable characters will be visualized in reverse mode, up to when this attribute will be disabled.

**NOTE**

This command is not available for displays mounting an *Hitachi HD61830B* controller.

REVERSE ATTRIBUTE DEACTIVATION

<table>
<thead>
<tr>
<th>TVI 950</th>
<th>Code: 27 33 48  (1B \text{ 21  30} \text{ Hex})</th>
<th>Mnemonic: ESC ! 0</th>
</tr>
</thead>
</table>

**ADDS Viewpoint**

| Code: 27 48 80 15 \(1B \text{ 30  50  0F} \text{ Hex}\) | Mnemonic: ESC 0 P SI |

This command deactivates the reverse representation mode for alphanumeric characters. After this command, the representable characters will be visualized in normal mode, restoring the default visualization mode.

**NOTE**

This command is not available for displays mounting an *Hitachi HD61830B* controller.
KEYBOARD MANAGEMENT COMMANDS

Here follows a list of the several **BG keyboard** management commands available only when running **TRKDGL** firmware.

### KEYCLICK ON LOCAL BUZZER ACTIVATION

**TVI 950 e ADDS Viewpoint**

Code: 27 56 (1B 38 Hex)

Mnemonic: ESC 8

The **KeyClick** feature, that is a buzzer sound feedback when a key is pressed, is enabled. The sound will be produced by the buzzer installed on the **RKD LT** board.

### KEYCLICK ON LOCAL BUZZER DEACTIVATION

**TVI 950 e ADDS Viewpoint**

Code: 27 57 (1B 39 Hex)

Mnemonic: ESC 9

The **KeyClick** feature, that is a buzzer sound feedback when a key is pressed, is disabled. No sound will be produced by the buzzer installed on the **RKD LT** board on a key hit.

### KEYCLICK ON BG KEYBOARD ACTIVATION

**TVI 950 e ADDS Viewpoint**

Code: 27 58 (1B 3A Hex)

Mnemonic: ESC :

The **KeyClick** feature, that is a buzzer sound feedback when a key is pressed, is enabled. The sound will be produced by the buzzer installed on the **BG keyboard**.

### KEYCLICK ON BG KEYBOARD DEACTIVATION

**TVI 950 e ADDS Viewpoint**

Code: 27 59 (1B 3B Hex)

Mnemonic: ESC ;

The **KeyClick** feature, that is a buzzer sound feedback when a key is pressed, is disabled. No sound will be produced by the buzzer installed on the **BG keyboard** on a key hit.
EXTERNAL LEDS MANAGEMENT COMMANDS

Here follows a list of the several external LEDs management commands available according to the installed firmware.
Please remark that the LEDs work correctly only if they are connected in common cathod mode, as previously explained.

EXTERNAL LEDS ACTIVATION

**TVI 950 e ADDS Viewpoint**

*Code:* 27 50 mask (1B 32 mask Hex)
*Mnemonic:* ESC 2 ASCII(mask)

The 8 external LEDs, connected to CN1, are actived, according to the instructions given in mask. The bits of mask have the following meaning:

```
mask =   bit 7   bit 6   bit 5   bit 4   bit 3   bit 2   bit 1   bit 0
        LED7   LED6   LED5   LED4   LED3   LED2   LED1   LED0
```

If a certain bit is set to 0 the corresponding LED will be OFF, viceversa il wil be ON if the bit is set to 1.
Please refer to figures 3 and 4, reported in the previous pages, to match the LEDs numbers here used to the corresponding pins on CN1 connector of RKD LT board.
If, for example, the User wants to activate LED1 and LED4, he/she will have to send the sequence: 27 50 18.

EXTERNAL LEDS REVERSED ACTIVATION

**TVI 950 e ADDS Viewpoint**

*Code:* 27 53 mask (1B 35 mask Hex)
*Mnemonic:* ESC 5 ASCII(mask)

The 8 external LEDs, connected to CN1, are actived, according to the instructions given in mask but complemented in comparision with the previous command. The bits of mask have the following meaning:

```
mask =   bit 7   bit 6   bit 5   bit 4   bit 3   bit 2   bit 1   bit 0
        LED7   LED6   LED5   LED4   LED3   LED2   LED1   LED0
```

If a certain bit is set to 1 the corresponding LED will be OFF, viceversa il wil be ON if the bit is set to 0.
Please refer to figures 3 and 4, reported in the previous pages, to match the LEDs numbers here used to the corresponding pins on CN1 connector of RKD LT board.
If, for example, the User wants to activate LED1 and LED4, he/she will have to send the sequence: 27 53 237.
ALPHANUMERIC SCREENSHOTS MANAGEMENT COMMANDS

Here follows a list of the several alphanumeric screen management commands available according to the installed firmware.

SCREENSHOT VISUALIZATION

TVI 950 e ADDS Viewpoint

| Code: | 27 122 n.scr. H n.scr. L |
|       | (1B 7A n.scr. H n.scr. L Hex) |
| Mnemonic: | ESC z ASCII(n.scr. H) ASCII(n.scr. L) |

The alphanumeric screenshot whose number is n.scr. is displayed. n.scr. is a 16 bits parameter that must be sent as two bytes, first the high byte (bits 15÷8) then the low bytes (bits 7÷0).

An alphanumeric screenshot is a page of text that contains enough characters to cover exactly the screen of the display being used; these screenshots are generated using the RKD_EDIT.EXE program and are stored in the EPROM on the RKD LT, where also the firmware is stored. It is so possible to create the text masks to recall next in visualization; this allows to reduce the size of the User program and the amount of data to transmit. The RKD_EDIT.EXE program creates a binary file sized 64 KBytes; this file will be stored in the EPROM on the RKD LT, using an EPROM programmer, starting from the following addresses:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>EPROM for TRKDAF firmware</td>
<td>-&gt; 8000 Hex</td>
</tr>
<tr>
<td>EPROM for TRKDGL firmware</td>
<td>-&gt; 10000 Hex</td>
</tr>
</tbody>
</table>

Please remark that the EPROM involved is a 27C010 sized 128 KBytes whose content must not be modified.

For further informations about how to use the RKD_EDIT.EXE program, please refer to its own on line help.

The n.scr. parameter must be equal to or greater than 1 and equal to or lower than the number of the last screenshot available, indicated by the RKD_EDIT.EXE program during the creation of the alphanumeric screenshots.

If, for example, the User wants to display the screenshot number 258, he/she will have to send the sequence: 27 122 1 2.

NOTE
After the visualization of a screen shot, the cursor is placed in the Home position.
SCREENSHOT MEMORIZATION IN EEPROM

**TVI 950 e ADDS Viewpoint**

*Code:* 27 33 67 n.scr. car. 1 ... car. n  

(1B 21 43 n.scr. car. 1 ... car. n Hex)

*Mnemonic:* ESC ! C ASCII(n.scr.) ASCII(car. 1)…ASCII(car. n)

This command stores in the EEPROM on the RKD LT board the alphanumeric screenshot whose number is indicated by n.scr.

An alphanumeric screenshot is a page of text that contains enough characters to cover exactly the screen of the fluorescent display being used; so the number of characters that must be sent is always:

\[ n = \text{NUMBER OF ROWS} \times \text{NUMBER OF COLUMNS} \]

The characters in the screenshot must be visualizable, so their codes must be in the range 32÷255 (20÷FF Hex), while the n.scr. parameter must be equal to or greater than 1 and equal to or lower than the number of the last screenshot available, which depends on the type of display used and the size of the EEPROM installed, according to the following table:

**FIGURE 38: NUMBER OF SCREENSHOTS STORABLE IN EEPROM**

<table>
<thead>
<tr>
<th>DISPLAY</th>
<th>EEPROM 24c04 (512 bytes)</th>
<th>EEPROM 24c16 (2048 bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Futaba 20x2 characters</td>
<td>12</td>
<td>50</td>
</tr>
<tr>
<td>Futaba 20x4 characters</td>
<td>6</td>
<td>25</td>
</tr>
<tr>
<td>Futaba 40x1 characters</td>
<td>12</td>
<td>50</td>
</tr>
<tr>
<td>Futaba 40x2 characters</td>
<td>6</td>
<td>25</td>
</tr>
<tr>
<td>Futaba 40x4 characters</td>
<td>3</td>
<td>12</td>
</tr>
</tbody>
</table>

SCREENSHOT VISUALIZATION FROM EEPROM

**TVI 950 e ADDS Viewpoint**

*Code:* 27 33 68 n.scr.  

