

Inhaltsverzeichnis

DEVICE HMI2	3
1. Introduction	3
1.1 Device declaration	3
1.2 Operation	3
1.2.1 The display	3
1.2.2 Recursive views	4
1.2.3 Data entry	4
1.2.4 Keyboard functionality in data entry	4
1.2.5 Values of variables related to buttons or leds	5
1.2.6 Values of the dis0÷31 variables	7
1.3 Parameters table	7
1.4 States table	12
1.5 Commands table	12

DEVICE HMI2

1. Introduction

HMI2 is a device designed to perform the functions of an interface between the user, a display up to 32 digit, leds (up to 32) and a keyboard (up to 32 keys). Manage 3 recursive contemporary views configurable each with its own number of characters, decimal point and position within the display. It also has the capability of data entry (introduction of a data via the keyboard) fully configurable than the number of characters, at the location and decimal digits with the ability to control the upper and lower limits of the data introduced.

1.1 Device declaration

The device declaration mode in the configuration unit is:

```
-----
; Internal device declarations
-----
<device_name>    HMI2  TCamp
```

Where:

<device_name>	the name assigned to the device
HMI2	keyword that identifies the device
TCamp	sample time device (3÷250 ms)



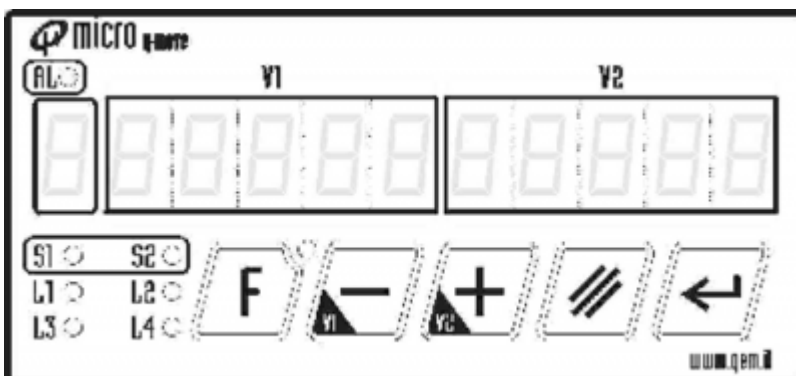
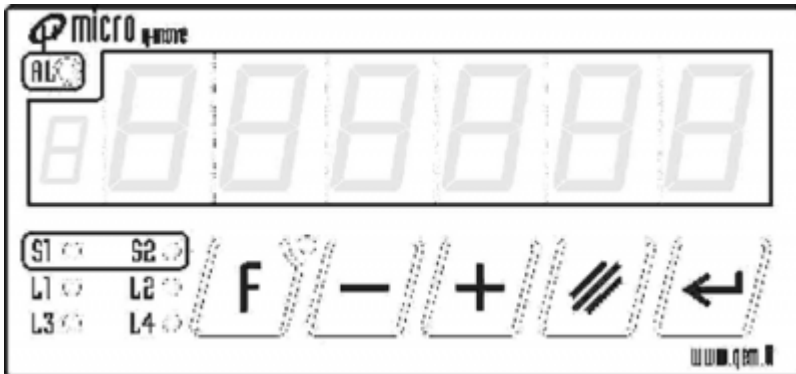
Attenzione: È necessario che tutte le voci di definizione siano presenti sulla stessa linea

1.2 Operation

We analyze in more detail the characteristics of the HMI2 device.

