ALARMS AND TEXT DISPLAY

DISP BUS

User's manual

Notes Rel.: 1.1 – January 2001

- Information in this document may be modified without notice.
- MS-DOS and WINDOWS are registered mark of Microsoft Corporation.
- For additional details contact: **DUEMMEGI** srl, via LONGHENA 4 - 20139 MILANO
  Tel.: 02 / 57.30.03.77 – Fax: 02 / 55.21.36.86

INDEX

1- INTRODUCTION

DISP family of **DUEMMEGI** displays allows to show custom messages in many applications, such as industrial and domestic plants, hospital, electrical machines, etc..

Thanks to the flexibility of these devices, the displays of **DISP** family make easy to understand any information related to the occurrence of alarms or events.

DISP devices may be employed in many applications; in example:

- Electrical machines
- Industrial plants
- Domestic plants
- Building automation
- Signalling of alarm and/or operating status

The following models are available:

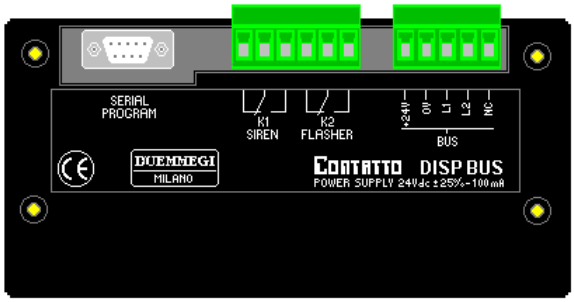
DISP 16D	16 messages display driven by direct inputs
DISP 15B	15 messages display driven by binary-coded inputs
DISP 31B	31 messages display driven by binary-coded inputs
DISP 63B	63 messages display driven by binary-coded inputs
DISP 127B	127 messages display driven by binary-coded inputs
DISP 255B	255 messages display driven by binary-coded inputs
DISP-S	500 messages display driven by RS232 or RS485 serial input and printer output
DISP BUS	250 messages display for **DUEMMEGI Contatto** bus system

This manual refers to DISP BUS model.

2- DISP BUS: GENERAL CHARACTERISTICS

LCD display 2 x 16 characters with back-lighting
Message programming by front panel pushbuttons or by PC
Messages recalled through the **DUEMMEGI Contatto** bus line
250 messages, each one made by 2 main lines and 2 hidden lines
1 base message made by 2 lines (stand-by message)
1 alarm pending message made by 2 lines
Events storing (MEM mode) or real time display (NOMEM mode)
Storing of up to 32 events in chronological order (in MEM mode)
2 free-potential contacts (relays internally driven in LOCAL mode or through the bus in REMOTE mode) for siren e flasher (or any other signalling device)
The status of all pushbuttons on the front panel and of internal relays can be monitored through the bus
Remote ACK and RESET commands can be sent through the bus
Display contrast can be adjusted by front panel pushbuttons

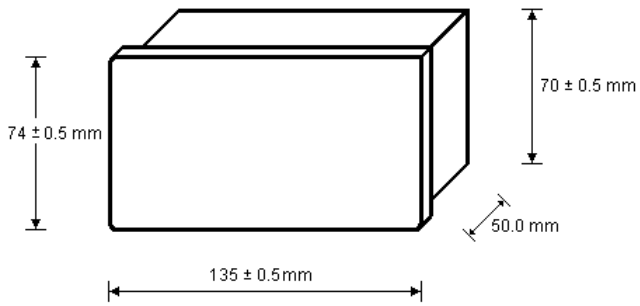
3- CONNECTIONS

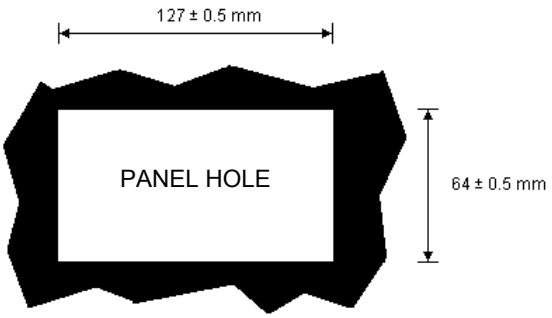


Description of the connections:

+24V	+24Vcc ± 25% positive power supply
0V	Power supply common
L1	L1 bus line of **DUEMMEGI Contatto** system
L2	L2 bus line of **DUEMMEGI Contatto** system
SERIAL PROGRAM	Connector for messages programming by PC
K1 (SIREN)	Free-potential contact of internal relay for alarm device (e.g.. siren)
K2 (FLASHER)	Free potential contact of internal relay for alarm device (e.g. flasher)

4- OUTLINE DIMENSIONS





5- TECHNICAL DATA

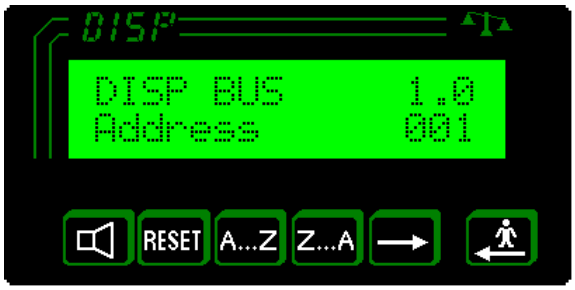
Power supply	24Vdc ± 25%
Current consumption	100mA MAX
Display	LCD with back-lighting and contrast adjust by front panel
Characters (each line)	16
Lines	2
Characters height	8 mm
Number of messages	250 (4 lines each one) + 1 of 2 lines (stand-by message) + 1 of 2 lines (alarm pending message)
Connections	By removable terminal blocks
Protection degree	Front: IP 53 – Back: IP20

K1 and K2 output contacts:

Max switching voltage	60Vdc or 125Vac
Max switching current	1A
Max operating power	30W in dc – 60VA in ac

6- FUNCTIONAL DESCRIPTION

DISPBUS shows, at power-on and during 1 second about, the type, firmware version and base address (e.g. as shown by the following figure).



Then, the stand-by message will be displayed (line 1 and line 2 of the message zero).

DISPBUS behaves exactly as an assemblage made by an input module and two output modules of **DUEMMEGI** Contatto family.

For this reason, a **base address** (Address in the figure on left-side) has to be assigned to the device. This address may be in the range **1 to 126**; said "x" the base address assigned to DISPBUS, it will take automatically the following configuration:

- Input module "x": reports the front panel pushbuttons status (6 points) and the internal relay status (2 points)
- Output module "x": command, in binary format, for the stored messages recalling
- Output module "x+1": command to force, through the system bus, ACK and RESET events (2 points) and remote command of the two internal relay (2 points)

The base address may be programmed both by the front panel pushbuttons and by a Personal Computer (see paragraph describing programming feature).

WARNING: DISPBUS base address cannot be programmed by Contatto ***FXPRO programmer***.

6.1- Operating modes

DISPBUS allows several operating modes:

MEM (memory): each message, when recalled, remains in the displaying queue even if the binary code sent by bus changes; if several messages had recalled, these ones will be cyclically displayed with a period of 2 seconds about

NOMEM (no memory): displayed message is always that related to the last binary code received by bus

LOCAL: two output relays are locally handled by DISPBUS, in a way similar to the standard ISA-M alarm sequence

REMOTE: two output relays are exclusively driven by the bus; in other words, handling of the relays will be left to the **Contatto** system controller (MCP)

Setting of operating mode may be done by front panel pushbuttons or by PC (see paragraph describing programming feature).

6.2- Relay operation in LOCAL mode

In LOCAL mode and without alarm pending, K1 relay (siren) is de-energized and K2 relay (flasher) is energized; the siren must be then connected to the normally open contact and the flasher must be connected to the normally closed contact. This last connection ensures at least a visual warning when the power supply of DISPBUS fails (intrinsic safety). **The just described connections will be assumed in the following two paragraphs**.

