BUCHER automation

Application-oriented manual

60887613_01

Control of alphanumeric HMIs (LCD) and printers

Bucher Automation AG has created this document with the requisite care and based on the current state of technology. Changes and further technical developments to our products are not automatically made available in a revised document. Bucher Automation AG shall accept no liability or responsibility for errors of content or form, missing updates or any damage or disadvantages arising therefrom.

Bucher Automation AG
Thomas-Alva-Edison-Ring 10
71672 Marbach/Neckar, Germany
T +49 7141 2550-0
info@bucherautomation.com

Technical support
T +49 7141 2550-444
support@bucherautomation.com

Sales
T +49 7141 2550-663
sales@bucherautomation.com

www.bucherautomation.com

Translation of the german original User Manual

Revision	2.00
Date of issue	5/16/2024

Table of contents

1 Introduction

This application-oriented manual describes how to control HMIs with text display, and printing and serial interfaces from within the application program of a JC-3xx or JC-4xx control system.

1.1 Typographical conventions

This manual uses different typographical effects to support you in finding and classifying information. Below, there is an example of a step-by-step instruction:

✓ This symbol indicates requirements which have to be met before executing the following action.

► This sign or a numbering at the beginning of a paragraph marks an action instruction that must be executed by the user. Execute the instructions one after the other.

⇨ The target after a list of instructions indicates reactions to, or results of these actions.

ⓘ Info

Further information and practical tips

In the info box you will find helpful information and practical tips about your product.

2 Connecting alphanumeric HMIs

This chapter describes how to connect a Bucher Automation AG alphanumeric HMI to a JC-3xx or JC-4xx controller.

2.1 Summery of HMIs

The following table gives a summary of the range of Bucher Automation AG alphanumeric HMIs designed to connect to the controller.

Designation	Display	Keys	Interface cable
LCD 16	4 lines, 20 characters each	– 5 function keys with LED – Optionally expandable with keyboard module NUM25	Cable assembly no. 192 xM
LCD 23	2 lines, 24 characters each	– Cursor left – Cursor right – ENTER ([↲])	Cable assembly no. 192 xM
LCD 25	2 lines, 24 characters each	– 5 function keys with LED	Cable assembly no. 192 xM
LCD 27	2 lines, 24 characters each	– Function keys – Cursor keypad – Clear – ENTER ([↲])	Cable assembly no. 192 xM
LCD 34	2 lines, 24 characters each	– 5 function keys – Numeric keypad	Cable assembly no. 192 xM
LCD 52	4 lines, 16 characters each	– 6 function keys – Numeric keypad	KAY-0533-0025
LCD 54	4 lines, 16 characters each	– 8 function keys – Numeric keypad – Emergency stop	KAY-0533-0025
LCD 54Z	4 lines, 16 characters each	– 8 function keys – Numeric keypad – Emergency stop – Two-hand control device	KAY-0533-0025
LCD 60	2 lines, 40 characters each	– 8 function keys with LED – Numeric keypad	KAY-0386-xxxx
LCD 110	4 lines, 20 characters each	– 8 function keys with LED – Numeric keypad	Cable assembly no. 192 xM

Tab. 1: Summery of HMIs

Info

LCD 110 connection

Connecting an LCD 110 to a JC-4xx controller is possible provided that the display's voltage supply (DC 24 V) is **not** taken off the controller.

This requires an external voltage source; for details see Verdrahtung im Multi-Display-Modus [▶ 9].

2.2 Connecting an HMI

Prefabricated cable assemblies are available to connect a Bucher Automation AG alphanumeric HMI to a controller. We recommend that the prefabricated cable assemblies listed in the above table be used.

Interface

Connect the HMI to the X11 socket via serial interface. using RS-422 interface standard connections.

