

KEYPAD

PORTABLE ONE-HAND TERMINAL FOR **italora** PRINTERS

OPERATIVE MANUAL



Via Strina 8 - 20082 Binasco (Mi) - Italy

Phone: ++39.02.90092074 – Fax: ++39.02.9055461

Web site:

www.italora.it

email:

sales@italora.it

Generals Features

Keypad is a product able to program **italora** printers without a direct connection to PC, thanks to internal (E²PROM) and extended memory devices (RamCard or Flash) where you may store programs for printers. You could program RamCard/Flash directly by a PC and Keypad, you don't need an additional add-on board inside the PC.

Keypad, provided with an alphanumeric keyboard, allows you the activation of internal printer layouts, data input for variable fields composition, logo activation and batch printing of a labels number as preferred. Included in Keypad firmware there are MACRO interpreter and PLU (Product Look Up) capabilities; they allow you to print labels directly from a serial device (i.e. scales), without any firmware modification to device itself.

The Keypad display helps the user showing and explaining all the operations to do with built-in messages; a setup function allows to select language, start mode and serial line (to PC) parameters.

Technical Specifications

- Keyboard with 28 keys, 10 alphanumerics, 10 functions and 8 specials
- Liquid Crystal Display (LCD): 2 lines and 16 columns.
- 3 Serial Ports:
 - * one RS232 port with a 25 pins male connector for **printer connection**
 - * one port, working like RS232 or RS485 or TTL, with a 9 pins male connector for the communication to PC
 - * one auxiliary port CURRENT LOOP 20mA, provided with a modular 6-poles phone-like socket
- Depending on the model, you may have one of the following two options:
 - * RamCard 68-Pin PCMCIA Slot. SRAM up to 16 Mbyte supported.
 - * Internal 512 Kbyte permanent Flash memory
- Internal permanent memory (2Kbyte E²PROM)
- Co-axial socket for 5 V DC power supply.
- Power supply, 5V-DC 100mA, ~220 / ~240 V supply line



Keypad power supply

Keypad is not equipped with any internal battery so it needs a power supply when connected to a PC. When it is connected to a printer it can get power supply directly from printer itself only if the printer is a last generation model (starting from 01 January 1997).

Features

1. PRINT button to set copies number and run a batch printing.
2. MENU button to set-up Keypad default conditions.
3. RamCard/Flash programming with sequences of printers commands; each sequence is identified by an user programmed name; possibility to download RamCard/Flash data to a PC; possibility to duplicate already existing labels, changing their data (this feature is valid only for RamCard models).
4. Sequences downloading from RamCard/Flash to printer; sequence selection allowed by scrolling available names, automatic seek of the name or specifying the number of sequence to download.
5. Internal memory (E²PROM) programming with table of label layouts stored in printer; each layout has an user-programmed name. For each layout it is possible to specify how many variable fields are active and if they are numeric fields, alphanumeric fields or logo fields.
6. Layout activation, variable data / logo number input and batch copies printing; layout selection allowed by scrolling available layouts names.
7. "PIPE" operating mode.
8. Programming and use of Keypad as a MACRO Interpreter. In this mode of working you can connect Keypad between the printer and a peripheral device (such as a weight indicator).
9. Calculation and printing of result of two data multiplication (i.e. price/Kg * weight); see MACRO chapter.
10. PLU mode and goods database management to create labels for food industry world (composite barcodes, weight input from a scale, calculation of totals as "Weight * Price", management of partial and general totals);

Function keys

F1	Key to select the program mode: then you can scroll what you want to program: E ² PROM, RamCard/Flash or MACRO interpreter, Database or PLU
F2	Key to select the activation of a layout (yet programmed in printer permanent memory) stored in internal Keypad memory and input its variable or logo fields
F3	Key to select the download procedure of RamCard/Flash stored series: you can choose the series by scrolling its name or writing its number. Then you can input variable or logo fields
F4	Key to select the “PIPE” mode
F5	Key to select the “MACRO Interpreter” mode
F6	Key to select the PLU mode for goods database management
F8	Key to insert special parameters for Continuous Film Kit
F9	Reserved key
F10	Printer reset
MENU	Setup of: start mode, language, serial line (to PC) parameters selection, decimal digit, RamCard/Flash manage, batch print counter
PRINT	Choose and print of required labels
SHIFT	To select Uppercase/Lowercase during data input
ESC	Escape
ENTER	Confirm
BKSP	Back space
	Scroll up.
	Scroll down.

Alphanumeric keys

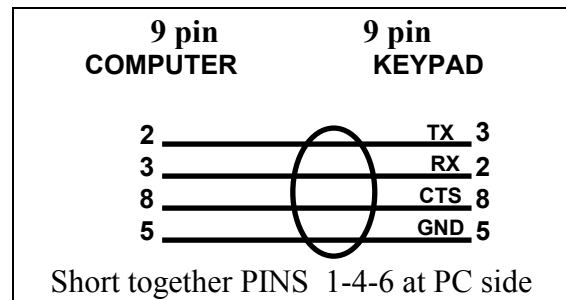
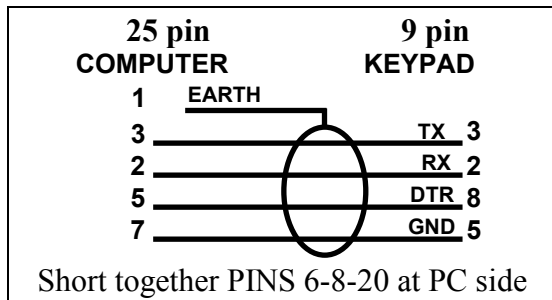
0	With it you can edit: ‘0’	‘ ’ (blank)	‘-’ (minus)	‘,’ (comma)	‘#’
1	With it you can edit: ‘1’	‘s/S’	‘t/T’	‘u/U’	‘%’
2	With it you can edit: ‘2’	‘v/V’	‘w/W’	‘x/X’	‘&’
3	With it you can edit: ‘3’	‘y/Y’	‘z/Z’	‘.’ (point mark)	‘”
4	With it you can edit: ‘4’	‘j/J’	‘k/K’	‘l/L’	‘*’ (asterisk)
5	With it you can edit: ‘5’	‘m/M’	‘n/N’	‘o/O’	‘+’ (plus)
6	With it you can edit: ‘6’	‘p/P’	‘q/Q’	‘r/R’	‘/’ (slash)
7	With it you can edit: ‘7’	‘a/A’	‘b/B’	‘c/C’	‘.’
8	With it you can edit: ‘8’	‘d/D’	‘e/E’	‘f/F’	‘°’ (degree)
9	With it you can edit: ‘9’	‘g/G’	‘h/H’	‘i/I’	‘=’

Programming steps

Keypad can be programmed by a PC with following steps:

1. Write a text file with data you want to store in Keypad.
2. Set PC COM port by typing (in DOS) “MODE COMx *baud,parity,data,1,P*” with:
x Number of used PC serial port
baud 1200, 2400, 4800, 9600 or 19200
parity N, E or O
data 8 or 7
3. Select on Keypad (with MENU key) the same serial line parameters you have selected on PC.
4. On Keypad press F1 and scroll to run the desired data receiving procedure
5. Download text file typing (in DOS) “COPY *filename* COMx”

SERIAL CABLE



Internal memory (E²PROM) programming

Internal memory programming let you store in internal memory of Keypad (E²PROM) a table showing how many layout are stored **inside a printer**; each layout is identified by a name.