(1B 21 44 n.scr. Hex)

*Mnemonic:* ESC ! D ASCII(n.scr.)

The screenshot, corresponding to number n.scr., is recalled from the EEPROM and visualized on the display, starting from the first character.

The number of characters contained in the screenshot depends on the type of display used, as stated in the previous paragraph \( n = \text{Rows} \times \text{Columns} \), while the value of n.scr. must be included in the range 1÷n.max, where the value n.max can be obtained by the previous table.

**NOTE**

After the visualization of a screenshot, the cursor is placed in the Home position.
SCREENSHOT READ FROM EEPROM

TVI 950 e ADDS Viewpoint

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

The screenshot, corresponding to number n.scr., is recalled from the EEPROM and sent to the master control unit, starting from the first character.

The number of characters contained in the screenshot depends on the type of display used, as stated in the previously ($n = \text{Rows} \times \text{Columns}$), while the value of n.scr. must be included in the range $1 \div n_{\text{max}}$, where the value n.max can be obtained by the previous table.

GRAPHIC MODE COMMANDS

Here follows a list of the several graphic mode commands, available only under TRKDGL.

**REVERSE ATTRIBUTE SETTING**

TVI 950 e ADDS Viewpoint

- **Code:** 27 09 reverse (1B 09 reverse Hex)
- **Mnemonic:** ESC HT ASCII(reverse)

This command enables or disables the REVERSE attribute, according to the value of its parameter, in detail:

- reverse: 0 -> REVERSE attribute **disabled** (default)
- 1 -> REVERSE attribute **enabled**

This command is useful when the User wishes to delete a graphic primitive or parts of the display, in fact it is enough to activate the REVERSE attribute and redraw the picture to be deleted.

**CHARACTERS ZOOM SETTING**

TVI 950 e ADDS Viewpoint

- **Code:** 27 01 zoom (1B 01 zoom Hex)
- **Mnemonic:** ESC SOH ASCII(zoom)

This command sets the zoom level for the characters; in detail the new character size will be as reported in the following list:

- zoom: 0 -> 8x8 pixels characters (default)
- 1 -> 16x16 pixels characters
- 2 -> 24x24 pixels characters
- 3 -> 32x32 pixels characters
- 4 -> 40x40 pixels characters

**NOTE**

After the zoom level setting, the cursor is placed in the Home position.
CHARACTERS ELONGATION SETTING

**TVI 950 e ADDS Viewpoint**

**Codice:** 27 07 elong  
**Mnemonico:** ESC BEL ASCII(elong)

This command sets the elongation level for characters visualization; in detail the new character elongation level will be as reported in the following list:

- **elong:** 0 -> **Normal:** 8x8 pixels characters  
- **elong:** 1 -> **Vertical:** 8x16 pixels characters  
- **elong:** 2 -> **Orizzontal:** 16x8 pixels characters

CHARACTER CURSOR PLACEMENT

**TVI 950 e ADDS Viewpoint**

**Code:** 27 00 r c  
**Mnemonico:** ESC NUL ASCII(r) ASCII(c)

This command moves the cursor to the row and column indicated by r and c; this position is not absolute but depends on the values of zoom or elongation being used, so this position depends on the character size (for example, position 1,1 and zoom 0 correspond to the pixel whose coordinates are 8, 8 while the same position and zoom 1 correspond to the pixel having coordinates 16, 16). The values of r and c must be in the range 1÷maximum, where maximum depends on the type of graphic display used and the value of zoom currently set.

ABSOLUTE CURSOR PLACEMENT

**TVI 950 e ADDS Viewpoint**

**Codice:** 27 06 y x 00  
**Mnemonico:** ESC ACK ASCII(y) ASCII(x) NUL

This command moves the cursor to the row and column indicated by x and y; this position is absolute, so it does not depend on the value of the other parameters. The characters received after this command will be displayed from the indicated point, going to the right and to the bottom. The values of x and y must be in the range 1÷maximum, where maximum depends on the type of graphic display used.

**NOTE**
The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.
CHARACTERS WRITE DIRECTION

TVI 950 e ADDS Viewpoint

Codice: 27 10 dir (1B 0A dir Hex)
Mnemonic: ESC LF ASCII(dir)

This command sets the characters write direction; in detail according to the value of the parameter the direction will be:

- dir: 0 -> *Horizontal* characters writing (default)
- 1 -> *Vertical* characters writing

Vertical writing moves from top to bottom and writes the characters rotated of 90 degrees clockwise.

STRING VISUALIZATION

TVI 950 e ADDS Viewpoint

Code: 27 08 car. 1 … car. n 13 (1B 08 car. 1 … car. n 0D Hex)
Mnemonic: ESC BS ASCII(car. 1)…ASCII(car. n) CR

This command visualizes a string n characters long terminated by code 13 (CR), starting from the current cursor position and using the currently set attributes (zoom, elongation, reverse, etc.). The string is displayed only after the reception of the CR and must contain only characters in the range 32-126 (20-7E Hex).

NOTE
This command displays the received characters only after the completion of the transmission, differently from the direct visualization commands. This way it is possible to avoid partial visualizations due to communication delays (buffer full, slow transmission from the master control unit, etc.).

ONE PIXEL VISUALIZATION

TVI 950 e ADDS Viewpoint

Code: 27 29 y x 00 (1B 1D y x 00 Hex)
Mnemonic: ESC GS ASCII(y) ASCII(x) NUL

This command turns ON the pixel at the coordinates x and y.
The values of the x and y parameters must range from 0 to a maximum that depends on the type of display used.

NOTE
The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.
DRAWING A LINE

TVI 950 e ADDS Viewpoint

Code: 27 03 y1 x1 00 y2 x2 00
(1B 03 y1 x1 00 y2 x2 00 Hex)

Mnemonic: ESC ETX ASCII(y1) ASCII(x1) NUL ASCII(y2) ASCII(x2) NUL

This command draws a line starting from the point at coordinates x1, y1 and ending at the point at coordinates x2, y2. The values of these parameters must range from 0 to a maximum that depends on the type of display used.

NOTE: The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.

DRAWING A RECTANGLE

TVI 950 e ADDS Viewpoint

Code: 27 02 y1 x1 00 y2 x2 00
(1B 02 y1 x1 00 y2 x2 00 Hex)

Mnemonic: ESC STX ASCII(y1) ASCII(x1) NUL ASCII(y2) ASCII(x2) NUL

This command draws a rectangle whose top left corner is located at x1, y1 and whose bottom right corner is located at x2, y2. The values of these parameters must range from 0 to a maximum that depends on the type of display used.

NOTE: The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.

DRAWING A FILLED RECTANGLE

TVI 950 e ADDS Viewpoint

Code: 27 04 y1 x1 00 y2 x2 00
(1B 04 y1 x1 00 y2 x2 00 Hex)

Mnemonic: ESC EOT ASCII(y1) ASCII(x1) NUL ASCII(y2) ASCII(x2) NUL

This command draws a filled rectangle whose top left corner is located at x1, y1 and whose bottom right corner is located at x2, y2. The values of these parameters must range from 0 to a maximum that depends on the type of display used. Please remark that this command, opportunely matched with the reverse attribute, allows to delete rectangular parts of the display, leaving unchanged the rest.