6.3- LOCAL – MEM operation

When DISPBUS is set as LOCAL and MEM, the operating sequence is the following:

- when DISPBUS receive, by the bus, a non-zero binary code, the siren and flasher will be activated; the related message will be shown on the display, alternated to the alarm pending message (message 000,3-000,4 – see paragraph describing the programming feature)
- if another non-zero binary code is received by the bus, the related message will be added to the queue and the display will cyclically show all recalled message and the alarm pending message
- pushing  button (ACK), the siren will be switched off, the flasher will remain in its active state and the display will show, during some seconds, the first recalled message (first out feature), then the cyclic displaying restarts
- after removing of the alarm causes, DISPBUS may be restored by a RESET sequence (pushing, in sequence, of pushbuttons 

and ): the siren will be switched off, the displaying queue will be reset and the stand-by message will be displayed (message 000,1-000,2, see paragraph describing the programming feature)

Notes:

1. if a new alarm occurs after ACK, the siren will restart
2. if some alarms are still active after a RESET sequence, the described sequence restarts as soon as a non-zero binary code will be received by the bus
3. if the 16th character of line 1 of a message is the symbol #, then the receiving of the related binary code from will not influence the internal relays
4. DISPBUS stores, in chronological order, up to 32 events; any other event exceeding 32 will be ignored; the message shown after the alarm pending message is the first occurred, the second one is the second occurred and so on
5. pushing  button, it is possible to change from automatic (cyclic) to manual displaying; in manual mode, the message related to the pending alarms can be displayed by pressing 

(backward) and 

(forward) buttons

6.4- LOCAL – NOMEM operation

When DISPBUS is set as LOCAL and NOMEM, the operating sequence is the following:

- when DISPBUS receives, by bus, a non-zero binary code, the siren and flasher will be activated; the related message will be shown on the display
- pushing  button (ACK), the siren will be switched off, the flasher will remain in its active state and the display will show the message recalled at the previous point
- when a zero binary code will be received by the bus, the siren and falsher will be switched off and the stand-by message will be displayed

Notes:

1. if more alarms are pending, the bus controller (MCP) cyclically sends to DISPBUS the related binary codes; this means that the siren restarts (if silenced before by the ACK pushbutton) at every binary code. This strange operation is an implicit operation because DISPBUS has not in the queue the alarm already acknowledged (remember that we are in NOMEM mode). It is better to use the REMOTE-NOMEM mode and handle the internal relays by the program downloaded in MCP controller
2. the RESET sequence in LOCAL-NOMEM mode has no relevance

6.5- REMOTE – MEM operation

When DISPBUS is set as REMOTE and MEM, the operating sequence is as before described for LOCAL-MEM mode, with the exception that two internal relays will be always de-energized; the handling of these relays, when required, must be implemented in the program downloaded in the system controller (MCP).

6.6- REMOTE – NOMEM operation

When DISPBUS is set as REMOTE and NOMEM, the operating sequence is as before described for LOCAL-NOMEM mode, with the exception that two internal relays will be always de-energized; the handling of these relays, when required, must be implemented in the program downloaded in the system controller (MCP).

6.7- RESET of alarms

If the chosen operating mode is without storing (NOMEM), the RESET sequence has no relevance.

On the contrary (MEM mode), the RESET sequence allows to restore the situation; this means that all alarms in the queue will be removed.

The RESET sequence is the following:

- silence the siren by pushing  button
- push  button; DISPBUS will displays the text shown in the left-side figure
- push  button to confirm the RESET of the message queue

If the confirmation does not occur in 2,5 seconds, the RESET request will be automatically rejected.

6.8- Bus loss warning (BUS FAILURE)

When the signal on the L1 and L2 bus lines is not detected by DISPBUS (e.g. when the MCP controller is disconnected, or when the bus is broken), DISPBUS warns about this condition displaying the text shown in the left-side figure. Note that in this condition the base address is displayed too.

