

OPERATION MANUAL FOR DISPLAYS

DT-101G, DT-103 and DT-104G

Index

1.-INTRODUCTION	1
2.-GENERAL FEATURES	3
2.1 General features DT-101G.....	3
2.2 General features DT-103G and DT-104G.....	4
2.3 Sizes.....	5
3. EXTERNAL CONNECTION	7
3.1 DT-101G and DT-104G Inputs connection.....	7
3.2 DT-101G and DT-104G Serial line connection.....	9
3.3 DT-103G external connection.....	11
4.-PLC PROGRAMMING	13
4.1 Messages without variables.....	14
4.2 Messages within variables.....	15
4.2.1 Phase 1 - Enable message within variable.....	16
4.2.2 Phase 2 - Transference and update a variable's value.....	17
4.2.3 Phase 3 - End of message within variable.....	19
4.2.4 Errors transmitting variables.....	19
4.2.5 Variable's format.....	20
4.2.6 Points and comas programming.....	21
4.3 Print messages.....	24
5.-PROBLEMS. What to do if	25

1. Introduction.

DT messages of display family is composed by models **DT-101G, DT-103G and DT-104G**. They are the ideal complement for any installation controlled by a PLC (Programmable Logic Control). They can be used with any PLC.

DT-100G family runs by a microprocessor. They constitute an element of control technologically advanced, that has a great flexibility and a great advantage, which is its simplicity.

You will observe that the display is easy to program and easy to handle, allowing you to make effective any kind of application in a short lap of time.

You will verify that the handling of the messages, independently of the PLC that you use, is extremely simple and can be done by any programmer, even with little experience. The handling of variables will not that easy, and will need a little more expert programmer.

This manual describes the technical features and the set of instructions necessary of its use. We recommend carefully its reading before using the display.

WARNING: The communication of the displays is only monodirectional, from the PLC to the display. So in fact, the communication itself doesn't guarantee that the message sent to the display have been received and shown on the on the screen.

2. General Features.

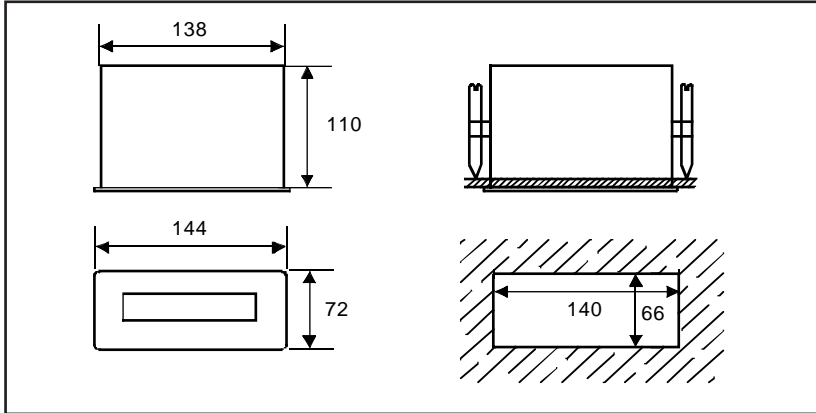
2.1 DT-101G General Features.

Power supply	230VAC $\pm 10\%$ 50/60Hz 5,5VA approx.
Parallel inputs	Types: NPN, PNP, Contact.
.....	Voltage: (level 1) 18 to 26VDC.
.....	Voltage: (level 0) 0 to 2.5VDC.
.....	Maximum ripple: 500mV.
.....	Input resistance: 2Kohm.
Display	16 alfanumerical characters, 6mm height
.....	14 segment character. Fluorescent type.
.....	3 meters visibility.
Mesages and Memory	254 Messages, max.eith 60 characters per messages.
.....	7423 total characters (all messages)
.....	Memory type: EEPROM.
Assembly & Size	144 (with) x 72 (height) x 110 (bottom)
.....	Assembly in panel: 136 (with) x 66 (height)
Temperature range	Working temperature:between 0 and 50°C.
.....	Storage temperature: between: -10°C to 60°C.
.....	Humidity: 5-95% without condensation
Serial line input RS-232C	Transmission speed: 1200 bauds.
.....	Parity: NO
.....	Number of bits: 8
.....	Stop bits: 2
.....	Protocol: Ready / Busy
.....	CR code: CR + LF
.....	Set of characters: Standard USA