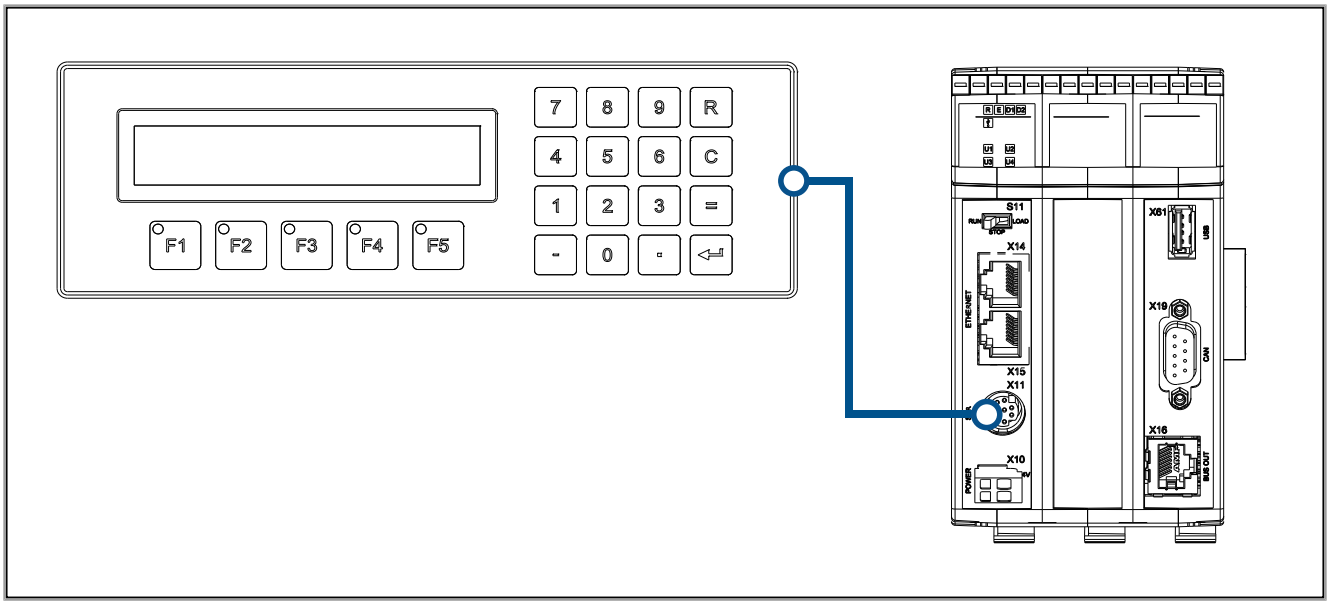


Fig. 1: Connecting HMI and controller

Info

Limitations

– While the device supports several different hardware drivers, only one hardware interface is available. This means: If communication takes place for example via RS-422, the system cannot communicate simultaneous and independently via RS-232.

– The maximum cable length is < 30 m.

2.3 Connecting several HMIs: Multi-display mode

Multi-display mode allows for up to 4 alphanumeric HMIs to connect to a controller through one serial interface. The connected HMIs can be set up to display identical or different text strings and/or variable content.

Interface

Connect the HMI to the X11 socket via serial interface. using RS-422 interface standard connections.