Variable fields data types information, such as logo field, can be assigned to each layout in the table.

Variable fields order must be the same in printer layout and Keypad table.

If a printer layout has no variable field you need to introduce just a layout name without variable data information in the table.

Commands for internal memory programming are collected in **sequences**, which are separated each other by a breaker string (“\$\$” = 2 characters + Enter key of PC); only the last sequence must be terminated by the shutting string (“\$!” (2 characters + Enter key of PC).

The same layout IDentifier can be programmed more then one time in the table; i.e. layout ID ‘A’ can be introduced 3 times with different names so the user can drive 3 printers with 3 different ‘A’ layout drawings (one for each printer) with or without variable data or logo fields.

When the ASCII file containing internal memory commands is ready, you have to send it to the printer, following the instruction of ‘Programming Steps’ chapter. Now the internal memory is programmed and you can read it by pressing function key **F2** and confirming by **Enter** key when you have selected the proper layout.

Now let’s take a look of internal memory commands.

Example:

```
A,Electric Motor           // To activate the layout
l,10,15,Motor Logo        // To place the logo
n,r.p.m.:                 // To input a variable data
$!
```

Set up of multiple data output

Syntax: p,HowManyOut

Multiple out command makes Keypad able to produce many outputs with the same input data. After executing of this command, every 'n' or 'a' command produces a number of output programmed as 'p' command parameter. Default is 1.

Example:

```
C,Shorts                   // To activate the layout
a,Color                   // This input produces 1 output
p,2
n,Size                    // This input produces 2 output
$!
```

Fixed texts reading from printer permanent memory

Syntax: f,Description

This command shows on LCD display the Description and let you input index of fixed text to use as variable data. Fixed text available are the ones already stored into printer permanent memory; fixed text you choose is sent to the printer in variable data form. To store fixed text into printer memory see command ?73&... into the printer programming manual.

Example:

```
B,Jackets                 // To activate the layout
f,Model:                 // To use a fixed text from printer
f,Color:                 // To use a fixed text from printer
n,Size:                  // To input a variable data
```

Partial data input

Syntax: j

With this command is possible to skip some data input in the iterations subsequent to the first; in particular after the input of all data fields at the first time, you will be asked for input of only fields placed after the 'j' command.

Example:

```
A,Weighing // To activate the layout
a,Date: // To input a variable data
j
n,Weight: // To input a variable data
$!
```

With this example the user will be asked for input of Date and Weight in the first iteration, and only Weight at the subsequent.

Input of a numeric data valid as number of batch copies too

Syntax: c,Description

With this command it's possible to input a numeric data, which will be send to printer as variable field and as number of labels to print. Labels number includes already the first one, printed just after input of last variable data.

To be valid, this command must be the last of sequence.

Example:

```
C,Oranges
a,From: // Alphanumeric data input
c,How many copies: // Numeric data input and set of n° of batch copies
$!
```

WARNING: "\$\$" and "\$!" can be **ONLY** used as shutting commands. "," is used as separator character, it cannot be used in a layout name, variable field or logo description.

WARNING: When you input *data* for a variable field, Keypad transfers it to printer with command ?25&*data*; so, if on printer is activated a layout with two variable fields, label will be printed after input of the second field from Keypad.

WARNING: When Keypad receives data to be stored in internal memory it clears all previously stored data. You need to organize an ASCII file including all information you want to download to Keypad.

Call another sequence stored in RamCard/Flash

Syntax: s,Sequence_Number (The first Sequence in RamCard/Flash has number 1)

This command is useful to jump into another sequence from the one being read; the called sequence will be transmitted to printer. This command **must be the last in the sequence**: further commands before the shutting string (“\$\$”) will not be processed.

Enter automatically into MACRO mode

Syntax: m

This command is useful to enter into Macro mode directly from a RamCard/Flash sequence

Read variable fields from a DataBase

There are two commands that let you extract data from the database stored into RamCard/Flash.

The first one let you input manually the index of record you want to use.

The second one let you scroll database searching for desired record; records are visualized on display during scroll.

1: Manual input of record index

Syntax: d,Description,Field1,Field2,...,FieldN

The command reads some fields from the RamCard/Flash database and outs them to printer as variable fields. “Description” is shown on LCD display, while record number is asked as input from keyboard; fields index is specified in the command parameters.

Example:

```
READ DATABASE             // Sequence name
...                         // Printer commands
d,Descr. on LCD,0,1,3       // Reading of fields n. 0,1,and 3 of input specified record
$!
```

2: Scrolling records of DataBase

Syntax: D, Description,ENTRY, Field1,Field2,...,FieldN

Description is visualized on display; ENTRY must be always composed by 5 digits and it is the index of first record visualized on display when scroll begins. By pressing UP and DOWN arrows you can move through records. When desired record is visualized on display you must confirm with ENTER key to extract data from fields programmed as parameter of command (Field1,Field2,...,FieldN).

Example:

```
Apples                     // Sequence name
...                         // Printer commands
D,Apples,00025,1,2,3       // Starting from record 25, scroll of database records
$!
```

Set up of multiple data output

Syntax: p,HowManyOut

Multiple out command makes Keypad able to produce many outputs with the same input data. After executing of this command, every 'n' or 'a' command produces a number of output programmed as 'p' command parameter. Default is 1.

Example:

```
SHORTS
... // Printer commands
a,Color // This input produces 1 output
p,2
n,Size // This input produces 2 output
$!
```

Fixed texts reading from printer permanent memory

Syntax: f,Description

This command shows on LCD display the Description and let you input index of fixed text to use as variable data. Fixed text available are the ones already stored into printer permanent memory; fixed text you choose is sent to the printer in variable data form. To store fixed text into printer memory see command ?73&... into the printer programming manual.

Example:

```
B,Jackets // To activate the layout
f,Model: // To use a fixed text from printer
n,Size: // To input a variable data
```

Input of a variable data with a default value (only for RAMCARD version)

Syntax: N,Description,Default for numeric data
A,Description,Default for alphanumeric data
Description : max 16 characters
Default: max 40 characters

This command makes you able to digit, with Keypad, variable data for still activated layout; on display, during input, are shown both 'Description' and 'Default' associated to data. If new data corresponds to default, it's enough to press Enter key to confirm it, otherwise you have to digit whole new data and to press Enter key to confirm. New data is stored into RamCard and it will be proposed as default data at the next input.

Input of a numeric data valid as number of batch copies too

Syntax: c,Description

With this command it's possible to input a numeric data, which will be send to printer as variable field and as number of labels to print. Labels number includes already the first one, printed just after input of last variable data.

To be valid, this command must be the last of sequence.

Example:

C,Oranges

a,From: // Alphanumeric data input

c,How many copies: // Numeric data input and set of n° of batch copies

\$!

WARNING: “\$\$” and “\$!” can be used **ONLY** as shutting commands.

WARNING: When you input *data* for a variable field, Keypad transfers it to printer with command `?25&data`; so, if on printer is activated a layout with two variable fields, label will be printed after input of the second field from Keypad.

WARNING: When Keypad receives data to be stored in RamCard/Flash it clears all previously stored data; so you need to organize an ASCII file including all layouts you want to download to Keypad.

How to send RamCard/Flash contents to a PC

It's possible, with a PC, to read RamCard/Flash contents: that's the way.