7- INPUT AND OUTPUT POINTS

As said before, DISPBUS DISPBUS behaves exactly as an assemblage made by an input module and two output modules of **DUEMMEGI Contatto** family. Said "x" the base address chosen for DISPBUS, the following ***input points*** are available:

Address/point	Function	Address/point	Function
x.1	Status of button	x.5	Status of button
x.2	Status of button RESET	x.6	Status of button
x.3	Status of button A..Z	x.7	Status of relay K1 (siren)
x.4	Status of button Z..A	x.8	Status of relay K2 (flasher)

The available ***output points***, that are then driven by the bus, are the following:

Address/point	Function	Address/point	Function
(x+1).1	Remote command button	(x+1).5	Not available
(x+1).2	Remote command button RESET	(x+1).6	Not available
(x+1).3	Not available	(x+1).7	Command for relay K1 (siren)
(x+1).4	Not available	(x+1).8	Command for relay K2 (flasher)

Note:

1. Remote commands for ACK and RESET pushbuttons allow the implementation of remote acknowledge and reset functions; in other words, pushing the related button on the front panel or send through the bus a command of the same function are equivalent actions

2. The commands of relays K1 and K2 by the bus are allowed when DISPBUS is set in REMOTE mode

The output address x is reserved to the binary code for the recalling of the message stored in DISPBUS.

8- PUSHBUTTONS FUNCTION

The 6 pushbuttons on the front panel, during normal operation, perform the following functions:

	ACK: acknowledge and siren silencing
	Request of queue reset; the confirmation may occur within 2,5 seconds by ACK button
	Show next message when the automatic cyclic displaying of messages is disabled
	Show previous message when the automatic cyclic displaying of messages is disabled
	Switch from the automatic to manual displaying of messages and vice-versa. When in manual mode, DISPBUS return to automatic mode when a new alarm occurs
	Show the auxiliary lines of the current message (lines 3 e 4 of each message)

During the programming mode, these pushbuttons performs other functions; refer to the related paragraph for details.

9- PROGRAMMING

9.1- Manual programming by the panel pushbuttons

The message programming may be executed by the front panel keyboard of DISPBUS. To enter the programming mode, contemporarily push the buttons  ,  and  .

During the manual programming the display shows the number of the current message and the number of the current line on the first line, and the text of the related message on the second line. First displayed line is 000,1 (message 0 line 1) and it is related to the stand-by message (no alarm pending).

More precisely:

- 000,1 and 000,2 are the lines displayed during no alarm pending condition (stand-by message)
- 000,3 and 000,4 are the lines displayed after the last alarm currently pending (the so-called alarm pending message) during the cyclic displaying of the alarms
- Lines from 001,1-001,2 to 250,1-250,2 will be displayed when DISPBUS receives the related binary code (in the range 1 to 250); lines from 001,3-001,4 to 250,3-250,4 are the auxiliary lines of the related main message

Messages from 251 to 254 are not used.

The lines of message 255, not available from the bus, define the DISPBUS setting as here bottom explained:

- 255,1: defines LOCAL or REMOTE mode
- 255,2: defines MEM or NOMEM mode
- 255,3: defines the base address of DISPBUS (it is a number in the range 0 to 126)
- 255,4: defines LCD contrast (it is a number in the range 0 to 100)

The programming procedure counts two operating mode:

- message searching
- message edit

These modes can be easily identified because in the edit mode a cursor under the current character to be edited is shown (the cursor is a small line under the character); the cursor is not displayed during searching mode.

In the searching mode the pushbuttons perform the following functions:

A..Z	Next message
Z..A	Previous message
→	Enter the edit mode
←	Enter the edit mode (same function as previous button)

In the edit mode the pushbuttons perform the following functions:

A..Z	Next character (or next option)
Z..A	Previous character (or previous option)
→	Move cursor to right
←	Move cursor to left
🔊 + ←	Copy current character to left-side position
🔊 + →	Copy current character to right-side position
🔊 + A..Z	Insert character «A» at the cursor position
🔊 + Z..A	Insert syimbol «#» at the cursor position

Valid Functions for both modes:

RESET + A..Z	Copy current message in the clipboard
RESET + Z..A	Paste the message from the clipboard to the current message
→ + ←	Quit programming
🔊 + RESET	Save the current message

9.2- Programming by Personal Computer

DISPBUS features a DB9 connector to communicate with a Personal Computer through the RS232 serial port.

The required connection cable is a standard type with 9-pole male connector at one edge and 9-pole female connector at the other edge. The two connector must be pin to pin connected (pin 1 to pin 1, pin 2 to pin 2, etc.).