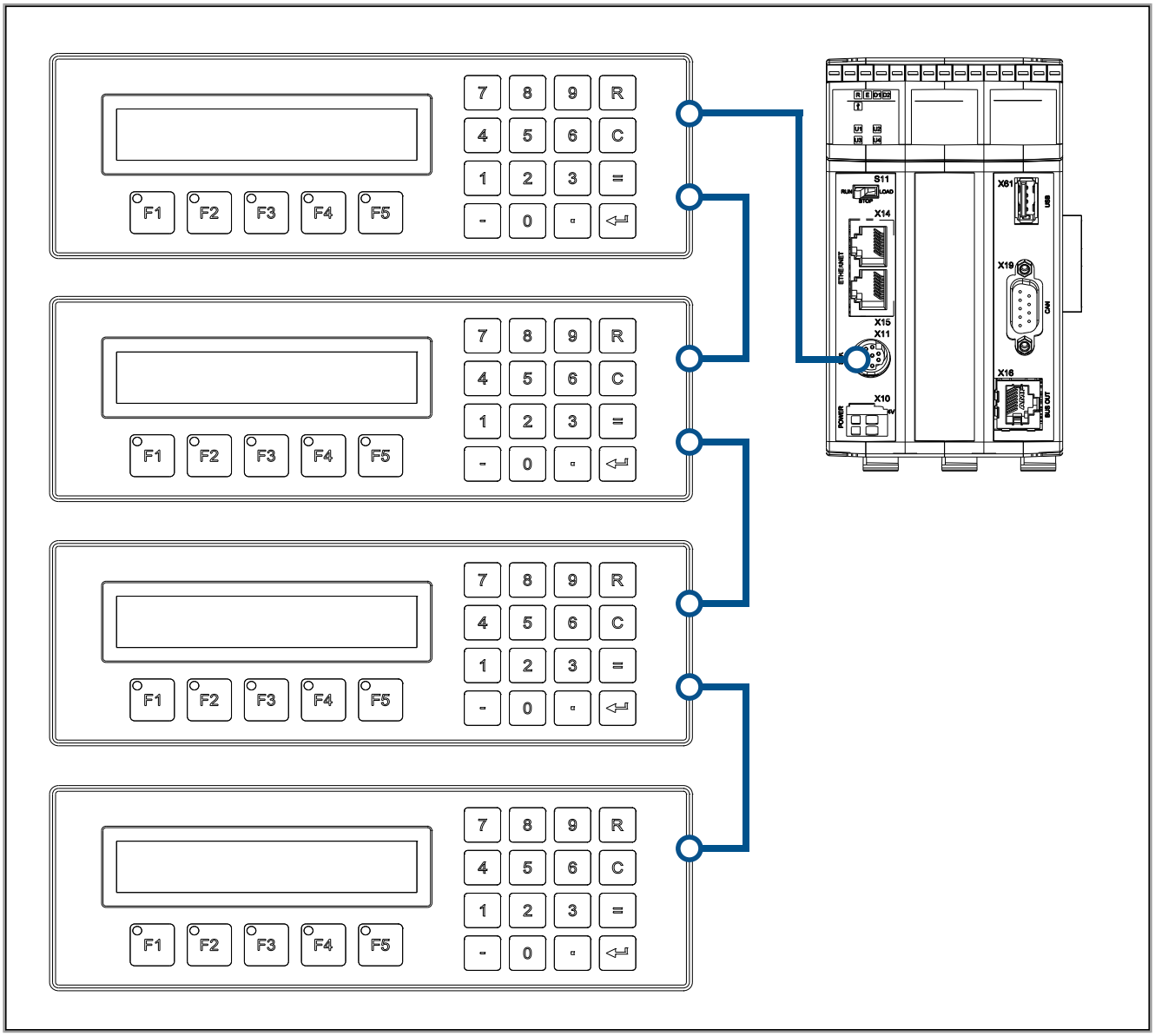


Fig. 2: Connecting several HMIs to the controller

Info

Limitations

- While the device supports several different hardware drivers, only one hardware interface is available. This means: If communication takes place for example via RS-422, the system cannot communicate simultaneous and independently via RS-232.
- The maximum cable length is < 30 m.