2.1 DT-103G & DT-104G General Features.

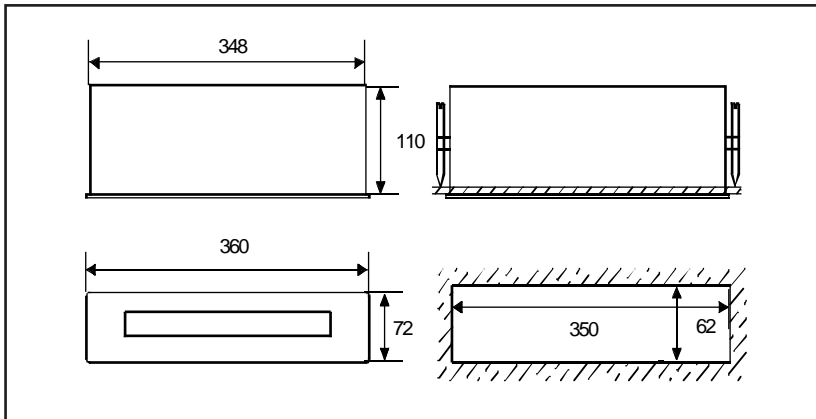
Power supply	230VAC \pm 10% 50/60Hz 11VA approx.
Parallel inputs	Types: NPN, PNP, Contact.
.....	Voltage: (level 1) 18 to 26VDC.
.....	Voltage: (level 0) 0 to 2.5VDC.
.....	Maximum ripple: 500mV.
.....	Input resistance: 2Kohm.
Display	20 alfanumerical characters, 15mm height
.....	5 x 7 matrix of points. Fluorescent type.
.....	Capital I letters, small letters and ASCII.
.....	10 meters visibility.
Mesages and Memory	254 Messages, max.eith 60 characters per
.....	messages.
.....	7423 total characters (all messages)
.....	Memory type: EEPROM.
Assembly & Size	DT-104G: 360 (with) x 72 (height) x 110 (bottom)
.....	Assembly in panel: 350 (with) x 62 (height)
.....	DT-103G: 357(with) x 82 (height) x 110 (bottom)
.....	External assembly: with or whitout foot (opcio-
.....	nal)
Temperature range	Working temperature:between 0 and 50°C.
.....	Storage temperature: between: -10°C to 60°C.
.....	Humidity: 5-95% without condensation
Serial line input RS-232C	Transmission speed: 1200 bauds.
.....	Parity: NO
.....	Number of bits: 8
.....	Stop bits: 2
.....	Protocol: Ready / Busy
.....	CR code: CR + LF
.....	Set of characters: Standard USA

2.3 Size.



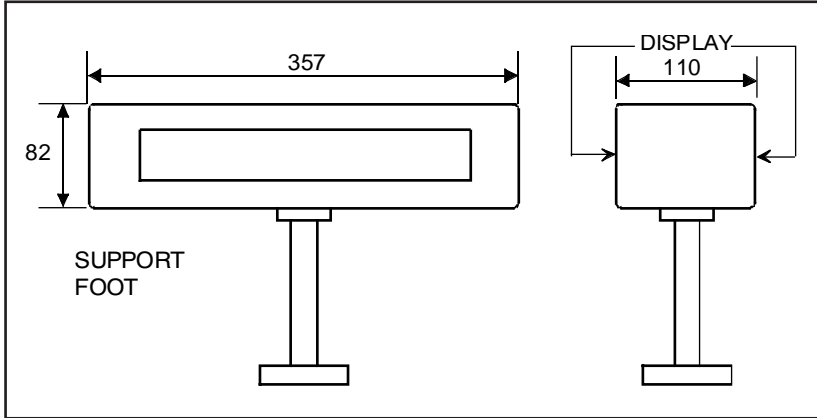
2.3.1 DT-101G size.

The DT-101G display has a plastic box. The front panel has an IP-65 protection.



2.3.2 DT-104G size

The DT-104G display has a metallic box. The front panel has IP-65 protection.



2.3.3 DT-103G size.

The DT-103G has a methacrylate box with a metallic base. The foot is an option

3.- External connection.

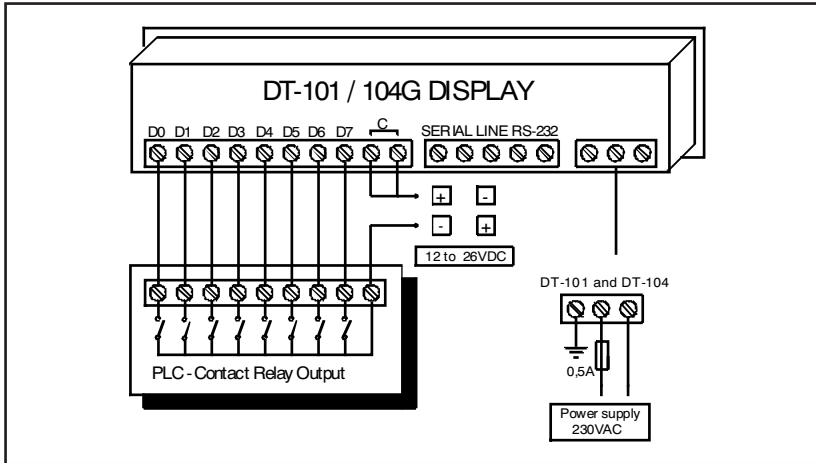
The messages displays family (DT-101G, DT-103G and DT-104G) admit any kind of inputs: NPN, PNP and CONTACT.