1. Connect PC to 9 pins Keypad serial port, with an appropriate cable (see chapter 'Serial cables')
2. On PC side open a terminal (i.e. Windows™ terminal), and set transmission parameters as follow:
 - Right number of serial PC port (COMx)
 - Baud rate = 9600
 - Parity = No one
 - Data Bit = 8
 - Stop Bit = 1
 - Flow Control = Hardware (RTS / CTS)
3. When PC terminal is ready to capture data, press F3 on Keypad
4. Now press F10 on Keypad; RamCard/Flash data are sent to PC. On Keypad display is shown a transmission message; it will disappear at the end of transmission.

How to duplicate a RamCard sequence changing its data

It's possible to duplicate a RamCard sequence, putting the new one after all others, giving it a new name; this feature is useful if source sequence contains the special commands 'N' or 'A' for input of a variable data with a default. Let's see the way.

First of all enter RamCard mode with F3 key, then scroll up/down to the sequence you want to copy; now press F1 key. You'll be asked for the new name to give to the new sequence: input name and confirm with Enter key. Pressing Esc key at the prompt will abort the copy.

NB: This feature is not supported by FLASH models.

How to print a layout

Activate and print a layout already stored in printer memory

To get printed a label stored in printer with or without variable data or logo fields you need to do some short operations.

1. Select transmit procedure of layout activation ID by pressing F2
2. Select the layout you want to activate by Scrolling and press Enter key to confirm sequence or Esc to exit without transmitting anything.
3. Input variable data or number of logo if required.
4. Input number of copies.
5. Loop to step n. 3

Print a layout read from RAMCARD/FLASH

To get printed a label stored in RamCard/Flash you need to do some short operations.

1. Select transmit procedure by pressing F3
2. Choose the desired sequence as follows:
 - Scroll keys (UP and DOWN) to scroll once at a time sequences names. Confirm selection by pressing ENTER key, abort with ESC key
 - Numeric keys to input the sequence number you want to transmit to printer. Confirm selection by pressing ENTER key, abort with ESC key
 - SHIFT key to input the name (maybe only the first letters) of sequence you want to retrieve; when a match is found but it isn't the desired, you can seek forward by pressing DOWN arrow key. Confirm selection by pressing ENTER key, abort with ESC key
3. Input variable data or number of logo if required.
4. Input number of copies.
5. Wait end of transmission.

Examples

Some short samples for internal memory data storing and RamCard/Flash

For INTERNAL MEMORY (E²PROM)

A,MOTOR
n,Serial Number
l,220,150,Motor logo
a,Product descr.
\$\$
B,Label 2
n,Input 4 dgt.
\$\$
A,Without variabl
\$!

For RAMCARD/FLASH

GLASS
?00&
?07&80
?06&-13
?52&10,161,270,2,11;Handle with care
?22&134,244,310,70,2
?16&130,0,318,318,4
?09&2
?10&1
?11&2
?13&2
?52&11,168,183,8,58;5499981284
?16&164,87,76,88,2
?15&132,76,316,2,4
?22&134,80,24,164,2
?22&420,80,24,164,2
?22&178,95,48,72,0
?52&10,191,12,5,11;Fine Glass
?52&10,263,98,2,11;6 Cristal
?52&10,263,136,2,11;glasses
?14&1
\$\$
TEXTILE LABEL
?00&
?07&80
?06&-13
.....
?09&2
?10&1
?11&2
?13&2
?52&11,241,232,0,90;165209
?52&10,242,105,6,11;Cotton 100%
?52&10,256,9,6,11;Textile
?52&10,256,46,6,11;application
?22&208,0,240,90,2
l,500,160,Textile logo
\$\$
ANOTHER LAB.
....
...
\$!

The “PIPE” mode

When you press **F4**, Keypad enter into “PIPE” mode of operation.

In this mode Keypad transmits to the printer (connected to 25 pins serial port) all what it receives in input from 9 pins serial port. So the behaviour of Keypad is like a “tube”, in which all enters one side must exit (unchanged) the other one.

The “PIPE” mode makes possible also to transmit RamCard/Flash commands sequences to printer: if the string

“<CR LF>s,xxx” , where <CR LF> = Carriage Return + Line Feed (ENTER key on PC)
s = command to transmit a RamCard/Flash sequence to printer
, = comma
xxx = number of RamCard/Flash sequence to transmit to printer

is received by 9 pins serial port, then the RamCard/Flash sequence number xxx is transmitted to printer.

When the sequence ends, Keypad goes on with normal pipe transmission. (**Warning:** Some ‘special’ commands maybe present in the RamCard/Flash sequence are not processed.)

There is the possibility to start-up Keypad in “PIPE” mode; you may do it by pressing the **menu** key and scrolling the correct choice.

The “MACRO” mode

General

In the MACRO-INTERPRETER mode it’s possible to program and activate the “record decoding” operation. In this mode Keypad becomes an interface between printer and a non programmable serial device such as a weight/price indicator; the indicator sends data in form of a record to Keypad, it decodes record and extracts the variable fields informations; then sends them to printer in command ?25&... form. There is also the possibility to program MACRO to execute some multiplication (between data from string record or from keyboard input) and to print their results on label in command ?25&... form.

To set up this working mode you must follow the next steps:

- Program labels layouts on printer permanent memory.
- Program MACRO Interpreter on Keypad by following steps reported in “Programming steps” chapter.
- Connect printer to Keypad through 25 pins serial port.
- Connect Keypad to serial device through 9 pins serial port.
- Select (with **MENU** key) on Keypad the same serial parameters of device serial line.
- Press on Keypad **F5** function key to enter MACRO mode and waiting for records (maybe input data from Keypad keyboard if requested).

Records from serial devices may contain some additional information such as which layout to activate on printer and how many batch copies there are to do.

MACRO Interpreter programming

Programming of MACRO-Interpreter allows to:

1. Specify the input record type (fixed length or fixed terminator character).
2. If fixed terminator character, specify the ASCII code of terminator character.
3. If fixed length, specify the number of characters record is made up.

4. Specify the position (into record from serial device) at which Keypad will find the number of layout to activate on printer; you could also select to activate always the 'A' layout.
5. Specify how many variable fields each layout is composed by.
6. For each variable field of each layout, specify the position (into record from serial device) of 'start_field' and 'end_field'.
7. Specify the position (into record from serial device) at which Keypad will find the number of batch copies to do; you could also select to deactivate always the batch printing.
8. Execute up to 5 multiplications per label between data contained all into string record and send the results to printer in command ?25&... form.
9. Execute up to 5 multiplications per label between 2 data: the first contained into string record and the second inputed from keyboard. Results will be sent to printer in command ?25&... form.
10. Input a 'Price' data and multiply it for a 'Weight' contained in the record from serial device. If you use this feature, the first time you enter the MACRO mode you will be asked for input of Price data. This data will be printed on label as last variable field but one, while the result of multiplication will be printed on label as last variable field.