NOTE: The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.
DRAWING A CIRCUMFERENCE

**TVI 950 e ADDS Viewpoint**

**Code**: 27 05 y x 00 r arc.in. n.arc.

**Mnemonic**: ESC ENQ ASCII(y) ASCII(x) NUL ASCII(r) ASCII(arc.in.) ASCII(n.arc.)

This command draws a circumference, or an arc of circumference, having ray equal to \( r \) and center at coordinates \( x \), \( y \). The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

The remaining codes in the sequence allow the user to draw half-circumferences or quarters of circumferences, they indicate the arc number from which to start drawing (arc.in.) and the number of arcs to draw (n.arc.); the values of these parameters must be in the range 1..4.

The following figure shows the correspondence between arcs and their numbers.

![Figure 39: Arcs numeration and disposition](image)

If, for example, the user wants to display a "C" shaped half-circumference having ray of 20 pixels and center located at (120, 60), he/she will have to send the sequence: 27 5 60 120 0 20 2 3.

**NOTE**: The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.

**DRAWING A CIRCUMFERENCE WHOSE RAY IS 3 PIXELS**

**TVI 950 e ADDS Viewpoint**

**Code**: 27 39 y x 00 (1B 27 y x 00 Hex)

**Mnemonic**: ESC ’ ASCII(y) ASCII(x) NUL

This command draws a circumference, having ray equal to 3 and center at coordinates \( x \), \( y \). The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

**NOTE**: The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.
DRA WING A FI LLED CI RUMFERENCE WHOSE RAY IS 3 P IXELS

TVI 950 e ADDS Viewpoint

Code: 27 38 y x 00 (1B 26 y x 00 Hex)
Mnemonic: ESC & ASCII(y) ASCII(x) NUL

This command draws a filled circumference, having ray equal to 3 and center at coordinates x, y. The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

Please remark that this command, opportunely matched with the previous one, allows to create easily on/off visual indications, useful, for example, in a synoptic display.

NOTE: The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.

DRA WING AN ARROW POINTING TO THE TOP

TVI 950 e ADDS Viewpoint

Code: 27 37 y x 00 (1B 25 y x 00 Hex)
Mnemonic: ESC % ASCII(y) ASCII(x) NUL

This command draws an arrow oriented to the top, as shown in the following figure:

![Figure 40: Top oriented arrow](image-url)

The parameters x and y correspond to the coordinates of the pixel indicated by the “X” in the above figure. The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

NOTE: The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.
DRAWING AN ARROW POINTING TO THE BOTTOM

TVI 950 e ADDS Viewpoint

Code: 27 33 y x 00 (1B 21 y x 00 Hex)

Mnemonic: ESC ' ASCII(y) ASCII(x) NUL

This command draws an arrow oriented to the bottom, as shown in the following figure:

![Figure 41: Bottom oriented arrow](image)

The parameters \( x \) and \( y \) correspond to the coordinates of the pixel indicated by the "X" in the above figure. The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

NOTE: The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.

DRAWING AN ARROW POINTING TO THE RIGHT

TVI 950 e ADDS Viewpoint

Code: 27 31 y x 00 (1B 1F y x 00 Hex)

Mnemonic: ESC US ASCII(y) ASCII(x) NUL

This command draws an arrow oriented to the right, as shown in the following figure:

![Figure 42: Right oriented arrow](image)

The parameters \( x \) and \( y \) correspond to the coordinates of the pixel indicated by the "X" in the above figure. The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

NOTE: The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.
DRAWWING AN ARROW POINTING TO THE LEFT

TVI 950 e ADDS Viewpoint

Code: 27 35 y x 00 (1B 23 y x 00 Hex)

Mnemonic: ESC # ASCII(y) ASCII(x) NUL

This command draws an arrow oriented to the left, as shown in the following figure:

![Left Oriented Arrow](image)

**FIGURE 43: LEFT ORIENTED ARROW**

The parameters x and y correspond to the coordinates of the pixel indicated by the "X" in the above figure.

The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

**NOTE:** The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.

DRAWWING A FILLED ARROW POINTING TO THE TOP

TVI 950 e ADDS Viewpoint

Codice: 27 36 y x 00 (1B 24 y x 00 Hex)

Mnemonico: ESC $ ASCII(y) ASCII(x) NUL

This command draws a filled arrow oriented to the top, as shown in the following figure:

![Top Oriented Filled Arrow](image)

**FIGURE 44: TOP ORIENTED FILLED ARROW**

The parameters x and y correspond to the coordinates of the pixel indicated by the "X" in the above figure.

The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

**NOTE:** The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.
**DRAWING A FILLED ARROW POINTING TO THE BOTTOM**

**TVI 950 e ADDS Viewpoint**

*Code:* 27 32 y x 00 (1B 20 y x 00 Hex)

*Mnemonic:* ESC SP ASCII(y) ASCII(x) NUL

This command draws a filled arrow oriented to the bottom, as shown in the following figure:

![Bottom Oriented Filled Arrow](image)

**Figure 45: Bottom Oriented Filled Arrow**

The parameters \( x \) and \( y \) correspond to the coordinates of the pixel indicated by the "X" in the above figure.

The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

**NOTE:** The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.

**DRAWING A FILLED ARROW POINTING TO THE RIGHT**

**TVI 950 e ADDS Viewpoint**

*Code:* 27 30 y x 00 (1B 1E y x 00 Hex)

*Mnemonic:* ESC RS ASCII(y) ASCII(x) NUL

This command draws a filled arrow oriented to the right, as shown in the following figure:

![Right Oriented Filled Arrow](image)

**Figure 46: Right Oriented Filled Arrow**

The parameters \( x \) and \( y \) correspond to the coordinates of the pixel indicated by the "X" in the above figure.

The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

**NOTE:** The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.
DRAWING A FILLED ARROW POINTING TO THE LEFT

**TVI 950 e ADDS Viewpoint**

**Code:** 27 34 y x 00 (1B 22 y x 00 Hex)

**Mnemonic:** ESC " ASCII(y) ASCII(x) NUL

This command draws a filled arrow oriented to the left, as shown in the following figure:

![Figure 47: Left oriented filled arrow](image)

The parameters x and y correspond to the coordinates of the pixel indicated by the "X" in the above figure.

The values of these parameters must range from 0 to a maximum that depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

**NOTE:** The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.

DRAWING GRADUATED AXISES AND GRID

**TVI 950 e ADDS Viewpoint**

**Code:** 27 28 y x 00 ly lx 00 gr dt (1B 1C y x 00 ly lx 00 gr dt Hex)

**Mnemonic:** ESC FS ASCII(y) ASCII(x) NUL ...

... ASCII(ly) ASCII(lx) NUL ASCII(gr) ASCII(dt)

This command allows to create quickly the cartesian axises with graduation and a grid on the display.

- **x**: Vertical coordinate of the cartesian axises origin; must be in the range 3+max.
- **y**: Horizontal coordinate of the cartesian axises origin; must be in the range 3+max.
- **lx**: Length of X axis; must be in the range 1+max.
- **ly**: Length of Y axis; must be in the range 1+max.
- **gr**: 0 = Grid hidden; 1 = Grid visible.
- **dt**: Distance between two graduations; must be in the rangee 3÷255 (3÷FF Hex).

The value of max parameter depends on the size of the picture to draw; as it must be always possible to show it completely in the display area.

**NOTE:** The code 00 (NUL) put at the end of the command sequence is present for compatibility to future versions and must be always sent for a correct execution of this command.
MISCELLANEOUS COMMANDS

Here follows a list of the several commands that manage various functions, available according to the installed firmware.

BUZZER TIMED ACTIVATION

TVI 950 e ADDS Viewpoint

| Code: | 07 |
| Mnemonic: | BEL |

The buzzer on the RKD LT board is activated for about one tenth of second.

PRESENCE CODE REQUEST

TVI 950 e ADDS Viewpoint

| Code: | 27  78 | (1B  4E Hex) |
| Mnemonic: | ESC  N |

After having received this command, the RKD LT board send the code 170 (AA Hex) to the master control unit.

This allows to know quickly whether the board is working correctly or it has terminated its initialization after a Power ON or a Reset.