Programming by PC

DISPBUS programming by PC can be performed in two ways:

- **from WINDOWS using the tool program "DISP"** free-distributed by **DUEMMEGI**
- **from MS-DOS**

Use of tool program "DISP" is more simple and more efficient than the MS-DOS option, so it may be preferred; for details on the use of this program, refer to its on-line help.

When the tool program "**DISP**" is not available, use, in MS-DOS mode, the procedure here bottom described.

The serial programming by PC in MS-DOS mode is performed in 2 steps:

- writing of a file containing the messages to be displayed and the settings
- downloading of the file to the memory of DISPBUS

Regarding to the first step, see next paragraph; the sequence of file downloading is here bottom described:

connect the serial port COM1 (or COM2) of PC to DISPBUS by a standard RS232 cable (9-pole, pin to pin connected). Before to switch on DISPBUS, set the serial port **COM1** digitizing on the keyboard of PC the following commands:

- MODE COM1:1200,N,8,1 <enter>
- DEBUG <enter> (the prompt – will appear)
- O 3fc 3 <enter> (the first characters of this command is O, not zero)
- Q <enter>

If using the serial port **COM2**, the proper commands are the following:

- MODE COM2:1200,N,8,1 <enter>
- DEBUG <enter> (the prompt – will appear)
- O 2fc 3 <enter> (the first characters of this command is O, not zero)
- Q <enter>

At this point, switch on DISPBUS; the file containing the messages will be downloaded by the following command:

- COPY FILE.MSG COM1: <enter> (when using COM1)
- COPY FILE.MSG COM2: <enter> (when using COM2)

FILE.MSG is the name of the file containing the messages (FILE may be any other name, limited to 8 characters).

To quit the programming mode, remove RS232 cable. During programming, the display shows the message "RS232 PROGRAM" on the first line and a animated bar on the second line to give an indication of the percentage of the elapsed time against the total time required.

If an error occurs during programming, the message "ERROR ⇒ABORTED" will be displayed on the second line. In this case, switch off DISPBUS to quit the error condition and check that the file does not contain syntax errors. When DISPBUS detect that the PC does not send any other character, it assumes that the programming sequence is ended and it shows on the second line the message "⇒COMPLETED".

To return to normal operating mode, remove RS232 cable; on the contrary, if a new downloading is required, switch off and then switch on again DISPBUS.

9.3- Syntax of message file

The file containing the message is an ASCII file and may be written using any text editor (e.g. EDIT of MS-DOS). Each line is made by the message number, 16 characters of the text to be displayed and optional comments, separated by commas:

MMM,L, MESSAGE TEXT , comment

The following syntax is mandatory:

The line must start with the message number MMM (in the range 000 to 255)
MMM must be in 3 digit format
L (line number) must be in the range 1 to 4
MESSAGE TEXT must contain 16 characters (no one more, no one less); if last character of line 1 of a message is the symbol «#», then the siren will be disabled for that message
Commas before and after the line number L are mandatory
Any characters after the 16th will be considered as a comment and will be ignored during the download to DISPBUS

Empty lines are not allowed. The file may contain only part of the messages (it is not required to program all available messages).

Message 000 and 255 are reserved for stand-by message, alarm pending message and settings; more precisely:

Lines 000,1 and 000,2 forms the stand-by
Lines 000,3 and 000,4 forms the alarm pending message and will be shown during the cyclic displaying of occurred alarms
255,1: defines LOCAL or REMOTE mode
255,2: defines MEM or NOMEM mode
255,3: defines the base address of DISPBUS (number in the range 0 to 126)
255,4: defines the contrast of LCD (number in the range 0 to 100)