MACRO-Interpreter is programmed by ?59&... command; it has five switches, having the following syntax and functionality:

?59&1, Layout, Fld_Num, Data

Layout	upcase letter that shows which layout the following start/end values refer to: there are up to 26 available layout (A..Z)
Fld_Num	integer that shows how many variable fields there are into LAYOUT (max 19 fields)
Data	list of integer numbers, showing the start/end position for each variable fields into record from serial device (see example below). Range: 0..255

With this switch you can program the 'Start/Stop' position table of variable fields for each layout you have on printer. If you have programmed the layout 'L' on printer with 1 only variable field, you can tell Keypad to extract 1 variable field from string record when the layout 'L' is activated: you do this by programming the start and stop position (into record from serial device) of field to be extracted. (I.E. ?59&1,L,1,Start,Stop<CR LF> Note that the first byte of string record from serial device has number 0)

?59&2, LayPos, ND_Lay, Str_T,	{	FL_Siz (if Str_T = 0)
	}	EndChr (if Str_T = 1)
LayPos		position (into record from serial device) of the first byte of the number corresponding to layout to activate. Range: 0.. 255. (Remind that the first Layout [A] is coded by number 0 into the record, the second [B] by number 1, and so on until the last [Z] corresponding to number 26)
ND_Lay		integer telling how many digits layout number (into record from serial device) is made by. Range: 0..4. If you select 0, Keypad will always activate 'A' layout, regardless of what it will find at LAYPOS
Str_T		record type. Legal values are: 0 for fixed length records and 1 for fixed terminator character records
FL_Siz		integer showing the fixed length record size. Range: 0..255
EndChr		it's the ASCII code of terminator character for record. Range: 1..255

With this switch you tell Keypad two important facts. The first thing is which layout to activate (with ?05&... command) when a record string has arrived; you can choose to activate always the 'A' layout or to look into record string for number of layout. If you choose this second one you have to tell Keypad at which position (into record from serial device) this number starts (Laypos) and how many bytes it is made by (ND_Lay). The second thing is to tell Keypad if the record string will be a fixed length one or a fixed terminator character one (Str_T); if record string has always the same length you have to program its value (FL_Size), otherwise you have to program the ASCII code of terminator character (EndChr).

?59&4, Pos_Bat, ND_Bat

Pos_Bat integer showing the position (into record from serial device) of batch copies number. Range: 0..255
 ND_Bat integer showing how many digits batch number (into record from serial device) is made by. Range: 0..4. Choosing 0 will result in no batch operation.

With this switch you tell Keypad if it must look into record string for batch copies number. If so you must program the start position (into record from serial device) of number (Pos_Bat) and how many bytes it's made by (ND_Bat).

?59&5, Layout, Num_Molt, Data

Layout upcase letter that shows which layout the following start/end values refer to: there are up to 26 available layout (A..Z)
 Num_Molt number of multipl. to do when Layout is activated (max. 5 per layout)
 Data table of start/stop positions (into record from serial device) of operands of multiplications. If there is one only multiplication, Data will be composed by "Start,Stop," (of 1st operand) and "Start,Stop<CR LF>" (of 2nd operand).

With this switch you can associate up to 5 multiplications to each available layout (**in this mode layouts are 'A' ... 'P'**); data of operations are taken from record string. Results of operations are sent to printer as last variable fields of layout in ?25&... command form. If you have a serial scale that sends a record string containing weight and unit price, you can print the total price on label as result of multiplication between these 2 fields. Note that before sending results, Keypad sends to printer the variable fields programmed with ?59&1,... switch. (See 'Example 2' below)

?59&7, Start, Stop

Start Start position (into record from serial device) of Weight data
 Stop Stop position (into record from serial device) of Weight data

This switch is useful if you have a Price that never changes once input. So when you enter MACRO mode the first time you will be asked for input of Price that will be the first operand of multiplication: it will remain unchanged until you exit MACRO mode. The second operand (Weight) will be read from record string from Start byte to Stop byte. In this mode Keypad sends first the variable fields programmed with ?59&1,... switch, then the Price value and at last the result of multiplication; all in ?25&... command form.

Important: Switches 1, 2, 4 of ?59& command are indispensable for correct programming of MACRO, while switches 5, 7 are optionally programmable. You can program one only switch (between 5 and 7) at a time; this means Keypad "remembers" only the last switch seen (between 5 and 7).

Example 1

Hypothesis:

- * Two programmed layouts into printer, A (with 2 variable fields) and B (with 1 only variable field)
- * The serial device has the following two output strings:

1)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	*	0	1	*	W	E	I	G	H	T	*	P	R	I	C	E	.	.	@

2)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	*	0	2	*	O	N	L	Y	_	P	R	I	C	E	@

* @ is the terminative (programmable) character of the output string: suppose it has ASCII code 15 decimal

The code to implement this feature is the following: (/*...*/ = comments)

```
?59&1,A,2,5,10,12,18 /* 'A' layout, 2 variable fields, start/end data */
?59&1,B,1,5,14 /* 'B' layout, 1 variable field, start/end data */
?59&2,0,1,1,15 /* Layout Position = 0, one digit, string with terminator character, */
/* terminator character = ASCII(15) */
?59&4,2,2 /* Batch copies number position in record = 2, */
/* how many digits = 2 */
$! /* File Terminator */
```

Example 2 (multiplication)

Suppose indicator outputs the next string:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
W	e	i	g	h	t	*	P	r	i	c	e	/	K	g	*	N	a	m	e	P	r	o	d

You want to print (as variable fields) the name of product (1st var. field on layout), the weight (2nd var. field on layout), the unit price (3rd var. field on layout) and the result of multiplication between Weight and Price/Kg (4th var. field on layout).

So you have to program on printer a layout (suppose 'A') with 4 variable fields, the last of which must be that will contain the result of multiplication.

Once layout 'A' is programmed on printer, you have to program MACRO on Keypad; the commands to do it are the next: (/*...*/ = comments)

```
?59&1,A,3,16,23,0,5,7,14 /* 3 var. field read from string record */
?59&2,0,0,0,24 /* Always layout 'A', string record lenght=24 byte */
?59&4,0,0 /* No batch copies */
?59&5,A,1,0,5,7,14 /* 1 var. field: multiplication associated with layout 'A', */
/* start/stop position of operands */
$! /* File Terminator */
```

When string record goes from serial indicator to Keypad, it reads three variable fields and transmit them to printer in command ?25&... form, then it reads the multiplication operands, executes the calculus and transmits the result to printer in command ?25&... form. So all four variable fields in layout 'A' are filled with data and a label will be printed.

WARNING: Parameters and tables of MACRO Interpreter programming are stored in E²PROM: the MACRO occupied area is shared with the programming of printer layouts above discussed. Note that, when programmed, MACRO Interpreter reduces heavily the part of E²PROM normally available for layout programming. So, when programmed, MACRO Interpreter may cause lost of data for E²PROM stored layouts (i.e: if you have already programmed E²PROM with some layouts and then you program MACRO, you

have to re-program E²PROM layouts, because the layout part of internal memory is overwritten by MACRO; this new programming will be written in a layout dedicated part of E²PROM).

The internal memory becomes fully available again after a general reset.

WARNING: The file containing the MACRO commands (?59&...) MUST be closed by the string “\$! <CR LF>”

The PLU mode

Under **F6** function key you have some features about PLU mode. This working mode solves the problem linked to the food industry world labelling, with or without serial connection to a scale for weighing, calculation of total amount as multiplication between weight and unit price, management of “best before” date and “seasoning” date (for each product), partial and general total management, creating of composite barcodes. All this features are possible by programming the products table called sintetically PLU (Product Look Up); let’s take a look.

What’s PLU ?

PLU is a list of record (stored in RamCard/Flash); each one contains informations related to a product to label. Printing data are read from a goods database stored in RamCard/Flash and combined with the one read from a serial device (i.e. a scale) connected to 9 pins serial port of Keypad.

We have to remark the correspondance one to one between PLU records and products to label; each record corresponds to one only product.