EEPROM DATA WRITE

TVI 950 e ADDS Viewpoint

| Code: | 27  54  ind L  ind H  n  byte 1 ... byte n |
| Mnemonic: | ESC  6  ASCII(ind L) ASCII(ind H) ASCII(n) ... ASCII(byte 1)...ASCII(byte n) |

This command stores a block of n bytes in the EEPROM on the RKD LT board, starting from the address ind. This last is a 16 bits parameter that must be sent as two bytes, respectively the low byte (bits 7÷0) first and the high byte (bits 15÷8) next; its value must range from 32 (20 Hex) to a maximum that depends on the size of the EEPROM installed (24c04 = 512 bytes, 24c16 = 2048 bytes) and must allow to store wholly the n bytes sent.

Data to be written must be in the range 0÷255 (0÷FF Hex).

If, for example, the User wants to store the bytes 24, 65 and 2 from the address 258, he/she will have to send the sequence: 27  6  2  1  3  24  65  2.
EEPROM DATA READ

**TVI 950 e ADDS Viewpoint**

**Code:** 27 55 ind L ind H n
(1B 37 ind L ind H n Hex)

**Mnemonic:** ESC 7 ASCII(ind L) ASCII(ind H) ASCII(n)

This command reads and returns a block of n bytes from the EEPROM on the RKD LT board, starting from the address ind. This last is a 16 bits parameter that must be sent as two bytes, respectively the low byte (bits 7÷0) first and the high byte (bits 15÷8) next; its value must range from 32 (20 Hex) to a maximum that depends on the size of the EEPROM installed (24c04 = 512 bytes, 24c16 = 2048 bytes) and must allow to store wholly the n bytes sent.

**DIMMING FOR FLUORESCENT DISPLAY**

**TVI 950 e ADDS Viewpoint**

**Code:** 27 33 dimming (1B 21 dimming Hex)

**Mnemonic:** ESC ! ASCII(dimming)

Dimming for Futaba fluorescent display, as indicated by the dimming parameter:

- **dimming:** 00 -> Sets luminosity to 0%.
- 32 (20 Hex) -> Sets luminosity to 20%.
- 48 (30 Hex) -> Sets luminosity to 30%.
- 64 (40 Hex) -> Sets luminosity to 40%.
- 96 (60 Hex) -> Sets luminosity to 60%.
- 112 (70 Hex) -> Sets luminosity to 70%.
- 128 (80 Hex) -> Sets luminosity to 80%.
- 255 (FF Hex) -> Sets luminosity to 100%.

**NOTE**

The luminosity levels available depend on the type of fluorescent display used, please refer to the display manufacturer documentation to know which values can be used.
CUSTOM CHARACTERS

**TVI 950 e ADDS Viewpoint**

*Code:* 27 33 66 \( \text{ind} \) byte 1 ... byte 5

*(1B 21 42 \( \text{ind} \) byte 1 ... byte 5 Hex)*

*Mnemonic:* ESC ! B ASCII(\( \text{ind} \)) ASCII(byte 1) ... ASCII(byte 5)*

This command allows to create custom characters that will be stored starting from the \( \text{ind} \) address in the character memory of the fluorescent display connected to the RKD LT board. The board will display a custom character whenever it will receive its corresponding address \( \text{ind} \).

This command is available only when using Futaba M202SD01BA (20x2 characters “small”), M204SD01AA (20x4 characters), M404SD01BA (40x4 characters) fluorescent displays and compatible.

The values of \( \text{ind} \) depend on the display in use, in detail:

- **M202SD01BA:** \( \text{ind} \) must be in the range \( 248 \div 255 \) (F8÷FF Hex)
- **M204SD01AA:** \( \text{ind} \) must be in the range \( 205 \div 207 \) (CD÷CF Hex)
- **M404SD01BA:** \( \text{ind} \) must be in the range \( 252 \div 254 \) (FC÷FE Hex)

The remaining 5 bytes of the sequence contain the pattern of the custom character; for further informations about the meaning of their bits please refer to the specific documentation of the display.
EXTERNAL CARDS FOR RKD LT

RKD LT board can be connected to a wide range of grifo® cards, all the CPU (GPC® serie) cards can act as master control card, increasing its versatility. A complete set of modules makes easier the connection to the on board devices. Hereunder follows a short description for some of these boards.

**SPB 04-08**
Switch Power BUS 4-8 slots
Motherboard with 4-8 slots industrial BUS ABACO®; pitch 4 TE; standard power supply connectors; termination resistors; F type connector for SPC xxx power supply board; holes for rack mounting.

**ABB 05**
ABACO® Block BUS 5 slots
5 slots ABACO® mother board with Power Supply. Double power supply built-in; 5Vdc 2,5A section for powering the on-board logic; second section at 24Vdc 400mA galvanically coupled, for the optocoupled input lines. Auxiliary connector for ABACO® I/O BUS. Housing with hooks for DIN Ω rails.

**GPC® 51 - GPC® 51D**
General Purpose Controller 51 family
11 MHz 51 INTEL or 22 MHz 320 DALLAS µP BASIC type included; 16/24 TTL I/O lines; 1 or 2 RS 232 lines; Buzzer; RTC and 32K RAM backed Lithium battery; EPROM and EEPROM programmer; readable dip switch; 3 Timer Counter; 4 11 bit A/D lines and Keyboard Display Controller.

**GPC® 553**
General Purpose Controller 80C552 (3 TYPE)
80C552 µP 33 MHz; 1 RS 232 line; 1 RS 232 or RS 422-485 or Current Loop line; 16 TTL I/O lines; 8 A/D lines at 10 bits; 3 Timer Counter; RTC; 64K EPROM; 64K RAM (32K backed RAM-32K DIL EEPROM); 8K serial EEPROM; 2 PWM lines; Watch dog; 5 readable Dip switch; LCD interface.

**GPC® 188F**
General Purpose Controller 80C188
80C188 µP 20 MHz; 256K FLASH; 256K RAM Lithium battery backed; 8K serial EEPROM; 1 RS 232 line; 1 RS 232 or RS 422-485 or Current Loop line; 24 TTL I/O lines; RTC; 8 A/D lines at 12 bits; Watch dog; 8 Dip switch; 3 Timer Counter.

**GPC® 15A**
General Purpose Controller 84C15
Full CMOS card, 10÷20 MHz 84C15 CPU; 512K EPROM or FLASH; 128K RAM; 8K RAM and RTC backed; 8K serial EEPROM; 1 RS 232 line or RS 422-485 or Current Loop line; 32 or 40 TTL I/O lines; CTC; Watch dog; 2 Dip switches; Buzzer.
FIGURE 48: POSSIBLE CONNECTIONS DIAGRAM

- ITALIAN TECHNOLOGY grifo®
- ALL TYPE CPU
- CUSTOM Keyboard MATRIX 7x8
- Toshiba or Hitachi controller GRAPHIC DISPLAY
- FUTABA Fluorescent DISPLAY
- FBC 116 NCS 01 (OPTIONAL)
- ANY MOTHERBOARD TYPE WITH ABACO® BUS
- POWER SUPPLY +5Vdc ONLY
- BG Keyboard 56 keys
- CUSTOM Keyboard MATRIX 7x8
- 8 LEDs common katode
- Serial Line
- RS 232, RS 422, Current Loop
- ANY CPU TYPE GPC® 552, GPC® 15R, etc...
- PC like or Macintosh
- PLC

RKD LT Rel. 5.00
GPC® 150
General Purpose Controller 84C15
CPU Z80 at 16 MHz. Full CMOS card; 512K EPROM or FLASH; 512K RAM; RTC; Backed by external lithium battery; 4M serial FLASH; 1 RS 232 line + 1 RS 232 or RS 422-485 or current loop line; 40 I/O TTL; 2 timer/counter; 2 watch dog; dip switch; EEPROM; 12 bits A/D lines; activity LED.