Example of message file:

```
000,1,    NO ALARM    ,stand-by text, line 1
000,2,    PENDING     ,stand by text, line 2
000,3,    WARNING:    ,alarm pending text, line 1
000,4,   ALARMS PEND. ,alarm pending text, line 2
001,1,   FAULT PUMP   ,message 1,line 1
001,2,  BOILER ROOM   ,message 1,line 2
001,3, CALL HYDRAULIC ,message 1,line 3
001,4,    01/647589   ,message 1,line 4
002,1,   TANK H20    #,message 2,line 1, siren disabled
002,2,    EMPTY       ,message 2,line 2
002,3,     OPEN       ,message 2,line 3
002,4,    COCK  1     ,message 2,line 4
255,1,REM             ,mode REMOTE
255,2,NOM             ,mode NOMEM
255,3,023             ,base address 23
255,4,060             ,contrast 60%
```

10- PROGRAMMING MCP IN DISPBUS APPLICATION

This paragraph describes, by an example, the implementation of an alarm sequence totally handled by MCP controller of **Contatto** system.

The DISPBUS in the following example **must** be set as NOMEM and REMOTE; the base address is chosen as 003.

Description of the system to be implemented:

given 8 inputs (`I1.1 ÷ I1.8`), we want that at the activation of each input the following events occur:

- The message related to the occurred event will be displayed
- The event must be stored, then, even if the input returns to its non-active state, the alarm condition remains active
- Siren and flasher output will be activated

We want to silence the siren pushing ACK button on DISPBUS front panel; we want that this button does not influence the flasher output and the displaying of DISPBUS. Moreover, we want that RESET button on DISPBUS front panel restore the “system”, this means to remove the pending alarm from memory, but only if the related input has returned to its non-active state, and, if no more alarms are pending, the flasher output must be switched off.

After each acknowledge, any new alarms (that means not still stored) must activate the siren again.

In addition, the RESET button must not have any effect if the siren was not silenced before by ACK button.

The program for **Contatto MCP** controller that implements the just described specification is the following:

```
////////////////////////////////////////////////
// MCP PROGRAM FOR ALARM HANDLING WITH DISPBUS //
////////////////////////////////////////////////
// ALARM MEMORY DEFINITION: EACH MEMORY CELL IS SET BY RELATED INPUT
// (FROM I1.1 TO I1.8) AND RESET BY RESET BUTTON ON DISPBUS (I3.2),
// BUT ONLY IF THE SIRENA (O4.7=V202) WAS SILENCED AND RELATED INPUT HAS
// RETUORNED TO ITS NON-ACTIVE STATE
V1 = SI1.1 & (RV203 | I1.1)
V2 = SI1.2 & (RV203 | I1.2)
V3 = SI1.3 & (RV203 | I1.3)
V4 = SI1.4 & (RV203 | I1.4)
V5 = SI1.5 & (RV203 | I1.5)
V6 = SI1.6 & (RV203 | I1.6)
V7 = SI1.7 & (RV203 | I1.7)
V8 = SI1.8 & (RV203 | I1.8)

// SIREN OUTPUT (O4.7): EACH NE ALARM SWITCH ON THE SIRENA; THE SWITCH OFF OF
// THE SIREN OCCURS BY ACK BUTTON ON DISPBUS (I3.1)
V201 = TV1 | TV2 | TV3 | TV4 | \
       TV5 | TV6 | TV7 | TV8 | \
       RI3.1
V202 = SV201 & RI3.1 & R!V998
O4.7 = V202

// VIRTUAL POINT TO RESET THE ALARM MEMORY CELLS: V203 WILL BE ACTIVATED
// PUSHING RESET BUTTON (I3.2), BUT ONLY IF THE SIREN (O4.7=V202) IS OFF
V203 = !V202 & I3.2

// FLASHER OUTPUT (O4.8), DEFINED AS OR OF ALARM MEMORY CELLS; THE FLASHER
// WILL BE ACTIVATED UNTIL ALL ALARMS WILL BE REMOVED AND THE SYSTEM WILL
// BE RESET
V204 = V1 | V2 | V3 | V4 | \
       V5 | V6 | V7 | V8
O4.8 = V204

// BINARY OUTPUT BLOCK DEFINITION
BINARY 3 ( \
        B001 = V1    \
        B002 = V2    \
        B003 = V3    \
        B004 = V4    \
        B004 = V5    \
        B004 = V6    \
        B004 = V7    \
        B004 = V8    \
)
```

This program may be easily modified to be adapted to the various real applications.