The inputs voltage must be from 18V to 26VDC, with a maximum ripple of 500mV. Surpassing there values could cause unrecoverable damages to the display.

We suggest to use the same power supply to feed the display inputs as for the PLC, that probably will have those requirements.

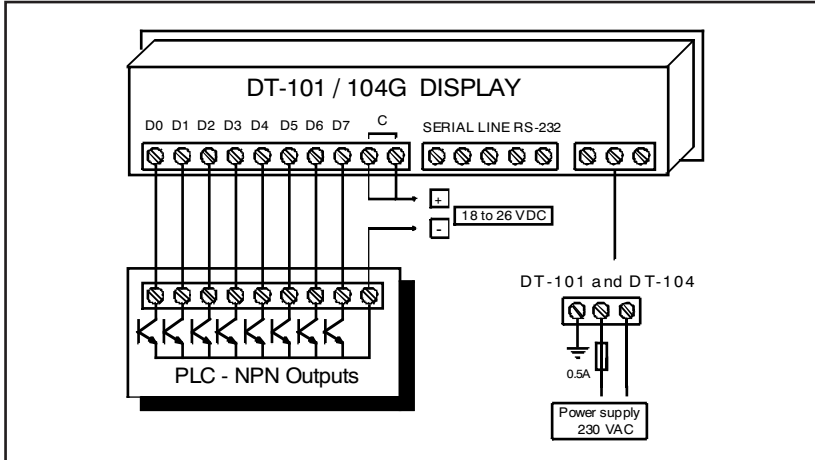
3.1 DT-101G and DT-104G Inputs connection.

When the PLC outputs is a contact one, the polarity of the connection is indifferent, the display admits any polarity. That means, the common of the display can be positive or negative.



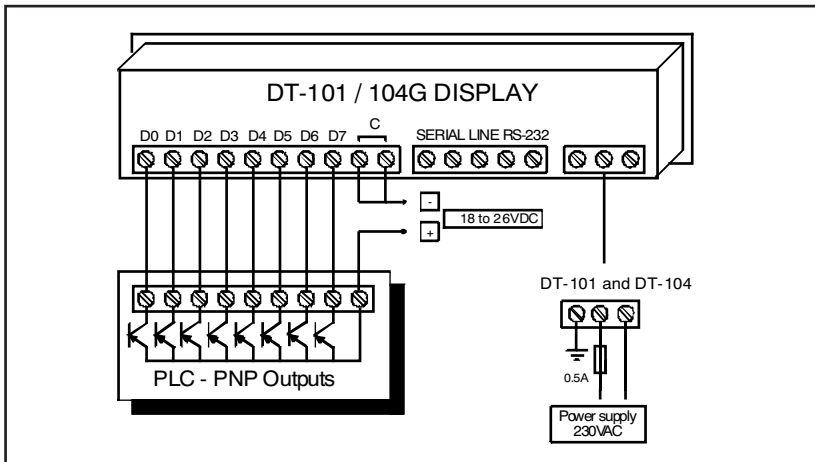
3.1.1 DT-101G/DT-104G connections to PLC. Contact relay outputs.

When the PLC output is a NPN one, you should connect the positive power supply to the common of the display, and the negative as the common of the outputs.



3.1.2 DT-101G/DT-104G connections to PLC. NPN outputs.

When the PLC output is a PNP one, you should connect the negative power supply to the common of the display, and the positive as the common of the outputs.