GPC® 15R
General Purpose Controller 84C15 with Relays
84C15 µP 16 MHz; 1 RS 232 line; 1 RS 232 or RS 422-485 or Current Loop line; 24 TTL I/O lines; 16 Opto-in 8 Relays; 4 Opto Coupled Timer Counter; RTC; 512K EPROM or FLASH; 512K backed RAM; 8K serial EEPROM; 8K Backed RAM Modul; Buzzer; Watch dog; 12 readable Dip switch; LCD interface.

GPC® 153
General Purpose Controller 84C15 (3 TYPE)
84C15 µP 16 MHz; 1 RS 232 line; 1 RS 232 or RS 422-485 or Current Loop line; 16 TTL I/O lines; 8 A/D lines at 12 bits; 4 Timer Counter; RTC; 512K EPROM or FLASH; 512K backed RAM; 8K serial EEPROM; Buzzer; Watch dog; 8 readable Dip switch; LCD interface.

GPC® 884
General Purpose Controller 80C188ES (4 TYPE)
80C188ES µP 40 MHz; 1 RS 232 line; 1 RS 232 or RS 422-485; 16 TTL I/O lines; 11 A/D lines at 12 bits; 3 Timer Counter; RTC; 512K EPROM or FLASH; 512K backed RAM; 8K serial EEPROM; Watch dog; 1 readable Dip switch; LCD interface.

GPC® 114
General Purpose Controller 68HC11 (4 TYPE)
68HC11 µP 16 MHz; 1 RS 232 or RS 422-485; 18 TTL I/O lines; 8 A/D lines at 8 bits; 3 Timer Counter; RTC; 32K EPROM; 32K backed RAM; 512 DIL EEPROM; Watch dog; 1 readable Dip switch; LCD interface.

NCS 01
New Connector Support
Serial communication support board; standard 16 pins RS 232 ABACO® serial connector; quick extraction connectors to connect directly to the field; 2 D25 connectors with standard RS 232 pin out; selectable DCE/DTE.

FBC xxx
Flat BLOCK Contact
This interconnection system “wires to board” allows the connection to many types of flat cable connectors to a terminal for external connections. Other interfacing for most popular connectors such as D, mini DIN, ACCESS.bus™, and so on, are available. Connection for DIN C Type and Ω rails.

BG Keyboard
56 keys keyboard on a 7x8 matrix; coloured and serigraphed keys; LED; Buzzer; comfortable plastic container; 20 pins flat to connect directly to RKD LT and MDU-RKD board.
IAF 404
Interface Adapter Futaba 40x4
Interface module for 40x4 characters Futaba M404SD01BA fluorescent displays; 20 pins connector to connect directly to RKD LT and MDU-RKD board; 20 pins standard I/O ABACO® connector; 26 pins connector to connect directly the display; screw terminal connector for power supply; holes for mounting.
Here follow the commands summarizing tables, complete of command codes, for the TRKDAL and TRKDGL, considering their several operational modes.

<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>Cursor Absolut Positioning</td>
<td>27</td>
<td>89 r c</td>
<td>1B 59 r c ESC Y ASCII(r) ASCII(c)</td>
</tr>
<tr>
<td>CLEAR PAGE</td>
<td>12</td>
<td>0C</td>
<td>FF</td>
</tr>
<tr>
<td>CLEAR END OF LINE</td>
<td>27</td>
<td>75</td>
<td>1B 4B ESC K</td>
</tr>
<tr>
<td>Cursor Disabled</td>
<td>27</td>
<td>80</td>
<td>1B 50 ESC P</td>
</tr>
<tr>
<td>Cursor Enabled And Fixed</td>
<td>27</td>
<td>79</td>
<td>1B 4F ESC O</td>
</tr>
<tr>
<td>Cursor Enabled And Flashing</td>
<td>27</td>
<td>77</td>
<td>1B 4D ESC M</td>
</tr>
<tr>
<td>BUZZER Timed Activation</td>
<td>07</td>
<td>07</td>
<td>BEL</td>
</tr>
<tr>
<td>ECHO Activation On Auxiliary Serial Port</td>
<td>27</td>
<td>51</td>
<td>1B 33 ESC 3</td>
</tr>
<tr>
<td>ECHO Deactivation On Auxiliary Serial Port</td>
<td>27</td>
<td>52</td>
<td>1B 34 ESC 4</td>
</tr>
<tr>
<td>Character Visualization</td>
<td>32-255</td>
<td>20-FF</td>
<td>SP+ASCII(255)</td>
</tr>
<tr>
<td>Screenshot Memorization In EEPROM</td>
<td>27</td>
<td>33 67 n.scr. car. 1 ... car. n</td>
<td>1B 21 43 n.scr. car. 1 ... car. n</td>
</tr>
<tr>
<td>Screenshot Visualization From EEPROM</td>
<td>27</td>
<td>33 68 n.scr.</td>
<td>1B 21 44 n.scr.</td>
</tr>
<tr>
<td>Screenshot Read From EEPROM</td>
<td>27</td>
<td>33 69 n.scr.</td>
<td>1B 21 45 n.scr.</td>
</tr>
<tr>
<td>External LEDs Activation</td>
<td>27</td>
<td>50 mask</td>
<td>1B 32 mask</td>
</tr>
<tr>
<td>Dimming For Fluorescent Displays</td>
<td>27</td>
<td>33 dimming</td>
<td>1B 21 dimming</td>
</tr>
<tr>
<td>Custom Characters</td>
<td>27</td>
<td>33 ind byte 1 ... byte 5</td>
<td>1B 21 42 ind byte 1 ... byte 5</td>
</tr>
</tbody>
</table>

**Figure A1: TRKDAL Commands (Adds Viewpoint) Table**
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOME</td>
<td>28</td>
<td>1C</td>
<td>FS</td>
</tr>
<tr>
<td>CURSOR LEFT</td>
<td>08</td>
<td>08</td>
<td>BS</td>
</tr>
<tr>
<td>CURSOR RIGHT</td>
<td>09</td>
<td>09</td>
<td>HT</td>
</tr>
<tr>
<td>CURSOR DOWN</td>
<td>10</td>
<td>0A</td>
<td>LF</td>
</tr>
<tr>
<td>CURSOR UP</td>
<td>11</td>
<td>0B</td>
<td>VT</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>Cursor Absolut Positioning</td>
<td>27</td>
<td>61 r c</td>
<td>ESC ASCII(r) ASCII(c)</td>
</tr>
<tr>
<td>CLEAR PAGE</td>
<td>12</td>
<td>0C</td>
<td>FF</td>
</tr>
<tr>
<td>CLEAR END OF LINE</td>
<td>23</td>
<td>17</td>
<td>ETB</td>
</tr>
<tr>
<td>Cursor Disabled</td>
<td>27</td>
<td>80</td>
<td>ESC P</td>
</tr>
<tr>
<td>Cursor Enabled And Fixed</td>
<td>27</td>
<td>79</td>
<td>ESC O</td>
</tr>
<tr>
<td>Cursor Enabled And Flashing</td>
<td>27</td>
<td>77</td>
<td>ESC M</td>
</tr>
<tr>
<td>BUZZER Timed Activation</td>
<td>07</td>
<td>07</td>
<td>BEL</td>
</tr>
<tr>
<td>ECHO Activation On Auxiliary Serial Port (bistable)</td>
<td>27</td>
<td>33 65</td>
<td>ESC ! A</td>
</tr>
<tr>
<td>ECHO Deactivation On Auxiliary Serial Port (bistable)</td>
<td>27</td>
<td>33 65</td>
<td>ESC ! A</td>
</tr>
<tr>
<td>Character Visualization</td>
<td>32-255</td>
<td>20-FF</td>
<td>SP+ASCII(255)</td>
</tr>
<tr>
<td>Screenshot Memorization In EEPROM</td>
<td>27</td>
<td>33 67 n.scr. car. 1 ... car. n</td>
<td>1B 21 43 n.scr. car. 1 ... car. n</td>
</tr>
<tr>
<td>Screenshot Visualization From EEPROM</td>
<td>27</td>
<td>33 68 n.scr.</td>
<td>1B 21 44 n.scr.</td>
</tr>
<tr>
<td>Screenshot Read From EEPROM</td>
<td>27</td>
<td>33 69 n.scr.</td>
<td>1B 21 45 n.scr.</td>
</tr>
<tr>
<td>External LEDs Activation</td>
<td>27</td>
<td>50 mask</td>
<td>1B 32 mask</td>
</tr>
<tr>
<td>Dimming For Fluorescent Displays</td>
<td>27</td>
<td>33 dimming</td>
<td>1B 21 dimming</td>
</tr>
<tr>
<td>Custom Characters</td>
<td>27</td>
<td>33 66 ind byte 1 ... byte 5</td>
<td>1B 21 42 ind byte 1 ... byte 5</td>
</tr>
</tbody>
</table>