1.2.1 The display

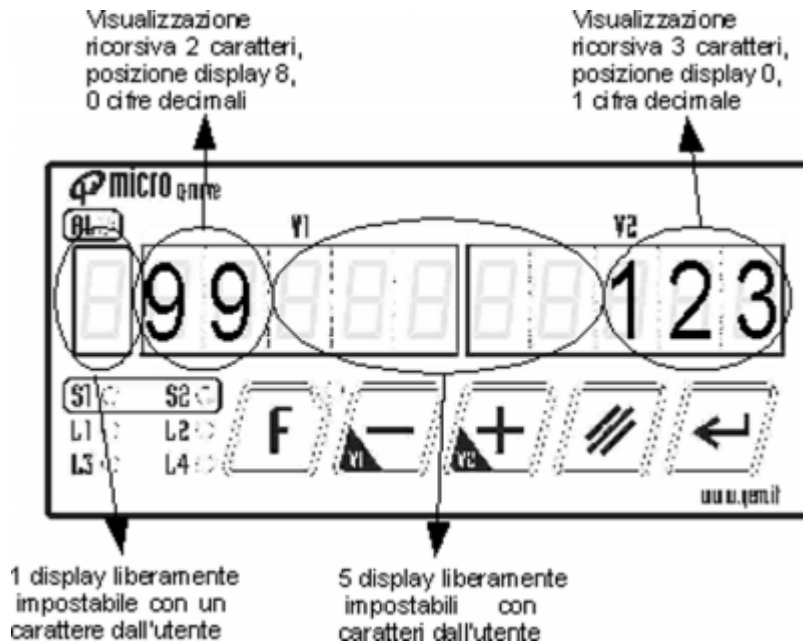
The display consists of a series of digits (characters) variables depending on the hardware platform of microQMove that you are using. For example, in a display digits D221 or D983 are 7, While the D231 digits arrive to 11.



The pictures are presented only as an example because the HMI2 device has an independent operation by the instrument that you are using.

1.2.2 Recursive views

For recursive view is a view that is updated continuously, more precisely at each sampling time of the device. With the HMI2 You can enable up to 3 of these, independent and positioned at within the display. Depending on the number of characters that you configured the display, the display area is used by the view itself so that in that area can no longer write the characters because they would be immediately overwritten.



In the picture there is an example of how you might configure the views to your liking. In this case there are 2 views of 2 characters and the other of a recursive 3 characters placed respectively at position 8 and position 0. Recursive display from 3 characters also has the number of decimal places set to 1. In display not occupied by recursive views you can set the characters (numbers or letters) so that they are showed; You cannot set characters on display at locations 0,1,2 and 8,9 Because views to overwrite them recursive.

For each recursive display the maximum and minimum values that can be showed depends as mentioned by number of digits and the sign enable. If the data to be displayed is less than the minimum or greater than this maximum value, the display will shows the out of range characters.

1.2.3 Data entry

Data entry functionality allows the user to enter a numeric value from the keyboard as it can configure the number of characters entered, the position on the display, the position of the decimal point. You can also exit the data entry with a button at will, or with special commands. A particular parameter to bit field allows you to enable control of the value of the data introduced in the upper and lower limits (set using the appropriate parameters), to enable the introduction of signed or unsigned data, and other features of tables description parameters and commands. This configurability allows through a few instructions of Qcl language to meet the various needs of human-machine interface.

1.2.4 Keyboard functionality in data entry

We will now detail the instructions in order to introduce keyboard data specifying the key functions. The HMI2 device provides the user with two ways to insert data, the standards and how they increase/decrease.

1.2.4.1 Standard mode

The table summarizes the functions of each key:

Key	Functionality
ENTER key	Confirmation of the introduced data: bounds checking of the datas in introduction (If enabled with the appropriate bits of the <i>deflags</i> parameter), copy the input value to the <i>devalue</i> parameter and produces the output from the insert data function.
CLEAR key	Delete/restore of data: execute alternately zeroing data in introduction, and restoration of its initial value.
PLUS key	Increases selected digit: execute the increase of the selected number.
MINUS key	Selection digit: moves the selection of the digit on the right than the one selected.