One PLU record describes all peculiarities necessary to print a product label.

Each Plu record is composed by three parts (the middle one is optional) and has the next structure:

First_Section TAB Middle_Section TAB Last_Section ENTER

TAB = PC Tabulator character = ASCII(9)

Now we’re gonna analysing each section deeply.

First Section

This part of PLU record allows to set up:

1) LABEL LAYOUT

An uppercase letter showing which layout the product will activate on printer

2) PRODUCT UNIT PRICE

It’s useful both for “weight” products (they need a weight from a scale) and “body” products (that is products with fixed weight like a can of drink, a bottle of milk, and so on). It’s always composed by 6 characters: if it is shorter than six characters, the remainings will be asterisks (‘*’); for instance if you want to program a price of £20.7 you have to program “20.70*”, 6 characters at all. If price has decimals, it must have 2 decimal digits.

3) “BEST BEFORE” DATE

It’s expressed in days, it’s calculated starting from the date of production stored into Keypad memory and it’s related to the selected product (each product can have an own best before date). It’s always composed by 4 digits; for instance 120 days will be programmed “0120”. If product has no best before date, you have to program this field with four “zeros” (“0000”). If the first character is a minus (‘-’) then the expressed days will be subtracted from packing date.

4) SEASONING DATE

It’s expressed in days, it’s calculated starting from the date of production stored into Keypad memory and it’s related to the selected product (each product can have an own seasoning date). It’s always

composed by 4 digits; for instance 65 days will be programmed “0065”. If product has no seasoning date, you have to program this field with four “zeros” (“0000”). If the first character is a minus (‘-’) then the expressed days will be subtracted from packing date.

5) FIXED PART OF BARCODE

This field is always composed by 7 digits, that will take part to the creation of left size of barcode (if its printing is enabled); if fixed part of barcode is made only by 6 digits, only first six digits of field will be read.

6) PRODUCT NAME

This field is always composed by 4 digits; the field shows the number of ingredients database record from which name will be read. The number identifies the record into database: if record is composed by more than one field, only the first one will be read. For instance: text contained into the first field of record n° 65 of database must be programmed as “0065”.

7) DATA/FIXED_CODE DIGITS INTO BARCODE

This field is always composed by 2 binary digits (‘0’ or ‘1’) showing how many data digits or fixed code digits will be printed into barcode. Fixed code part is on the left half of barcode, while data part occupies the right one. To understand better the matter take a look of exemples of barcodes just below. The next table summarizes the four possible combinations for this field:

	DATA digits	fixed CODE digits
0 0	5	6
0 1	5	7
1 0	6	6
1 1	6	7

8) FLAG

This field is always composed by 4 binary digits (‘0’ or ‘1’) showing (from left to right):

1. Enable of barcode print on the label (0 = OFF; 1 = ON)
2. Selection of Amount / Weight into barcode data part (0 = Amount; 1 = Weight)
3. Enable of production date print on label (0 = OFF; 1 = ON)
4. Selection between “Weight” and “Body” product (0 = Body ; 1 = Weight).
5. All data come through serial port from a serial device: Keypad executes no multiplications
6. Unused
7. Unused
8. Unused

Each field is separated from the successive by a semicolon (‘;’)

EXAMPLES OF BARCODES

- 1) fixed part = 6 digits; data part = 5 digits

c = fixed code

d = variable data part

k = automatic control characters



- 2) fixed part = 7 digits; data part = 5 digits
 c = fixed code
 d = variable data part
 k = automatic control character



The seventh fixed code digit replaces the middle control digit

- 3) fixed part = 6 digits; data part = 6 digits
 c = fixed code
 d = variable data part
 k = automatic control characters



Most significant data part digit replaces the last fixed code digit and middle control digits remains

- 4) fixed part = 7 digits; data part = 6 digits
 c = fixed code
 d = variable data part
 k = automatic control character



Most significant data part digit replaces the middle control digits

- 5) If data part contains a number of digits greater than the programmed one, barcode will be composed by twelve zeros ('0') to signal the error



Example of First Section programming

B;15.45*;0365;0100;1234567;0041;00;10111000 TAB

It has next meaning:

- Printer layout: B
- Unit price: 15.45
- Best before date: 365 days starting from production date
- Seasoning date: 100 days starting from production date
- Fixed barcode part: 1234567
- Product name: record n° 41 from ingredients database
- Data barcode part: 5 characters
- Fixed barcode part: 6 characters
- Print of barcode: enabled
- Print of amount into data barcode part
- Production date print: enabled
- "Weight" product
- Keypad executes no multiplications because all data come from serial device

Middle Section (optional)

This section is useful to print on the label list of texts (for instance ingredients) read from goods database. It's possible to program the print direction, co-ordinates, font to use, font magnifying and naturally from what database records data are read. If you program more than one consecutive field, data will be printed separated each other by a comma (','). Syntax of Middle Section is:

D;X;Y;F;EE;Ingr_1, Ingr_2,...., Ingr_n TAB

where parameters have the next meaning

D	=	Print Direction (Values: 0, 1, 2, 3)
X	=	Co-ordinate x
Y	=	Co-ordinate y
F	=	Font type (see printer programming manual)
EE	=	2 digits for font magnifying: 1st for width, 2nd for height
Ingr_i	=	Ingredients database record index from which data will be read
TAB	=	PC TABULATOR character

Middle section can be made by more than one consecutive block like the one above described; each block defines a different ingredient string.

Example of Middle Section programming

2;150;80;5;11;12,128,53,7

2	=	Printing direction
150	=	X
80	=	Y
5	=	Font n° 5
11	=	Font magnifying: 1 for width and 1 for height
12,128,53,7	=	Database records index from which data are read

Last Section

This section of PLU record allows to read and decode input data string from an external serial device. In particular in food industry applications it allows to read Weight field from a scale.

It is composed by couples of values separated by a comma (','); each couple contains the Start and Stop positions, inside input scale string, of data field you want to read.

Last section is really used only by "Weight" products, but it must be always present, even if only as a fictitious field, (i.e. 0,0), in "Body" products". More than one only couple can be programmed, if data fields to read from input record are many. **First couple of values always refers to weight. Besides, if all data are coming from serial device, the last couple of values refers to total amount.** Section MUST be terminated by a Carriage Return (ENTER key of PC).

Example of Last Section programming

2,8,10,16 ENTER

Meaning: Weight is read from scale input record starting from byte 2 to byte 8; other variable data field (i.e. tare) is read from scale input record starting from byte 10 to byte 16. (Notice that the 1st byte into scale record has number 0)

Let's take a look about other functions tied to use of PLU record table.

PLU mode: general functions

F6 key holds all PLU functions, that are explained below:

- **Printing product selection made by keyboard**

It allows to select using product (PLU) by Keypad: PLU number is written by Keypad keyboard.

- **Printing product selection made by serial input record**

Selection of product (PLU) of which you want a print is made by serial input record coming from a scale.

- **Production date programming**

It allows to store a date (in character string form) into Keypad. Starting from production date both best before and seasoning dates will be calculated. Packing date will remain stored into Keypad also at power off, until you program it again

- **Totals printing**

This function allows to print totals. It's possible to select between **partial** or **general** total. For more details see chapter "Totals".

- **Clearing totals**

It allows to clear both partial and general totals.

To select above functions press F6 key on Keypad and then scroll up or down until desired function; confirm with "Enter" key.