**FIGURE A2: TRKDAF COMMANDS (TVI 950) table**
<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>Cursor Absolut Positioning</td>
<td>27</td>
<td>89 r c</td>
<td>ESC Y ASCII(r) ASCII(c)</td>
</tr>
<tr>
<td>CLEAR PAGE</td>
<td>12</td>
<td>0C</td>
<td>FF</td>
</tr>
<tr>
<td>CLEAR END OF LINE</td>
<td>27</td>
<td>75</td>
<td>ESC K</td>
</tr>
<tr>
<td>Cursor Disabled</td>
<td>27</td>
<td>80</td>
<td>ESC P</td>
</tr>
<tr>
<td>Cursor Enabled And Fixed</td>
<td>27</td>
<td>79</td>
<td>ESC O</td>
</tr>
<tr>
<td>Cursor Enabled And Flashing</td>
<td>27</td>
<td>77</td>
<td>ESC M</td>
</tr>
<tr>
<td>Flashing Block Cursor</td>
<td>27</td>
<td>81</td>
<td>ESC Q</td>
</tr>
<tr>
<td>Reverse Attribute Activation</td>
<td>27</td>
<td>48 80 14</td>
<td>ESC 0 P SO</td>
</tr>
<tr>
<td>Reverse Attribute Deactivation</td>
<td>27</td>
<td>48 80 15</td>
<td>ESC 0 P SI</td>
</tr>
<tr>
<td>BUZZER Timed Activation</td>
<td>07</td>
<td>07</td>
<td>BEL</td>
</tr>
<tr>
<td>ECHO Activation On Auxiliary Serial Port</td>
<td>27</td>
<td>51</td>
<td>1B 33</td>
</tr>
<tr>
<td>ECHO Deactivation On Auxiliary Serial Port</td>
<td>27</td>
<td>52</td>
<td>1B 34</td>
</tr>
<tr>
<td>Character Visualization</td>
<td>32+255</td>
<td>20+FF</td>
<td>SP-ASCII(255)</td>
</tr>
<tr>
<td>Screenshot Visualization</td>
<td>27 122</td>
<td>n.scr. H n.scr. L</td>
<td>1B 7A ASCII(n.scr. H) ASCII(n.scr. L)</td>
</tr>
<tr>
<td>EEPROM Data Write</td>
<td>27 54</td>
<td>ind L ind H n</td>
<td>1B 36 ASCII(ind L) ASCII(ind H) ASCII(n)</td>
</tr>
<tr>
<td>EEPROM Data Read</td>
<td>27 55</td>
<td>ind L ind H n</td>
<td>1B 37 ASCII(ind L) ASCII(ind H) ASCII(n)</td>
</tr>
<tr>
<td>External LEDs Activation</td>
<td>27 50</td>
<td>mask</td>
<td>1B 32 mask</td>
</tr>
<tr>
<td>External LEDs Reversed Activation</td>
<td>27 53</td>
<td>mask</td>
<td>1B 35 mask</td>
</tr>
<tr>
<td>Keyclick On Local Buzzer Activation</td>
<td>27 56</td>
<td>1B 38</td>
<td>ESC 8</td>
</tr>
<tr>
<td>Keyclick On Local Buzzer Deactivation</td>
<td>27 57</td>
<td>1B 39</td>
<td>ESC 9</td>
</tr>
<tr>
<td>Keyclick On BG Keyboard Activation</td>
<td>27 58</td>
<td>1B 3A</td>
<td>ESC :</td>
</tr>
<tr>
<td>Keyclick On BG Keyboard Deactivation</td>
<td>27 59</td>
<td>1B 3B</td>
<td>ESC :</td>
</tr>
<tr>
<td>Presence Code Request</td>
<td>27 78</td>
<td>1B 4E</td>
<td>ESC N</td>
</tr>
</tbody>
</table>

**Figure A3: TRKDGL Commands for Alphanumeric Mode (Adds Viewpoint) Table**
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOME</td>
<td>28</td>
<td>1C</td>
<td>FS</td>
</tr>
<tr>
<td>CURSOR LEFT</td>
<td>08</td>
<td>08</td>
<td>BS</td>
</tr>
<tr>
<td>CURSOR RIGHT</td>
<td>09</td>
<td>09</td>
<td>HT</td>
</tr>
<tr>
<td>CURSOR DOWN</td>
<td>10</td>
<td>0A</td>
<td>LF</td>
</tr>
<tr>
<td>CURSOR UP</td>
<td>11</td>
<td>0B</td>
<td>VT</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>Cursor Absolute Positioning</td>
<td>27</td>
<td>61 r c</td>
<td>ESC = ASCII(r) ASCII(c)</td>
</tr>
<tr>
<td>CLEAR PAGE</td>
<td>12</td>
<td>0C</td>
<td>FF</td>
</tr>
<tr>
<td>CLEAR END OF LINE</td>
<td>23</td>
<td>17</td>
<td>ETB</td>
</tr>
<tr>
<td>Cursor Disabled</td>
<td>27</td>
<td>80</td>
<td>ESC P</td>
</tr>
<tr>
<td>Cursor Enabled And Fixed</td>
<td>27</td>
<td>79</td>
<td>ESC O</td>
</tr>
<tr>
<td>Cursor Enabled And Flashing</td>
<td>27</td>
<td>77</td>
<td>ESC M</td>
</tr>
<tr>
<td>Flashing Block Cursor</td>
<td>27</td>
<td>81</td>
<td>ESC Q</td>
</tr>
<tr>
<td>Reverse Attribute Activation</td>
<td>27</td>
<td>33 50</td>
<td>ESC ! 2</td>
</tr>
<tr>
<td>Reverse Attribute Deactivation</td>
<td>27</td>
<td>33 48</td>
<td>ESC ! 0</td>
</tr>
<tr>
<td>BUZZER Timed Activation</td>
<td>07</td>
<td>07</td>
<td>BEL</td>
</tr>
<tr>
<td>ECHO Activation</td>
<td>27</td>
<td>33 65</td>
<td>ESC ! A</td>
</tr>
<tr>
<td>On Auxiliary Serial Port</td>
<td>27</td>
<td>33 65</td>
<td>ESC ! A</td>
</tr>
<tr>
<td>ECHO Deactivation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Character Visualization</td>
<td>32</td>
<td>255</td>
<td>SP+ASCII(255)</td>
</tr>
<tr>
<td>Screenshot Visualization</td>
<td>27</td>
<td>122</td>
<td>ASCII(n.scr. H) ASCII(n.scr. L)</td>
</tr>
<tr>
<td>EEPROM Data Write</td>
<td>27</td>
<td>54 ind L ind H n byte 1 ... byte n</td>
<td>ESC 6 ASCII(ind L) ASCII(ind H) ASCII(n) ASCII(byte 1)...ASCII(byte n)</td>
</tr>
<tr>
<td>EEPROM Data Read</td>
<td>27</td>
<td>54 ind L ind H n</td>
<td>1B 37</td>
</tr>
<tr>
<td>External LEDs Activation</td>
<td>27</td>
<td>50 mask</td>
<td>1B 32 mask</td>
</tr>
<tr>
<td>External LEDs Reversed</td>
<td>27</td>
<td>53 mask</td>
<td>1B 35 mask</td>
</tr>
<tr>
<td>Keyclick On Local Buzzer</td>
<td>27</td>
<td>56</td>
<td>1B 38</td>
</tr>
<tr>
<td>Keyclick On Local Buzzer</td>
<td>27</td>
<td>57</td>
<td>1B 39</td>
</tr>
<tr>
<td>Keyclick On BG Keyboard</td>
<td>27</td>
<td>58</td>
<td>1B 3A</td>
</tr>
<tr>
<td>Keyclick On BG Keyboard</td>
<td>27</td>
<td>59</td>
<td>1B 3B</td>
</tr>
<tr>
<td>Presence Code Request</td>
<td>27</td>
<td>78</td>
<td>1B 4E</td>
</tr>
</tbody>
</table>