1.2.4.2 Increase/decrease mode

The table summarizes the functions of each key:

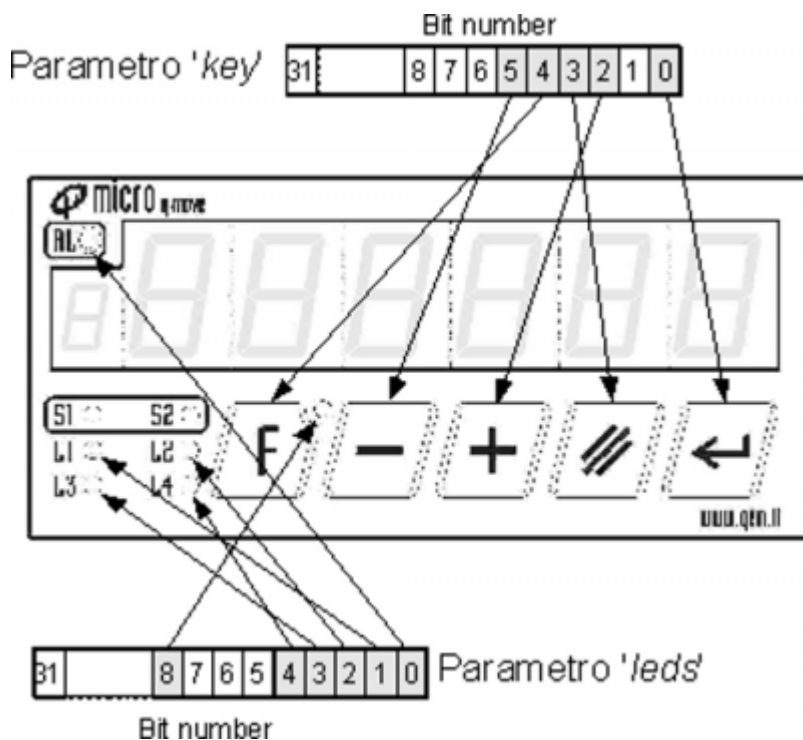
Key	Functionality
ENTER key	Confirmation of the introduced data: execute the bounds checking of the data in introduction (if enabled with the appropriate bits of the <i>deflags</i> parameter), copy the input value to the <i>devalue</i> parameter and produces the output from the function to insert data.
CLEAR key	Delete/restore of data: execute alternately zeroing introduction data, and the restoration of its initial value.
PLUS key	Increase value in introduction: execute the increase in introduction. While pressing the button, the growth is continuous and with speed which exponentially increases.
MINUS key	Decrease value in introduction: execute the decrease in value in introduction. While pressing the button, the decrease is continuous and with speed which exponentially increases.

When the zero bits of the *deflags* parameter is 0 the DATAENTRY functionality can be used for a easy view. This feature can be handy when the figure has a static value and therefore do not need to use recursive views. For example, if you need to display an error message and the error code itself for some time, the function that displays such a message can be achieved without having to alter the programming of recursive views.

1.2.5 Values of variables related to buttons or leds

This section shows the bit assignments of variables in charge in addition to the keys and leds on the various hardware MicroQmove.

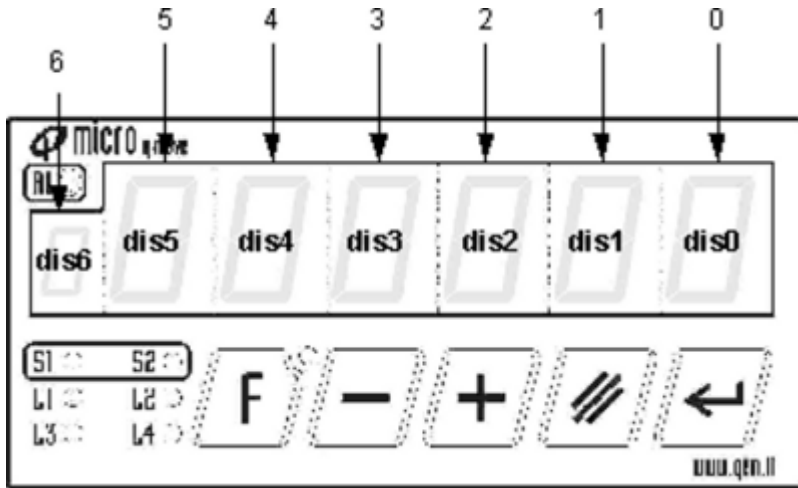
1.2.5.1 Keyboard, display and leds for D221 series



In summary:

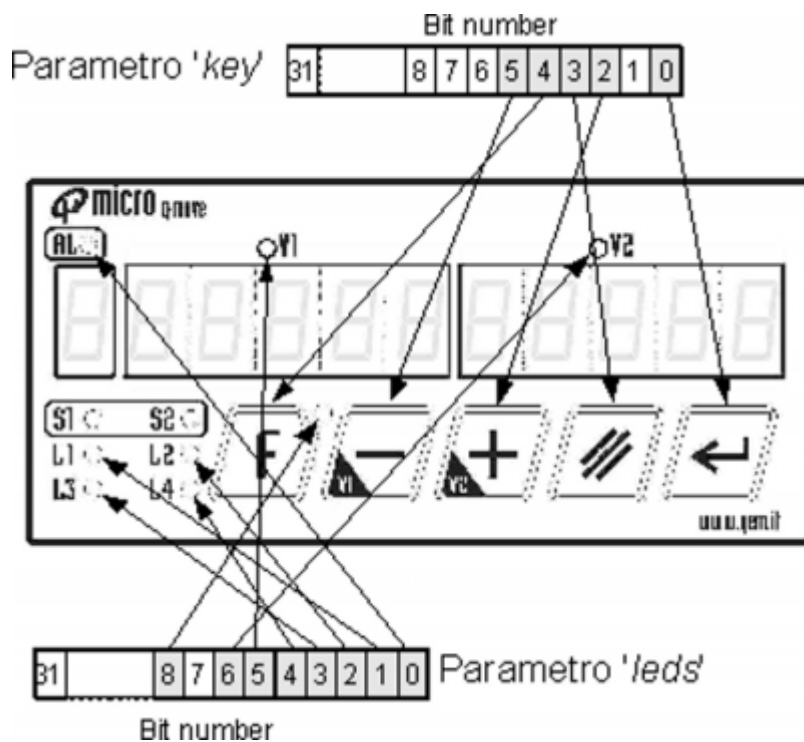
Key	Value key parameter
ENTER key	1
CLEAR key	8
PLUS key	4
MINUS key	32
F key	16
F + CLEAR keys	24

Offset value in relation with the digit



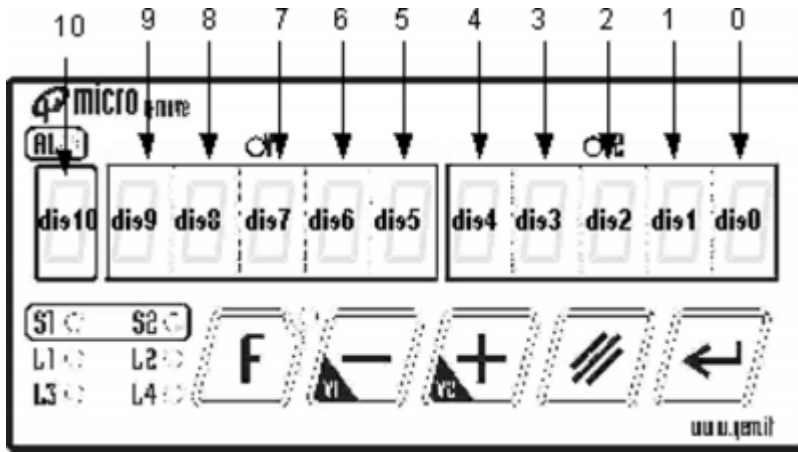
Name of dis0...31 parameters in relation to the digit

1.2.5.2 Keyboard, display and leds for D231 series



In summary:

Key	Value of key parameter parametro key
ENTER key	1
CLEAR key	8
PLUS key	4
MINUS key	32
F key	16
F + CLEAR keys	24



1.2.6 Values of the dis0÷31 variables

In this section shows the codes to use for viewing the and special characters. This codes must be used for the *dis0÷31* parameters.