Totals

See next two labels:

PARTIAL TOTAL	
Product Num.	518
Tot. Label N.	2
Tot. Weight Kg.	1,9
Tot. Amount Lit.	68570

GENERAL TOTAL	
Product Num.	*
Tot. Label N.	15
Tot. Weight Kg.	7,8
Tot. Amount Lit.	92540

As you can see, totals hold the next common informations: number of printed label, total weight, total amount. Besides them, partial total shows number of active product (PLU) too.

Totals work in this way: when you select one product from PLU table, if it is different from the last you selected the last time, general total is added with partial total and then this last one is cleared. The only way to clear general total, instead, is to use the function held by F6 key.

How can I program a "total" label on printer?

Print of total works with 'E' layout on printer: this layout must be already programmed when you ask for total print.

To program 'E' layout is enough to remind that it must contain 5 variable fields, that have next meaning:

1. Writing of "PARTIAL" or "GENERAL"
2. Number of selected PLU ("*" in General Total case)
3. Total labels printed
4. Total weight
5. Total amount

All other is left to the user fantasy.

How can I program a "product" label on printer?

Labels prepared for PLU use must have some characteristics about order of their variable fields.

First variable field is defined "Product name", and it will always contain the name of selected product (i.e. "Smoked salmon", "New Zeland KIWI", ecc...) read from goods database.

Now we can distinguish between "weight" or "body" product.

Let's take a look about "weight" products

The second field will contain net weight read from the scale.

After weight, possible other variable fields read from data input record will sent to printer; their number depends only from programming of the last section of PLU (they are optional).

After scale variable fields, the next one will contain unit price of product and it will be read from PLU record.

Next variable field on label will contain the production date, if its printing is enabled into PLU.

Next two fields will contain seasoning date and best before date respectively, if their printing is enabled into PLU record.

Next variable field will contain total amount; this field, together with product name, net weight and unit price, must always be present on label. Total amount is calculated as multiplication between “Net Weight” (from scale) and “Unit Price” (from PLU).

Last field on label must be reserved to barcode, if its printing is enabled into PLU record. Barcode type must be “EAN13 with Check Digit”.

Summarizing, for “weight” products we have:

1. Product name (necessary)
2. Net weight from scale (necessary)
3. Other fields from scale (optional)
4. Unit price of product (necessary)
5. Production date (optional)
6. Seasoning and best before dates (optional)
7. Total amount (necessary)
8. Barcode (optional)

So, a “weight” product label must contain at least 4 variable fields.

NOTE: If chosen product has all data coming from the scale (see First Section programming) so the variable fields programming order changes and becomes:

1. Product name (necessary)
2. Net weight from scale (necessary)
3. Unit price of product (optional)
4. Other fields from scale (optional)
5. Total amount (necessary)
6. Production date (optional)
7. Seasoning and best before dates (optional)
8. Barcode (optional)

Let’s take a look about “body” products

The second variable field on label will contain the production date, if its printing is enabled into PLU record.

Next two fields will contain seasoning date and best before date respectively, if their printing is enabled into PLU record.

Next variable field will contain total amount; this field, together with product name, must always be present on label. Total amount is read from field Price into PLU record.

Last field on label must be reserved to barcode, if its printing is enabled into PLU record. Barcode type must be “EAN13 with Check Digit”.

So, for “body” products we have:

1. Product name (necessary)
2. Production date (optional)
3. Seasoning and best before dates (optional)
4. Total amount (necessary)
5. Barcode (optional)

So, a “body” product label must contain at least 2 variable fields.

Programming steps of PLU

To use PLU you have to program the next:

- 1) Ingredients database
- 2) PLU record table
- 3) MACRO to read and decode data from a scale (optional)

Ingredients Database

It's useful to contain all basic ingredients and names of products. It can store up to 65536 records; to program it you must create an ASCII file containing data and having the next structure:

```
Anchovies in olive oil <ENTER> /* Ingredient n° 0 */
Anchovies in salt < ENTER > /* Ingredient n° 1 */
Apples < ENTER >
....
Vinegar < ENTER >
Water < ENTER >
Wax < ENTER >
$! < ENTER > /* It shuts the file */
```

Once you created correctly the file, you have to transfer it to Keypad RamCard/Flash following the instruction into "Programming steps" chapter. Before starting transmission from PC, you must put Keypad in receiving mode pressing F1 key and selecting DATABASE voice.

Products Look Up (PLU)

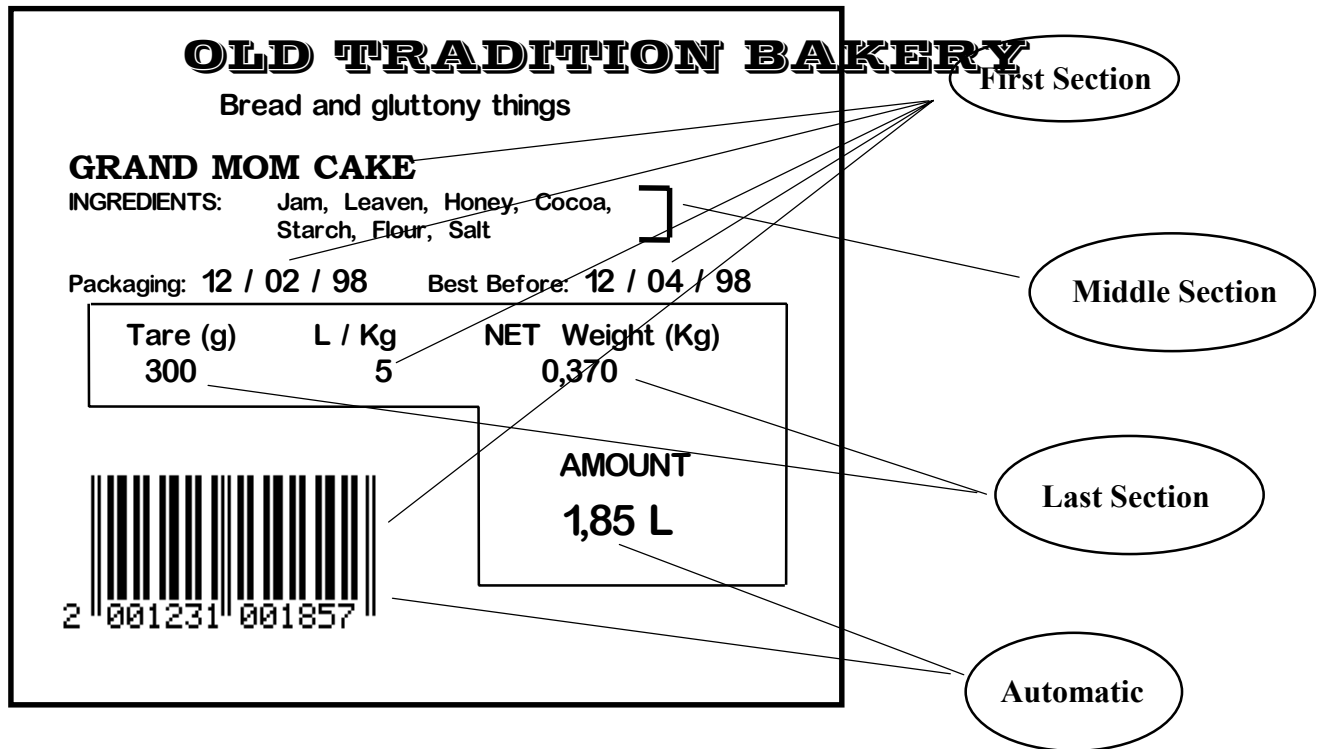
It is stored into RamCard/Flash and it can contain up to 3000 records; each record has the above mentioned structure, that here we're going to summarize:

```
B;2670**;0090;0000;8003935;0258;11;00100000 | 2;20;15;5;11;20,56,49 | 0,0 <ENTER>
          1st SECTION | 2nd SECTION | 3rd SECTION
                          TAB          TAB
```

It's useful to remind that the 2nd Section can also be absent, or it can be composed by more than one block with structure alike the one in the example.