2.3.1 Multi-display mode wiring

There are no prefabricated cables for connecting several HMIs to a controller.
Assemble the cables for multi-display mode according to the following block diagram:

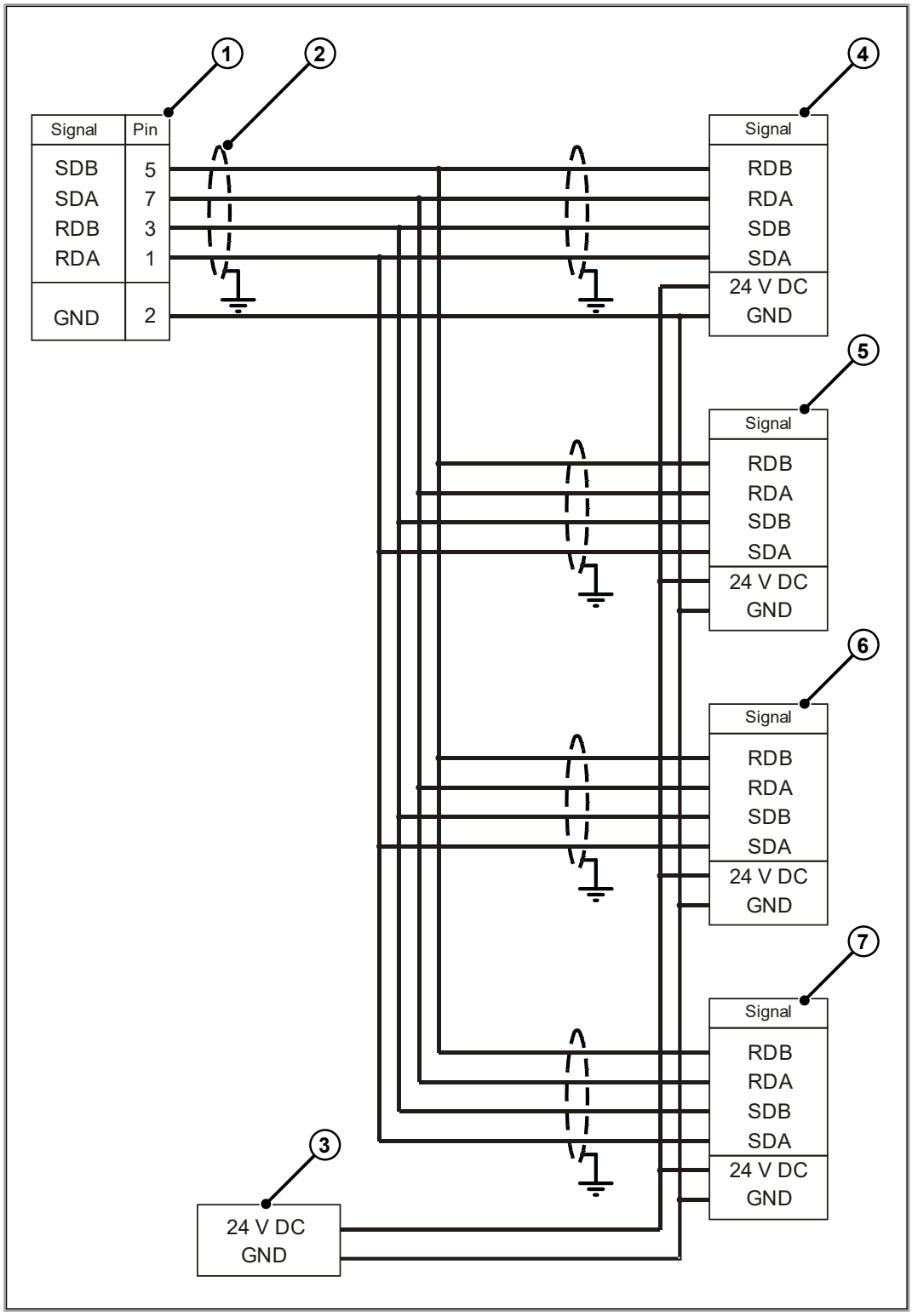


Fig. 3: Multi-display mode wiring

Number	Part	Description
1	Terminal X11	Serial interface of the controller
2	Shield	Use shielded cables both ends of which are connected to the metallized housing of the connector.
3	Power supply	In a multi-display setup, each HMI requires individual power supply.
4 ... 7	Terminals	HMI interfaces