Value	Character	Value	Character	Value	Character	Value	Character
0	0	14	E	28	J	42	~
1	1	15	F	29	c	43	M
2	2	16	G	30	h	44	&
3	3	17	H	31	!	45	[
4	4	18	I	32	@	46]
5	5	19	L	33	-	47	;
6	6	20	n	34	u	48	,
7	7	21	o	35		49	:
8	8	22	P	36	_	50	.
9	9	23	Q	37	=	51	8.
10	a	24	r	38	\$		
11	b	25	t	39	*		
12	c	26	U	40	^		
13	d	27	Y	41	%		



Note: the bit 7 (corresponding to the -128 decimal value) of the *dis0÷31* paramters can be used with each character to also display the decimal point. If we define the following constant in units of configuration:

CHAR_POINT &H80 ; bit to enable decimal point

you will be able to print dotted decimal strings easily. Ad esempio per stampare: „Pnr“

dvHMI:dis6 = CHAR_P

dvHMI:dis5 = CHAR_R ORB CHAR_POINT

dvHMI:dis4 = CHAR_N

dvHMI:dis3 = CHAR_R ORB CHAR_POINT

1.3 Parameters table

Name	Dimension	Default value	Access type	Unit of measure	Valid range	Write conditions	Description
key	Long	-	R	-	-	-	State of the keys Is a parameter to a bit field each representing the state of the modifier keys. Bit binding to keys refer to the appendices.
leds	Long	0	RW	-	-	-	State of the leds Is a parameter to a bit field each representing the state of the leds. By changing the value for each led can be turned on and off; for the association of the bits to leds refer to the appendices.
blinkleds	Long	0	RW	-	-	-	State of the leds Represents the blink status at any time in the keyboard led. This variable can also be changed from the device when actions are executed in the keyboard. Bit mapping reflects that of the parameter leds. Because an led flashes it must be activated by the appropriate bit of variable leds.

numdis	Byte	-	R	-	-	-	Number of the available displays Represents the number of the available displays. Depending on the hardware platform (D221 or D231 etc) on which the device will be used, this parameter will take on different values. For example on D231 will have the value 11.
dis0÷31	Byte	-	RW	-	-	-	Character viewing Represents the current contents of the display font in 0...31 positions. dis0 is the rightmost display and dis31 is the leftmost.
blinkchar	Long	0	RW	-	-	-	Flashing character It's a bit variable to enable the blink on a character. Each bit is a character. The least significant bit is associated with the rightmost display.
ScreenA	Long	0	RW	-	-	-	Value for recursive view It's a value to be recursively showed if enabled.
ScreenB	Long	0	RW	-	-	-	Value for recursive view It's a value to be recursively showed if enabled.
ScreenC	Long	0	RW	-	-	-	Value for recursive view It's a value to be recursively showed if enabled.
decptA	Byte	0	RW	-	0÷5	-	Decimal point for recursive view It is the position of the decimal point in the recursive view A.
decptB	Byte	0	RW	-	0÷5	-	Decimal point for recursive view It is the position of the decimal point in the recursive view B.
decptC	Byte	0	RW	-	0÷5	-	Decimal point for recursive view It is the position of the decimal point in the recursive view C.
ncharA	Byte	0	Read - Write	-	1÷7	-	Number of characters to display recursive It is the number of characters that make up the recursive view A.
ncharB	Byte	0	RW	-	1÷7	-	Number of characters to display recursive It is the number of characters that make up the recursive view B.
ncharC	Byte	0	RW	-	1÷7	-	Number of characters to display recursive It is the number of characters that make up the recursive view C.
offsA	Byte	0	RW	-	0÷numdis-1	-	Recursive view location It's location right display of the recursive view A. Valid range: 0 ÷ numdis-1.
offsB	Byte	0	RW	-	0÷numdis-1	-	Recursive view location It's location right display of the recursive view B. Valid range: 0 ÷ numdis-1.
offsC	Byte	0	RW	-	0÷numdis-1	-	Recursive view location It's location right display of the recursive view C. Valid range: 0 ÷ numdis-1.