To be programmed, PLU table must be saved as ASCII file; when file is complete, it must be sent to RamCard/Flash, following the instructions into "Programming steps" chapter. Before starting transmission, Keypad must be in receive mode; you do that pressing F1 function key and scrolling to "PLU" voice.

Next label summarizes various features of PLU mode:

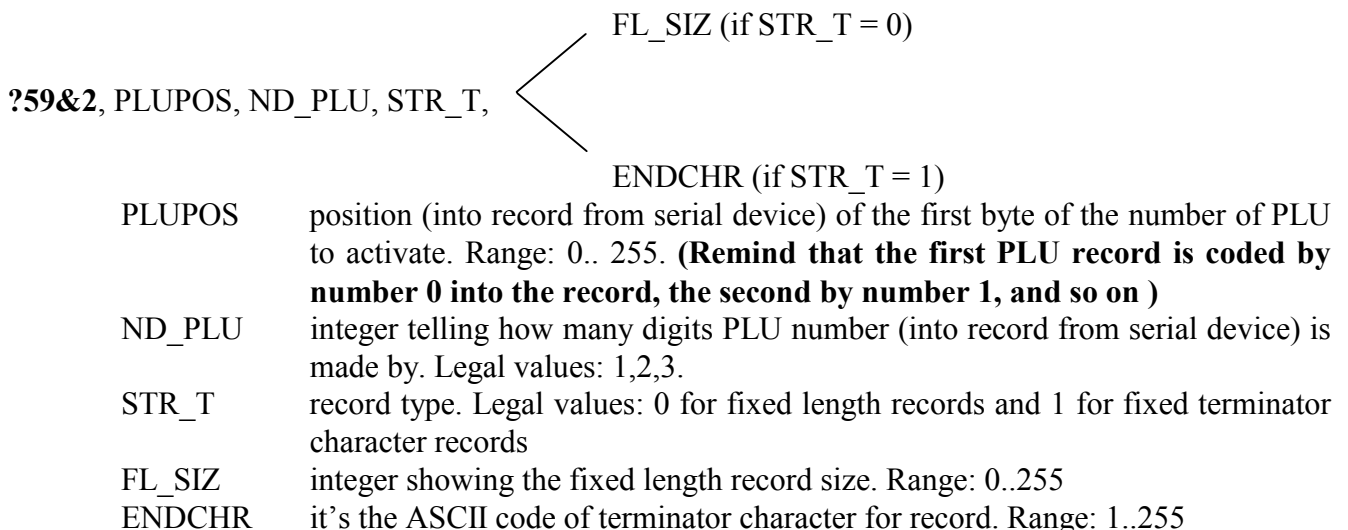


MACRO to read data from scale

If PLU mode includes data from a scale, (or from any other serial device), it's necessary to prepare Keypad with ?59&2,... command (MACRO) that has two main purposes:

1. establish the input record dimension or its terminator character
2. establish in which input record position Keypad will read PLU number

The syntax of ?59&2,... is the following:



To program ?59&2,... command it's necessary to make an ASCII text file following the below example and then to send it to Keypad following the instructions into "Programming Steps" chapter. Before starting transmission, Keypad must be in receive mode; you do that pressing F1 function key and scrolling to "MACRO" voice.

Example of ?59&2,... command

?59&2,10,3,0,38<ENTER>

\$!<ENTER>

In this example we tell Keypad that PLU number starts at byte n° 10 into input scale record (remind that the first byte has n° 0) and it is done by 3 digits: we tell also that record is a fixed length one ('0') and it will be 38 characters long.

WARNING: Like MACRO, PLU programming reduces the E²PROM space normally available for printer format. **If you have yet programmed some format in E²PROM and you program PLU, it is necessary to re-program printer formats on Keypad**

Global recapitulatory example for PLU using

Next example includes an Ingredient Database, a PLU record table, the MACRO command and records to simulate scale weight transmission. Descriptions for printer layouts programming are enclosed too.

1) Ingredients Database:

ANCHOVIES	/* Ingredient n° 0 */
BARBERA Wine 720 cl.	/* Ingredient n° 1 */
APRICOTS	
TUNNY-FISH 200 Gr.	
PEANUTS	/* ... */
SUNFLOWER OIL 1 Lt.	
DUCK	
MIN. WATER 12 Bottles.	/* Ingredient n° 7 */
WATER-MELON	/* Ingredient n° 8 */

\$!

2) PLU record table (notice that Middle Section was not programmed):

A;1*****;0090;0000;1234567;0008;11;10110000	3,8	/* Product n° 0 = Water-Melon */
B;2*****;0365;0000;1234567;0007;11;10100000	3,8	
A;3*****;0060;0000;1234567;0006;11;10110000	3,8	
B;4*****;0180;0000;1234567;0005;11;10100000	3,8	
A;5*****;0030;0000;1234567;0004;11;10110000	3,8	/* ... */
B;6*****;0365;0000;1234567;0003;11;10100000	3,8	
A;7*****;0060;0000;1234567;0002;11;10110000	3,8	
B;8*****;0365;0000;1234567;0001;11;10100000	3,8	
A;9*****;0090;0000;1234567;0000;11;10110000	3,8	/* Product n° 8 = Anchovies */

\$!

| |

TAB ENTER

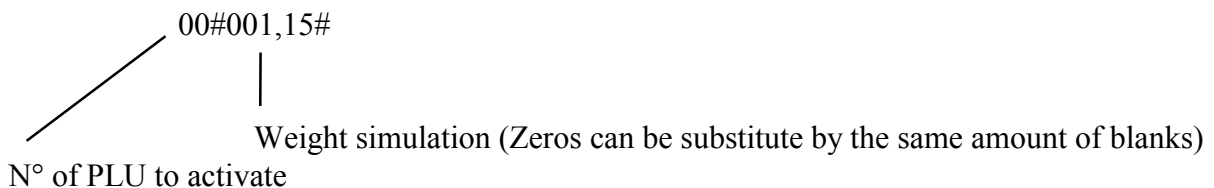
3) MACRO to decode scale record (in our simulation PC takes place of scale)

?59&2,0,2,0,10

\$!

4) RECORD '1' to simulate the scale:

You must save next record on PC as a 10 characters long text file.



5) RECORD '2' to simulate the scale:

You must save next record on PC as a 10 characters long text file.

02#007,39#

WARNING: It's indispensable to set the "Decimal digit" parameter (with Menu key of Keypad) accordingly with number of decimal digits present into scale weight record; possibilities are 0,1,2 and 3. For example, if scale works with a 2 decimal digits weight, value for Keypad parameter must be 2 (as in our demonstration).

6) Rules for printer layouts programming

Used layouts will be 'A', 'B', 'E', programmed as follows:

'A' layout: it must contain 7 variable fields, for next data (from the 1st to the 7th): Product name, Net weight, Unit price, Packaging date, Best before date, Total Amount, Barcode.