**Figure A4: TRKDGL Commands for Alphanumeric Mode (TVI 950) Table**
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURSOR DOWN</td>
<td>10</td>
<td>0A</td>
<td>LF</td>
</tr>
<tr>
<td>CARRIAGE RETURN</td>
<td>13</td>
<td>0D</td>
<td>CR</td>
</tr>
<tr>
<td>Character Cursor Placement</td>
<td>27 00 r c</td>
<td>0B 00 r c</td>
<td>ESC NUL ASCII(r) ASCII(c)</td>
</tr>
<tr>
<td>Cursor Absolut Positioning</td>
<td>27 06 y x 00</td>
<td>0B 06 y x 00</td>
<td>ESC ACK ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>CLEAR PAGE</td>
<td>12</td>
<td>0C</td>
<td>FF</td>
</tr>
<tr>
<td>Characters Zoom Setting</td>
<td>27 01 zoom</td>
<td>0B 01 zoom</td>
<td>ESC SOH ASCII(zoom)</td>
</tr>
<tr>
<td>Characters Elongation Setting</td>
<td>27 07 elong</td>
<td>0B 07 elong</td>
<td>ESC BEL ASCII(elong)</td>
</tr>
<tr>
<td>Characters Write Setting</td>
<td>27 10 dir</td>
<td>0B 0A dir</td>
<td>ESC LF ASCII(dir)</td>
</tr>
<tr>
<td>Reverse Attribute Setting</td>
<td>27 09 reverse</td>
<td>0B 09 reverse</td>
<td>ESC HT ASCII(reverse)</td>
</tr>
<tr>
<td>BUZZER Timed Activation</td>
<td>07</td>
<td>07</td>
<td>BEL</td>
</tr>
<tr>
<td>Character Visualization</td>
<td>32~126</td>
<td>20~7E</td>
<td>SP~</td>
</tr>
<tr>
<td>String Visualization</td>
<td>27 08 car. 1 ... car. 13</td>
<td>0B 08 car. 1 ... car. 0D</td>
<td>ESC BS ASCII(car. 1)...ASCII(car. n) CR</td>
</tr>
<tr>
<td>EEPROM Data Write</td>
<td>27 54 ind L ind H n byte 1 ... byte n</td>
<td>0B 36 ind L ind H n byte 1 ... byte n</td>
<td>ESC 6 ASCII(ind L) ASCII(ind H) ASCII(n) ASCII(byte 1)...ASCII(byte n)</td>
</tr>
<tr>
<td>EEPROM Data Read</td>
<td>27 55 ind L ind H n</td>
<td>0B 37 ind L ind H n</td>
<td>ESC 7 ASCII(ind L) ASCII(ind H) ASCII(n)</td>
</tr>
<tr>
<td>External LEDs Activation</td>
<td>27 50 mask</td>
<td>0B 32 mask</td>
<td>ESC 2 ASCII(mask)</td>
</tr>
<tr>
<td>External LEDs Reversed Activation</td>
<td>27 53 mask</td>
<td>0B 35 mask</td>
<td>ESC 5 ASCII(mask)</td>
</tr>
<tr>
<td>Keyclick On Local Buzzer Activation</td>
<td>27 56</td>
<td>0B 38</td>
<td>ESC 8</td>
</tr>
<tr>
<td>Keyclick On Local Buzzer Deactivation</td>
<td>27 57</td>
<td>0B 39</td>
<td>ESC 9</td>
</tr>
<tr>
<td>Keyclick On BG Keyboard Activation</td>
<td>27 58</td>
<td>0B 3A</td>
<td>ESC :</td>
</tr>
<tr>
<td>Keyclick On BG Keyboard Deactivation</td>
<td>27 59</td>
<td>0B 3B</td>
<td>ESC :</td>
</tr>
<tr>
<td>Presence Code Request</td>
<td>27 78</td>
<td>0B 4E</td>
<td>ESC N</td>
</tr>
<tr>
<td>One Pixel Visualization</td>
<td>27 29 y x 00</td>
<td>0B 1D y x 00</td>
<td>ESC GS ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>Drawing A Line</td>
<td>27 03 y1 x1 00 y2 x2 00</td>
<td>0B 03 y1 x1 00 y2 x2 00</td>
<td>ESC ETX ASCII(y1) ASCII(x1) NUL ASCII(y2) ASCII(x2) NUL</td>
</tr>
<tr>
<td>Drawing A Rectangle</td>
<td>27 02 y1 x1 00 y2 x2 00</td>
<td>0B 02 y1 x1 00 y2 x2 00</td>
<td>ESC STX ASCII(y1) ASCII(x1) NUL ASCII(y2) ASCII(x2) NUL</td>
</tr>
<tr>
<td>Drawing A Filled Rectangle</td>
<td>27 04 y1 x1 00 y2 x2 00</td>
<td>0B 04 y1 x1 00 y2 x2 00</td>
<td>ESC EOT ASCII(y1) ASCII(x1) NUL ASCII(y2) ASCII(x2) NUL</td>
</tr>
</tbody>
</table>

**FIGURE A5: TRKDGL COMMANDS FOR GRAPHIC MODE TABLE 1**
<table>
<thead>
<tr>
<th>COMMAND</th>
<th>CODE</th>
<th>HEX CODE</th>
<th>MNEMONIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Drawing A Circumference</td>
<td>27 05</td>
<td>1B 05</td>
<td>ESC ENQ ASCII(y) ASCII(x) NUL ASCII(r) ASCII(arc.in.) ASCII(n.arc.)</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCI(r) ASCII(arc.in.) ASCII(n.arc.)</td>
</tr>
<tr>
<td></td>
<td>r</td>
<td>r</td>
<td></td>
</tr>
<tr>
<td>Drawing A Circumference Whose Ray Is 3 Pixels</td>
<td>27 39</td>
<td>1B 27</td>
<td>ESC ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCI(r) ASCII(arc.in.) ASCII(n.arc.)</td>
</tr>
<tr>
<td>Drawing A Filled Circumference Whose Ray Is 3 Pixels</td>
<td>27 38</td>
<td>1B 26</td>
<td>ESC &amp; ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCI(r) ASCII(arc.in.) ASCII(n.arc.)</td>
</tr>
<tr>
<td>Drawing An Arrow Pointing To The Top</td>
<td>27 37</td>
<td>1B 25</td>
<td>ESC %</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>Drawing An Arrow Pointing To The Bottom</td>
<td>27 33</td>
<td>1B 21</td>
<td>ESC !</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>Drawing An Arrow Pointing To The Right</td>
<td>27 31</td>
<td>1B 1F</td>
<td>ESC US</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>Drawing An Arrow Pointing To The Left</td>
<td>27 35</td>
<td>1B 23</td>
<td>ESC #</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>Drawing A Filled Arrow Pointing To The Top</td>
<td>27 36</td>
<td>1B 24</td>
<td>ESC $</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>Drawing A Filled Arrow Pointing To The Bottom</td>
<td>27 32</td>
<td>1B 20</td>
<td>ESC SP</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>Drawing A Filled Arrow Pointing To The Right</td>
<td>27 30</td>
<td>1B 1E</td>
<td>ESC RS</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>Drawing A Filled Arrow Pointing To The Left</td>
<td>27 34</td>
<td>1B 22</td>
<td>ESC &quot;</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td>Drawing Graduated Axises And Grid</td>
<td>27 28</td>
<td>1B 1C</td>
<td>ESC FS</td>
</tr>
<tr>
<td></td>
<td>y  x  00</td>
<td>y  x  00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td></td>
<td>ly  lx 00</td>
<td>ly  lx 00</td>
<td>ASCII(y) ASCII(x) NUL</td>
</tr>
<tr>
<td></td>
<td>gr  dt</td>
<td>gr  dt</td>
<td>ASCII(gr) ASCII(dt)</td>
</tr>
</tbody>
</table>