scflags	Word	0	RW	-	-	-	Configuration for recursive views Is a value to bit fields to configure the recursive screenA, screenB and screenC views.		
							Bit	Value	Description
							0	0	Recursive view screenA disabled
								1	Recursive view screenA enable
							1	0	Recursive view screenB disabled
								1	Recursive view screenB enable
							2	0	Recursive view screenC disabled
								1	Recursive view screenC enable
							3	0	Recursive view sign screenA disabled
								1	Recursive view sign screenA disabled
							4	0	Recursive view sign screenB disabled
								1	Recursive view sign screenB disabled
							5	0	Recursive view sign screenC disabled
								1	Recursive view sign screenC disabled
							6	0	Leading Zero Blank screenA disabled
1	Leading Zero Blank screenA enabled								
7	0	Leading Zero Blank screenB disabled							
	1	Leading Zero Blank screenB enabled							
8	0	Leading Zero Blank screenC disabled							
	1	Leading Zero Blank screenC enabled							
devalue	Long	0	RW	-	-	-	Data entry value Data value in data entry. Is updated with the input value upon exit from the data entry.		
denchar	Byte	0	RW	-	1÷7	-	Number of characters entered in data entry Sets the number of characters entered for data entry.		
deoffs	Byte	0	RW	-	0÷numdis-1	-	Data entry position on the display Sets the position on the display, compared to the rightmost digit, of the data entry.		
dedecpt	Byte	0	RW	-	0÷5	-	Number of decimal digital for data entry Indicates the number of decimal places for data entry.		
deuplim	Long	0	RW	-	-	-	Maximum data value introduced with data entry Is the maximum value that can be given with data entry senza activate the flag signaling of exceeding the upper limit.		
delowlim	Long	0	RW	-	-	-	Minimum value of the data introduced with data entry Is the minimum value that can be take with the data entry without active lower limit signal flag.		

deflags	Word	0	RW	-	-	-	Configuration for data entry It's a value to bit fields to configure the Data Entry.		
							Bit	Value	Description
							0	0	View only
								1	Introduction data
							1	0	Leading Zero Blank not active (valid only for viewing then with bit 0 = 0)
								1	Leading Zero Blank active (valid only for viewing then with bit 0 = 0)
							2	0	Introduction/viewing sign enable
								1	Introduction/viewing sign disable
							3	0	Standard Data Entry
								1	Data entry with +/- key as to increase decrease
4	0	Control limits disabled							
	1	Control limits enabled							
deExKeymask	Long	-	RW	-	-	-	Output configuration buttons to data entry Enable mask of the exit key to the data entry. Is a bitmask that enables one or more keys to exit the data entry. The correspondence of the key bits is equal of the key parameter. Regardless of the value set in this parameter the ENTER key will exit to the data entry.		
deExitKey	Long	-	R	-	-	-	Output viewing key to the data entry Is a value to a bit field indicating the key that brought out to the data entry. The correspondence of the bit with key is the equal of the key parameter. If the value is 0 means you are signed out of data entry with <i>EXITDE</i> and <i>EXITDEC</i> commands.		
errcode	Byte	0	RW	-	0÷100	-	Error identification code When the HMI2 device cannot execute operations are caused by incorrect user programming reports this condition through activation of the <i>st_error</i> state. The HMI2 device, makes available, through the values of the <i>errcode</i> and <i>ervalue</i> variables, some information for a better understanding of the type of error and what condition the generated. This information and the <i>st_error</i> error status, continue until you run the appropriate <i>RSERR</i> command that deletes them. The following table specifies the values from the <i>errcode</i> variable:		
							Code	Description	
							0	No errors	
							1	Reports an error in the parameterization of recursive view A.	
							2	Reports an error in the parameterization of recursive view B.	
							3	Reports an error in the parameterization of recursive view C.	
							4	Reports an error in the parameters of the data entry.	