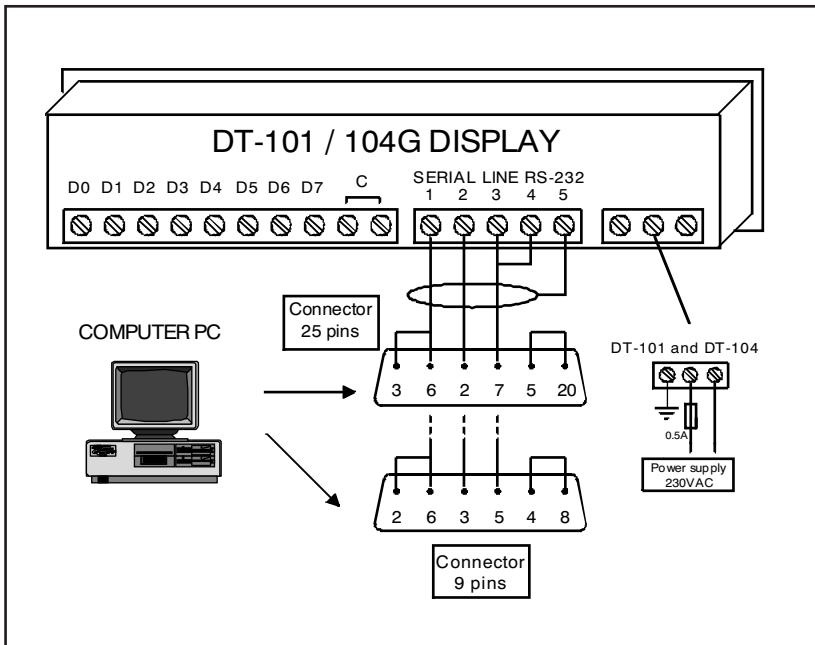


3.1.3 DT-101G/DT-104G connections to PLC. PNP outputs.

3.2 DT-101G and DT-104G Serial line connection.

The serial lines does a double function in the displays.

1) Communication with a PC compatible computer, in order to do the programming of texts and variables in the display from the DT program. The computer must be a truly PC compatible, models XT, AT, 386 or 486 with monochrome or colour monitor.



3.2.1 DT-101G/DT-104G Serial line connection to a PC.

It exists a set of cables with a connector assembled for PC and printer that have the following references:

SC-101A: DT-101/104G interconnection cable to a PC, 25 pins output.

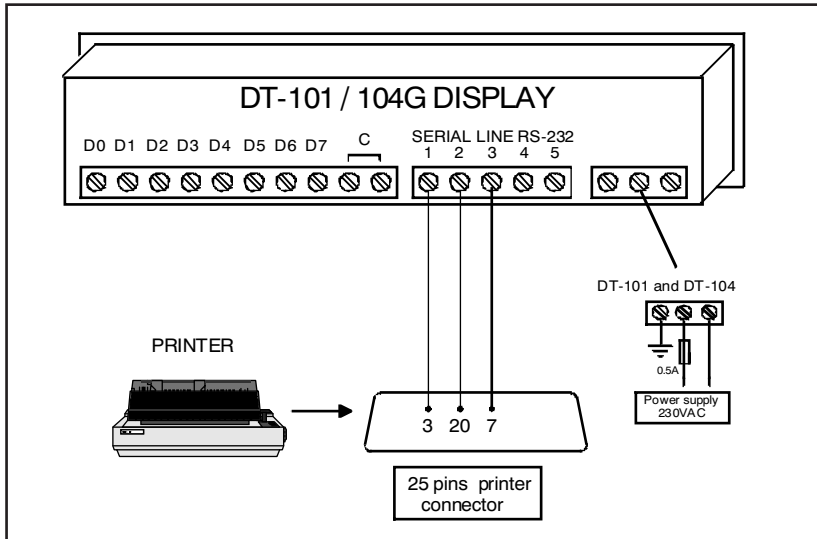
SC-101B: DT-101/104G interconnection cable to a PC, 9 pins output.

SC-101P: DT-101/104G interconnection cable to printer, 25 pins output.

2) Printer output to print the messages that have been enabled in the buffer. See the chapter "Printing messages".

Don't confuse the saving of the messages from the display buffer with the saving of the texts that have been generated with the "DT" program; in this case, the printer will be connected to the computer, and its configuration will depend on the computer, in which the connection could be serial or parallel.

You should configure your printer according to the following attributes:



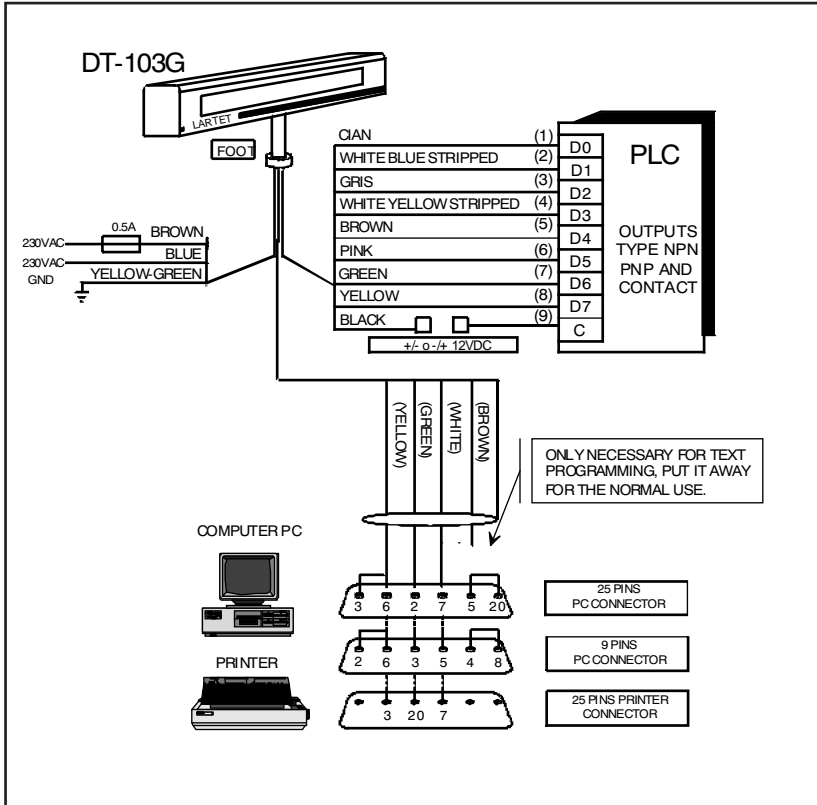
3.2.2 DT-101G/104G serial line connection printer.