'B' layout: it must contain 5 variable fields, for next data (from the 1st to the 5th): Product name, Packaging date, Best before date, Total Amount, Barcode.

'E' layout (used by total label): it must contain 5 variable fields, for next data (from the 1st to the 5th): Total type, selected PLU, Number of printed label, Weight, Total amount.

Once Keypad and printer are programmed, we can start with simulation, after having correctly connected PC to Keypad and Keypad to printer.

At first we want to clear all totals: strike F6, scroll and choose "Clear all Totals" and hit Enter.

Now we program the packaging date: strike F6, scroll and choose "Date of product." then confirm with Enter; now you can digit the date as 8 characters long string ("DD-MM-YY") and confirm with Enter.

Let's start choosing a PLU record activated by Keypad. Press F6 key, select "Manual PLU" and confirm with Enter; now you must insert the PLU number you want to use accordingly with product you want to print: digit '1' and confirm with Enter. A label referring to product n° 1 (which is a "body" product) will be printed. Now digit '4' for batch print and confirm with Enter; four labels just alike the first one will be printed. Now press Esc to return to main menu.

Now we want to print the Partial total referring to the last selected product. Press F6, scroll to "Print totals" and confirm with Enter; now select "Partial total" and confirm with Enter. A total label referring to product n° 1 will be printed.

Now we want to select a "weight" product. Press F6, select "PLU with scale" and confirm with Enter. Keypad begins to wait for scale record (PC record in our case). At DOS prompt on PC, set up serial port parameters (suppose to use the serial port n° 1) with command "MODE COM1 96,n,8,1,P" followed by ENTER.

Now we can send file, containing the simulation of scale record, from PC to Keypad (suppose files you created are named "RECORD1.TXT" and "RECORD2.TXT"); run DOS command "COPY RECORD1.TXT COM1". Printer prints a label of product n° 0 (which recalls the main text n° 8 = Water-Melon). Selection of product n° 0 was made by the input record (field "00" at the beginning of record).

Now we will recall the partial total print as we did before. New label will be referring to PLU n° 0.

We want now to simulate the weighing of another product by copying the second record from PC to Keypad; run DOS command "COPY RECORD2.TXT COM1". Printer prints a label of product n° 2 (which recalls the main text n° 6 = Duck). Selection of product n° 2 was made by the input record (field "02" at the beginning of record).

Now press Esc and call the partial total print, as we did before. New label will be referring to PLU n° 2.

To terminate the simulation we will call a general total print (which is the sum of all done weighings). Press F6, scroll to "Print Totals" and confirm with Enter; now scroll to "General Total" and press Enter. The general total label will be printed.

General reset

When Keypad is displaying the message “Hit a function key” you have the possibility to reset the terminal. This operation is made by pressing at the same time the up and down scroll arrows.

After a general reset a message is shown on display.

Reset involves the following object:

- Language: new language will be ENGLISH
- Serial line: new parameters will be 9600, N, 8, 1
- Internal memory: new setup is FULLY AVAILABLE (just under 2 Kbytes). The first 5 ayou (‘A’...’E’) are initialized to “NONAME”
- Startup mode: new setup is NORMAL MODE
- MACRO: new setup is NOT PROGRAMMED
- PLU new setup is NOT PROGRAMMED
- RamCard/Flash manage new setup is “ALL SEQUENCES”
- Decimal digit new setup is 0 for both “weight” and “total amount”
- Batch counter new setup is OFF

MENU key

With this feature you can choose to setup the next:

- Startup mode (normal or PIPE mode)
- Language of messages (English, Italian or French)
- Serial line to PC (N,8,1 or E,7,1 or O,7,1. Baud: 1200, 2400, 4800, 9600, 19200)
- Number of decimal digits contained in the weight coming from a scale (0, 1, 2, or 3)
- Number of decimal digits contained in the printed total amount during PLU mode (0, 1, 2, or 3)
- RamCard/Flash manage. You can select: all for sequences; both for sequences and PLU+Database; all for PLU+Database. Your choice is stored in E²PROM and is checked when you program something into RamCard/Flash: i.e. if “all sequences” is selected and you want to program “PLU”, an error message is displayed.
- Displaying of counter during a batch printing; if flag is setted ON, during the batch print you can see on display the total of labels to print and the partial that will increase after each printed label. When batch print finished, press ESC to exit mode.

INDEX OF CONTENTS

GENERALS FEATURES	2
TECHNICAL SPECIFICATIONS.....	2
KEYPAD POWER SUPPLY	2
FEATURES.....	3
FUNCTION KEYS.....	4
ALPHANUMERIC KEYS.....	4
PROGRAMMING STEPS.....	5
SERIAL CABLE.....	5
INTERNAL MEMORY (E²PROM) PROGRAMMING	5
INTERNAL MEMORY COMMANDS	6
<i>Layout describer</i>	<i>6</i>
<i>Variable data Input.....</i>	<i>6</i>
<i>Activating of a stored into printer Logo.....</i>	<i>6</i>
<i>Set up of multiple data output.....</i>	<i>7</i>
<i>Fixed texts reading from printer permanent memory</i>	<i>7</i>
<i>Partial data input.....</i>	<i>8</i>
<i>Input of a numeric data valid as number of batch copies too.....</i>	<i>8</i>
RAMCARD/FLASH LABELS PROGRAMMING.....	9
RAMCARD/FLASH SPECIAL COMMANDS.....	9
<i>Variable data Input.....</i>	<i>9</i>
<i>Activating of a Logo previously stored into printer.....</i>	<i>9</i>
<i>Call another sequence stored in RamCard/Flash.....</i>	<i>10</i>
<i>Enter automatically into MACRO mode.....</i>	<i>10</i>
<i>Read variable fields from a DataBase.....</i>	<i>10</i>
1: Manual input of record index.....	10
2: Scrolling records of DataBase.....	10
<i>Fixed texts reading from printer permanent memory</i>	<i>11</i>
<i>Input of a variable data with a default value (only for RAMCARD version).....</i>	<i>11</i>
<i>Input of a numeric data valid as number of batch copies too.....</i>	<i>12</i>
HOW TO SEND RAMCARD/FLASH CONTENTS TO A PC.....	13
HOW TO DUPLICATE A RAMCARD SEQUENCE CHANGING ITS DATA.....	13
HOW TO PRINT A LAYOUT	13
ACTIVATE AND PRINT A LAYOUT ALREADY STORED IN PRINTER MEMORY	13
PRINT A LAYOUT READ FROM RAMCARD/FLASH	14
EXAMPLES	15
THE “PIPE” MODE	16
THE “MACRO” MODE.....	16
GENERAL.....	16
MACRO INTERPRETER PROGRAMMING	16
EXAMPLE 1	18
EXAMPLE 2 (MULTIPLICATION).....	19
THE PLU MODE	20

WHAT'S PLU ?.....	20
PLU MODE: GENERAL FUNCTIONS.....	23
TOTALS	24
HOW CAN I PROGRAM A "TOTAL" LABEL ON PRINTER?	24
HOW CAN I PROGRAM A "PRODUCT" LABEL ON PRINTER?	24
PROGRAMMING STEPS OF PLU	26
GLOBAL RECAPITULATORY EXAMPLE FOR PLU USING	28
GENERAL RESET.....	30
MENU KEY	30
INDEX OF CONTENTS	31