**Figure A6: TRKDGL Commands for Graphic Mode Table 2**
APPENDIX B: ALPHABETICAL INDEX

A

ABACO® BUS 2, 4, 8, 22, 30
ABSOLUTE CURSOR PLACEMENT, command 52
ADDS VIEWPOINT 6, 38, 42
ALPHANUMERIC DISPLAY 2, 5, 8, 20, 36
ALPHANUMERIC SCREENSHOTS MANAGEMENT COMMANDS 49
AUTOREPEAT 2, 6, 38
AUXILIARY SERIAL PORT MANAGEMENT COMMANDS 45

B

BAUD RATE 2, 4, 8, 35, 37
BG KEYBOARD 2, 4, 5, 6, 8, 12, 14, 33, 38, 40, 47
BOARD CONFIGURATION 33
BUZZER 2, 4, 5, 8, 40, 47, 61
BUZZER TIMED ACTIVATION, command 61

C

CARD VERSION AND FIRMWARE 1
CARRIAGE RETURN, command 41
CARRIAGE RETURN + LINE FEED, command 42
CHARACTER CURSOR PLACEMENT, command 52
CHARACTERS ATTRIBUTES MANAGEMENT COMMANDS 46
CHARACTERS DELETION COMMANDS 43
CHARACTERS REPRESENTATION, command 40
CHARACTERS WRITE DIRECTION, command 53
CHARACTERS ZOOM SETTING, command 51
CHARACTERS ELONGATION SETTING, command 52
CLEAR END OF LINE, command 43
CLEAR PAGE, command 43
CONNECTORS 9, 15
CN 3 - SERIAL LINES CONNECTOR 16
CN 1 - EXTERNAL LEDS CONNECTOR 10
CN 2 - CONNECTOR FOR BG KEYBOARD 12
J1 - CONNECTOR FOR FLUORESCENT ALPHNUMERIC DISPLAY 20
J2 - CONNECTOR FOR GRAPHIC LCD DISPLAYS 21
J5 - COMPLEMENTAR CONNECTOR FOR SERIAL LINES 19
K1 - CONNECTOR FOR ABACO® BUS 22
CONSUMPTION 9
CONTRAST 2, 5, 9, 27
CPU 8, 23
CURRENT LOOP 2, 8, 16, 18, 19, 28, 38
CURSOR ABSOLUT POSITIONING, command 42
CURSOR ATTRIBUTES MANAGEMENT COMMANDS 44
CURSOR DISABLED, command 44
CURSOR DOWN, command 41
CURSOR ENABLED AND FIXED, command 44
CURSOR ENABLED AND FLASHING, command 44
CURSOR LEFT, command 40
CURSOR POSITIONING COMMANDS 40
CURSOR RIGHT, command 41
CURSOR UP, command 41
CUSTOM CHARACTERS 63

D
DEVICES 8
DIMMING FOR FLUORESCENT DISPLAY, command 62
DIP SWITCH 2, 4, 8, 15, 33
DRAWING A CIRCUMFERENCE, command 55
DRAWING A CIRCUMFERENCE WHOSE RAY IS 3 PIXELS, command 55
DRAWING A FILLED ARROW POINTING TO THE BOTTOM, command 59
DRAWING A FILLED ARROW POINTING TO THE LEFT, command 60
DRAWING A FILLED ARROW POINTING TO THE RIGHT, command 59
DRAWING A FILLED ARROW POINTING TO THE TOP, command 58
DRAWING A FILLED CIRCUMFERENCE WHOSE RAY IS 3 PIXEL, command 56
DRAWING A FILLED RECTANGLE, command 54
DRAWING A LINE, command 54
DRAWING A RECTANGLE, command 54
DRAWING AN ARROW POINTING TO THE BOTTOM, command 57
DRAWING AN ARROW POINTING TO THE LEFT, command 58
DRAWING AN ARROW POINTING TO THE RIGHT, command 57
DRAWING AN ARROW POINTING TO THE TOP, command 56
DRAWING GRADUATED AXISES AND GRID, command 60

E
ECHO ACTIVATION ON AUXILIARY SERIAL PORT, command 45
ECHO DEACTIVATION ON AUXILIARY SERIAL PORT, command 45
EEPROM 2, 5, 8, 38, 50, 61
EEPROM DATA READ 62
EEPROM DATA WRITE 61
ELECTRIC FEATURES 9
EPROM 2, 6
EXTERNAL CARDS FOR RKD LT 64
EXTERNAL LEDS ACTIVATION, command 48
EXTERNAL LEDS MANAGEMENT COMMANDS 48
EXTERNAL LEDS REVERSED ACTIVATION, command 48

F
FLASHING BLOCK CURSOR, command 44

G
GENERAL FEATURES 2, 8
GRAPHIC MODE COMMANDS 51
I
I/O 2, 8, 30
INSTALLATION 10
INTRODUCTION 1

J
JUMPERS 24
  2 PINS JUMPERS 27
  3 PINS JUMPERS 26
  5 PINS JUMPERS 24

K
KEYBOARD ACQUISITION 38
KEYBOARD MANAGEMENT COMMANDS 47
KEYCLICK 2, 4, 5, 6, 47
KEYCLICK ON BG KEYBOARD ACTIVATION, command 47
KEYCLICK ON BG KEYBOARD DEACTIVATION, command 47
KEYCLICK ON LOCAL BUZZER ACTIVATION, command 47
KEYCLICK ON LOCAL BUZZER DEACTIVATION, command 47

L
LCD GRAPHIC DISPLAY 2, 5, 8, 21, 27, 33, 34
LED 40
LEDS 2, 5, 8, 9, 10, 48

M
MATRIX KEYBOARD 2, 4, 8, 12, 38, 40
MISCELLANEOUS COMMANDS 61

O
ONE PIXEL VISUALIZATION, command 53

P
PARALLEL COMMUNICATION 2, 31, 32, 35, 37, 38
PARALLEL INTERFACE 4, 8
PARALLEL INTERFACE DESCRIPTION 30
PHYSICAL FEATURES 9
POWER SUPPLY 2, 9
PRESENCE CODE REQUEST, command 61

R
RECEPTION BUFFER 38
RELATIVE HUMIDITY 9
REVERSE ATTRIBUTE ACTIVATION, command 46
REVERSE ATTRIBUTE DEACTIVATION, command 46
REVERSE ATTRIBUTE SETTING, command 51
RKD_EDIT 6, 49
RS 232 2, 4, 8, 16, 17, 19, 28, 38
RS 422 2, 4, 8, 16, 17, 19, 28, 38
RS 422 TERMINATION CIRUCIT 9
S
SCREENSHOT MEMORIZATION IN EEPROM, command 50
SCREENSHOT READ FROM EEPROM, command 51
SCREENSHOT VISUALIZATION, command 49
SCREENSHOT VISUALIZATION FROM EEPROM, command 50
SERIAL COMMUNICATION 2, 8, 28, 35, 38
SIZE 2, 9
SOFTWARE DESCRIPTION 33
STRING VISUALIZATION 53

T
TECHNICAL FEATURES 8
TEMPERATURE RANGE 9
TRIMMER 15
TRKDAF 6, 36, 37, 40, 49
TRKDGL 5, 33, 35, 40, 46, 47, 49
TVI 950 6, 38

W
WEIGHT 9