Transmission speed..... 1200 bauds.
 N° de bits..... 8
 Parity..... No
 Stop bits..... 2
 Protocol Ready / Busy
 CR Code.. CR + LF
 Characters set:..... Standard USA

3.3 DT-103G external connection.

The DT-103G display model is supplied in two versions:

- a) DT-103GS, one display assembled on one side.
- b) DT-103GD, two displays, assembled on two opposite sides.



3.3.1 DT-103G External connection.

The connection is done by means of a multi-wired cable coded by colors, as shown in the joined figure. The cable's standard length is 3 metres, but it could be supplied with a different length under demand.

The connection is the same for both display models DT-103GS and DT-103GD.

We don't supply any kind of connector to link the PC to the printer, but it could be supplied under demand.

The DT-103G support foot is optional, and is supplied under demand.

The connection considerations about the inputs and the DT-103G serial line are the same that the ones that have been specified for the DT-101G and DT-104G models. The only difference is in the external connection.

NOTE: MESSAGES PROGRAMMING.

In order to program the messages, it is necessary to make a bridge between the brown and white cables of the multi-wired cable, which goes from PC to display. In normal mode this bridge must be open.

4 PLC PROGRAMMING.

Due to the great number of types and models of existing PLC, it results impossible to give any kind of rules about their programming. All the instructions and directives in this manual will be always about the display.

The DT-100G display series can be controlled by any type and model of PLC (Programmable Logic Control), because the control is done by means of their outputs. It will be necessary only a maximum of 8 PLC outputs to control the display.

There are two kind of messages that the DT-100G displays can visualize :

Messages without variables.

Messages with variables.

The process to enable and disable these two kind of messages is different. The coding of the messages without variables from the PLC is very easy, and it won't become any trouble for you. The coding of the messages within variables is a little more complex.

4.1 Messages without variables.

When we use the display exclusively to display messages without variables, the PLC outputs could be of two different types statical, or relay.

To enable a message code we have to send to the 8 PLC outputs its binary code. The message will be shown while we maintain enabled its code. To disable the message we must send the code "0" to the outputs.

The number of PLC outputs necessary to send the messages to the display depends on the number of the message that our application will need.

PLC OUTPUTS	MESSAGES OF DISPLAY
1.....	2
2.....	4
3.....	8
4.....	16
5.....	32
6.....	64
7.....	128
8.....	253

Example: If we are supposed to have about 40 messages in our applications, it will be enough to use 6 PLC outputs attached to the 6 displays inputs (from D0 to D5) and let the other two inputs (from D6 to D7) without connecting. So proceeding like this we'll have a government capacity of 64 messages, and we keep two PLC outputs free.

4.2 Messages within variables.

When we use the display to send messages within variables, it will be absolutely necessary to use the 8 PLC outputs attached to the 8 display inputs (from D0 to D7) The 8 PLC outputs should be solid state because they will work high speed .

To enable a message within variables you should follow those directives :

Phase 1 - Enable the text.

Phase 2 - Transference and update of the variable.

Phase 3 - End of message within variable.

We'll use the same display inputs to send the message and the variables, although the behaviour of the inputs will be different in every case, depending on if we're working in a MESSAGE MODE or if we're working in the VARIABLE MODE.

The display inputs in the phase 1 work exactly in the same way as when we're working without variables. But in phase 2 and 3 their meaning and use it's totally different, as you have to treat them as individuals bits, as we're going to see.

4.2.1 Phase 1 - Enable the message within variable.

When the display receives a binary code corresponding to the address of a message, automatically will identify whether the message is within or without variable because we'll have programmed it previously from the DT program, and its behaviour will be according to the specifications of the selected mode (See figure 4.1)

Like in the message without variables, to enable a message within variables you have to send the suitable code in binary to the 8 displays inputs, but you must do it by means of pulses, as we're explaining here:

- Enable the message's binary code, during a time T1 (80mS a 500mS).
- Disable all inputs during a time T2 (20mS a 500mS).

In phase 2, the meaning of the inputs is going to change.