errvalue	Byte	0	RW	-	0÷100	-	Identifying code of cause of the error Through the <i>errvalue</i> variable you can get more detailed information about the error. The following table specifies the values from the <i>errvalue</i> variable:	
							Code	Description
							0	No details about the error
							1	Signals that the viewing comes up on the display. Check the offset of the view and the number of characters.
							2	Note that the decimal point position specified is incorrect. For example, you cannot set the decimal point position equal to the number of characters.
							3	Indicates that the <i>uplim</i> parameter (upper data entry limit) is less than <i>lowlim</i> (lower data entry limit). The error does not occur if the limit checking is disabled.
							4	Reports an invalid value data for dataentry confirmed with the <i>EXITDEC</i> command.
							5	Report a collision between two recursive view or between one recursive view and the data entry. For collision refers to the use of a display area by multiple objects (recursive views or data entry).
6	Indicates that the sign is not valid. Occurs when you set a number 1 characters and the sign is enabled. Increase the number of the characters (minimum 2) or disable the sign.							
wrncode	Byte	0	RW	-	0÷100	-	Identification code warning When the parameters of the HMI2 device are programmed with out-of-range values they are not accepted and maintain their previous value. The device reports this condition through activation of the <i>st_warning</i> state. The HMI2 device, makes available, through the values of the <i>wrncode</i> and <i>wrnvalue</i> variables, some information to better understand the type of warning and what condition the generated. This information and the <i>st_warning</i> error status, continue until you execute the appropriate <i>RSWRN</i> command that will remove them. The following table specifies the values from the variable <i>wrncode</i> :	
							Code	Description
							0	No warning
							1	Report a warning in the parameterization of recursive view A.
							2	Report a warning in the parameterization of recursive view B.
							3	Report a warning in the parameterization of recursive view C.
							4	Report a warning in the data entry parameterization.
							5	Report a warning had to request of the <i>EXITDE</i> or <i>EXITDEC</i> command execution without that data entry is active (state <i>st_dentry</i> =0)
6	Report a warning had to request of the <i>DATAENTRY</i> command execution with the data entry already active (state <i>st_dentry</i> =1).							

wrnvalue	Byte	0	RW	-	0÷100	-	Identification code of the cause of warning Through the <i>wrnvalue</i> variable you can get more detailed informations about the error. The following table specifies the values from the <i>wrnvalue</i> variable:	
							Code	Description
							0	No details about the error
							1	Reports that the decimal point position specified is out of range.
							2	Reports that the number of specified characters is out of range.
							3	Reports that the specified offset is out of range (greater than the number of display)
4	Report a collision between two recursive views or between recursive and data entry. For collision refers to the use of a display area by multiple objects (recursive views or data entry).							

1.4 States table

Name	Default value	Description
st_modified	0	Data entry modified Reports that the data entry data has been changed.
st_dentry	0	Stato data entry Reports that the device is in data entry.
st_uplim	0	Upper limit exceeded state Reports that the introduced data is greater than the value set in the <i>deuplim</i> parameter. Cleared by the <i>DATAENTRY</i> command and updated upon exiting the data entry with confirmation of the data.
st_lowlim	0	Lower limit exceeded state Reports that the introduced data is less than the value set in <i>delowlim</i> parameter. Cleared by the <i>DATAENTRY</i> command and updated upon exiting the data entry with confirmation of the data.
st_exitcmd	0	Exit status with command EXITDE Reports the output from data entry with <i>EXITDE</i> command. Cleared by the <i>DATAENTRY</i> command and updated upon exiting the data entry.
st_error	0	There is an error Reports the error status of the device, to recognize the type of error you must refer to <i>errcode</i> and <i>errvalue</i> variables: 0: error not present, 1: error present
st_warning	0	Warning presence Reports the warning state of the device, to recognize the type of warning you must refer to <i>wrncode</i> and <i>wrnvalue</i> variables: 0: warning not present, 1: warning present

1.5 Commands table

Name	Condition	Description
DATAENTRY	st_dentry=0	Enters in the data entry procedure Allows you to enter the data entry procedure.
EXITDE	st_dentry=1	Leave from the data entry procedure Allows you to exit the data entry procedure.
EXITDEC	st_dentry=1	Leave the data entry procedure after confirmation Allows you to exit the data entry procedure with the confirmation data in introduction (how if we had pressed the button ENTER).
RSERR	st_error=1	Reset error state Reset the <i>st_error</i> state.
RSWRN	st_warning=1	Reset warning state Reset the <i>st_warning</i> state.

Documento generato automaticamente da **Qem Wiki** - <https://wiki.qem.it/>

Il contenuto wiki è costantemente aggiornato dal team di sviluppo, è quindi possibile che la versione online contenga informazioni più recenti di questo documento.