INPUTS MEANING IN VARIABLE MODE:

In phase 2 and 3, the meaning of the inputs from D0 to D7 is going to change thoroughly. Its meaning will be :

- **From D0 to D3** - Their are used to put the value of the digits in BCD.
D0=bit 0 D1=bit 1 D2=bit 4 D3=bit 8

Example : To send the "5" digit, the value from D0 to D3 must be:
D0=1 D1=0 D2=1 D3=0

- **D4** Not used.

- **D5 Message end.** Enabling this bit, the display will change from the mode within variable to the mode without variable .

- **D6 Data end.** Enabling this bit, the display will know that the next digit we send will be the first one DI 0.

- **D7 "STROBE"** Validates the value of the inputs D0 to D3.

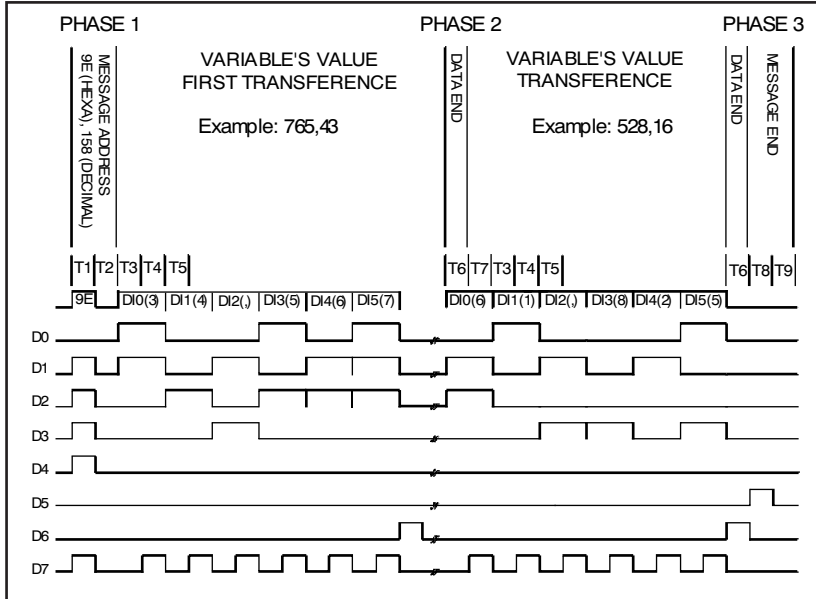
4.2.2 Phase 2 - Transference and update of a variable.

Once executed phase 1, the display is ready to execute the phase 2, receiving the value of the variables.

In this phase, the value of the variable will be sent sequentially digit by digit, always starting by the less significant digit, and ending by the more significant digit.

- The sequence starts by putting the value of the less significant digit DI 0 in the inputs D0 to D3. The display won't identify the digit until we enable the strobe input (D7). It is necessary a delay T3 (20mS to 50mS) between the sending of the digit value and the enabling of the strobe (D7), in order to avoid wrong readings of the digit's value. The strobe (D7) must be enabled during a time T4 (20mS to 500mS). The transference of the digit's value to the display is done when the strobe's flank is going up.
- Now the digit DI 0 and the strobe (D7) are enabled, and the first digit has been transferred. Before sending the second digit (DI 1) we have to disable the strobe (D7), maintaining it disabled during a time T5 (20mS to 500mS). Meantime while we're enabling the strobe (D7) we'll send the following digit (DI 1).
- Once time T5 has ended, we'll enable again the strobe during the time T4 for validating digit DI 1....and so on for the following digits of the variable.
- When we send the strobe corresponding to the last digit of the variable (the last digit will be the one according to the digit number assigned to the variable in the program DT), we must inform the display that the last digit has been send enabling End of Data bit (D6) during the time T6 (20mS to 500mS). The value of the variable won't be displayed until the End of Data bit (D6) enables.

End of data can be send after any intermediate digit, the digits of the variable will correspond to the ones sent until this moment, so it's possible to send variables with less digits than the number of digits that we had kept in the DT program.



When you have sent the End of Data (D6) we must wait a little time T7 (20mS minimum) in disabling state, and the display will remain waiting for the strobe (D7) corresponding to the first digit (DI 0) of the variable, or waiting for the value of the variable to change, and then restart the process again.

4.2.3 Phase 3 - End of message within a variable.

When we want to send a new message, if we are sending a message within variable, previously we must send the bit that will indicate the display to end with the actual message. This bit is D5 (End of Message) and it should remain a time T8 (20mS to 500mS).

After sending End of Message (D5), we should wait a time T9 (20mS to 500mS) for sending the address of the new message.

4.2.4 Errors transmitting variables.

If the delays specified before aren't respected, the display will show an error message coded as follows:

- ERROR 1: Addressing message time T1>500ms
Disabling all inputs time T2 < 20ms
- ERROR 2: Disabling all inputs time T2 > 500ms.
Enabling Strobe (D7) time T4 < 20ms
- ERROR 3: Enabling end of message (D5) time T8 > 500ms.
Disabling all inputs time T9 < 20ms
- ERROR 4: Enabling the Strobe (D7) time T4> 500ms.
Disabling the Strobe (D7) time T5 < 20ms
Disabling the Strobe (D7) time T5 > 500ms
Enabling end of data (D6) time T6 < 20ms
- ERROR 5: Enabling end of data (D6) time T6 > 500ms

When an error in transmission is found, to fix it you should sent to the inputs the code 253 (Hex. FD). You can do it also switching off/on the display.

4.2.5 Variables format.

The characters that can be sent as variables are the following:

CHARACTER	D3	D2	D1	D0	HEXADEC.
0	0	0	0	00
1	0	0	0	11
2	0	0	1	02
3	0	0	1	13
4	0	1	0	04
5	0	1	0	15
6	0	1	1	06
7	0	1	1	17
8	1	0	0	08
9	1	0	0	19
,	1	0	1	0A
.	1	0	1	1B
-	1	1	0	0C
?	1	1	0	1D
?	1	1	1	0E
?	1	1	1	1F

-The characters value from 0 to 9 in Hexadecimal is the same than in BCD (Binary Coded Decimal).

- The comma, “,” has an hexadecimal value of “A” (1010).

- The point, “.” has an hexadecimal value of “B” (1011).

- The subtraction sign “-“ has an hexadecimal value of “C” (1100).

- When the display finds an hexadecimal value of “D”, “E” y “F” it will show the sign “?” pointing out a wrong value.

- If the PLC sends more digits than the programmed ones, in the display will appear as many signs “?” as digits you had programmed before.

4.2.6 Points and comas programming.

The programming punctuation signs (points and comas) can be done in two different ways: editor mode, and PLC mode.

EDITOR MODE FROM THE DT PROGRAM.

You can do it by programming the points and comas at the moment of programming the text and variables with the DT program (See paragraph 2.1.5 message's editor manual TDL). In this option, the digits are sent together, and the display will place the points and comas at the programmed position. This option is the more advisable for applications where the position of the point and coma are invariable .

Example : In the case we wanted to send a temperature value, the text that should be programmed in the PC will be:

"TEMPERATURE" ###,## ° C"

Then if we send a sequence like the following one from the PLC:

- 1st Digit sent (DI 0) = 3
- 2nd Digit sent (DI 1) = 4
- 3rd. Digit sent (DI 2) = 5
- 4th. Digit sent (DI 3) = 6
- 5th. Digit sent (DI 4) = 7

The message that should appear will be: " TEMPERATURE 765,43°C ".

Remember that the format of a variable must be from 1 to 8 digits, that should be programmed at the moment of introducing the texts, including coma and/or points.

PLC MODE

Sending the characters (A or B in hexadecimal) at the moment of sending the variable. This option it's useful when the position of the coma is variable.

Example: If we want to send a temperature, as we have done before, but now modifying the position of the coma from the PLC. In this case the programmed text would be:

“TEMPERATURE ##### ° C” without placing the coma A (Hex), but reserving one more variable.

If we send the following sequence :

1st. Digit sent (DI 0) = 3
2nd. Digit sent (DI 1) = 4
3rd. Digit sent.(DI 2) = A
4th. Digit sent (DI 3) = 5
5th. Digit sent (DI 4) = 6
6th. Digit sent (DI 5) = 7

The message that will appear on the display would be : “TEMPERATURE 765,43°C”, exactly the same result that appeared in the example before (EDITOR MODE). If we send the coma in the 4th digit, instead of the 3rd one, and the message would be: “TEMPERATURE 76,543°C” .

The two programming modes (EDITOR MODE or PLC MODE) can be used together in the same variable. In this case the PLC MODE will have the priority over the EDITOR MODE.

Example : Programming a message within a variable in EDITOR MODE, like this:

“SPEED ###,### m/s”

1st Digit sent (DI 0) = 6
2nd Digit sent (DI 1) = 5
3rd Digit sent (DI 2) = B
4th Digit sent (DI 3) = 4
5th Digit sent (DI 4) = 3
6th Digit sent (DI 5) = 2
7th Digit sent (DI 6) = 1

The result will be : “SPEED 1234,56 “ instead of “SPEED 123,456” that would result in the EDITOR MODE.

For the negative numbers, you can send the sign “-“ (hexadecimal C) with only condition that it has to be the last digit.

At the moment of programming a variable, you're to consider that the minus sign (-) the coma (,) and the point (.) occupy one digit every one of them; therefore a negative number with a coma or a point could be only 6 digits long.

Examples.

RIGHT	WRONG
12345678	123456789 More than 8 digits
-1234567	-1234,567 More than 8 digits
-123,456	2,34,56 More than one sign
-1,4	,2456-7 More than one sign
0.0003	
.004	

CORRECTION OF ZEROS.

the display corrects automatically the zeros to the left at the left of the variable as shown here:

Programmed variable	Sent variable	Displayed value
#####000001212
###,## 000,01 0,01
###,## 3 0,03
##,### -48 -0,048
385 385

4.3 Printing messages.

Every time that you enable a message programmed previously as "printable" (see paragraph 2.1.4 of the message editor TDL manual), it will be keep in a buffer specially designed to store all the messages that have been enabled .

The storage is done sequentially, and in the same order in which they were enabled. The maximum capacity of this buffer is 255, in case of surpassing this number, without having voided the buffer, the messages will be lost sequentially, from the older to the newer.

The most important advantage of this buffer is that you don't need to have the printer switched on permanently. You only have to switch on the printer when you want to print out the messages stored in the buffer. This's very useful specially for the maintenance personnel, because with just one printer they can list all the messages from the different displays. Obviously, you can let the printer be on permanently, if you want.

To print all the buffer's messages, you only have to send the hexadecimal printing code "FE" (1111 1110). This code will be sent as any other message.

If you want to empty the buffer, you only have to send the hexadecimal code "FF" (1111 1111). Once the buffer has been emptied , all the messages stored before will be lost, even if they have been printed or not.

If we use this commands with ability, we can do different types of listings: in groups, message by message, etc.

The printer has to be serial type, with the configuration and connection explained on the point 3.2 of this manual.

5.- PROBLEMS. What to do if...

A) When we switching on the display, **it doesn't work.**

- Verify the power supply cable of the DT-101/103/104G.
- Verify the external fuse installed by the user.
- Verify the internal fuse. For the DT-101G, remove the box rear panel, extract the printed circuit card and locate the fuse. For the DT-103G, loose the four holding screws , and locate the fuse in the motherboard of the printed circuit. For the DT-104G, loose the four screws of the rear panel and lift it to access to the fuse located on the printed circuit card.

B) The display always show the 00 code message.

The display doesn't receive any code, verify:

- The wiring of the inputs and their common.
- The power supply from 12 to 24 VDC and its ripple.
- The PLC doesn't send any code, verify the PLC outputs.

C) When sending a code, **appears an unexpected message .**

-Verify that the code sent corresponds with the message coded in the message's list edited with the program DT.

- Verify that the wires of the inputs are attached correctly.
- Verify that the wires of the inputs are connected .
- The inputs power supply isn't between 12 and 24 VDC.

D) **The display shows different messages every now and then.**

-The power supply's inputs ripple is greater tan 500mV. Substitute or fix the external power supply.

-We send a message within a variable and the display doesn't understand it. Therefore, it is sending messages which address corresponds with the variable's digit's value that the PLC is sending. Verify the texts edited within variables and/or the PLC's program.

E) During the transmission of the message, it appears the text "**TRANSMISSION ERROR**" on the display.

- Verify the cable between the computer and the display:

-The display should show the text "**PROGRAMMING**". If it doesn't, verify the bridge between the pins 3 and 4 in the display's connector.

-If everything seems to be right, then switch off/on the display and try again.

F) During the transmission of the messages between the PC and the display, in the PC appears the text "**ERROR IN SERIAL TRANSMISSION LINE COM1 or COM2**"

-Verify the cable between the computer and the display.

-Verify that the configuration of the serial line COM1 or COM2 corresponds with the port in which we have attached the display's cable. Change the configuration of the serial line PC'S1 or PC'S2. See 2.5 from the editor TDL manual.

-Verify that the file of messages that you have created matches with the type of display we are working with. That means, you can't send a file that has been edited for a DT-101G to a DT-103G/104G display.

- Verify that the display shows the text "PROGRAMMING".

G) On the display appears the text "**Number of characters > 7423**"

-The maximum number of characters that the EEPROM memory can store has been surpassed .

- Make the texts shorter, or delete some messages.

H) In the computer appears "**COM1 or COM2, NOT INSTALLED**"

- The computer doesn't have any serial card installed, or they doesn't work..

- I) In the computer appears «**DISK NOT READY**».
- The door of the drive in which you're working isn't closed.
- J) In the PC screen appears “**CANNOT PRINT A FILE NOT EDITED**”
- The name of the file that you want to print is wrong, verify and try again.
- K) In the PC screen appears “**BAD FILE NAME**”
- The file name doesn't fulfil the rules of the MS-DOS operative system.
- L) In the PC screen appears “**FILE OF MESSAGES IS NOT FOR**”

-You're trying to edit an existing file that has been edited before for another kind of display. You have to configure the type of display corresponding with the file you want to edit. One file that has been edited for one type of display cannot be converted onto another type of display.