Cable specification

The following minimum requirements apply to cable sets:

Parameter	Description
Core cross-sectional area	0.14 mm^2
Maximum cable length	< 30 m
Shield	Complete, not paired

2.4 Interface cable assembly no. 192 xM

The interface cable assembly 192 xM connects the following types of HMIs to the controller:

– LCD 16
– LCD 23
– LCD 25
– LCD 27
– LCD 34
– LCD 110

Male connector specification (controller end)

Parameter	Description
Type	8-pin male MiniDIN connector
Manufacturer	KYCON
Item	KMDLA - 8P
Recommended core cross section	0.051 mm^2 ... 0.128 mm^2

Male connector specification (HMI end)

Parameter	Description
Type	15-pin male SUB-D connector in metallized housing (quality grade 3)
Manufacturer	Various
Recommended core cross section	0.128 mm^2 ... 0.25 mm^2

Specifications of connecting cable

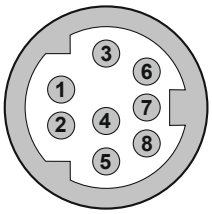
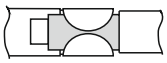
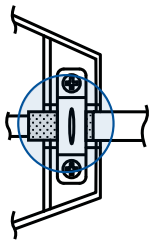
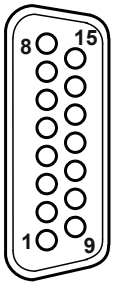
Parameter	Description
Number of cores	6
Recommended core cross section	0.14 mm^2
Maximum cable length	< 30 m

Cable shielding

- Complete, not paired
- The shield must connect to the connector enclosures on both ends of the cable with the greatest possible surface area. Place the shield as follows:
  - Bundle the shielding.
  - Clamp it under a strain relief.
  - Wrap it with copper foil.

Wiring diagram

The interface cable assembly no. 192 xM is wired as follows:

Controller	Shield		HMI
SER/X11 			
	Connect shield with the greatest possible surface area Requires metallized housing		
Pin	**Signal**		**Pin**
6	DC +24 V		15
2	GND		12
5	SDB	RDB	6
7	SDA	RDA	7
3	RDB	SDB	4
1	RDA	SDA	5

Available lengths

Available cable length options of prefabricated interface cable assembly no. 192 xM:

Component	Description	Item number
Cable assy no. 192 2.5M	From controller to HMI 15-pin Sub-D, length 2.5 m	60860011
Cable assy no. 193 5M	From controller to HMI 15-pin Sub-D, length 5 m	60860012
Cable assy no. 192 10M	From controller to HMI 15-pin Sub-D, length 10 m	60872142
Cable assy no. 192 15M	From controller to HMI 15-pin Sub-D, length 15 m	60872884

2.5 Interface cable KAY_0386-xxxx

The interface cable assembly KAY_0386-xxxx connects the LCD 60 HMI model to the controller.

Male connector specification (controller end)

Parameter	Description
Type	8-pin male MiniDIN connector
Manufacturer	KYCON
Item	KMDLA - 8P
Recommended core cross section	0.051 mm^2 ... 0.128 mm^2

Female connector specification (HMI end)

Parameter	Description
Type	15-pin female Sub-D connector in metallized housing (quality grade 3)
Manufacturer	Various manufacturers
Recommended core cross section	0.128 mm^2 ... 0.25 mm^2

Specifications of connecting cable

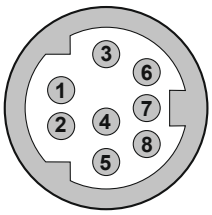
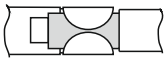
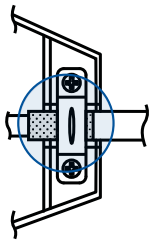
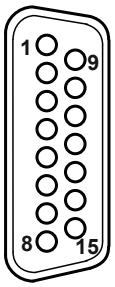
Category	Description
Number of cores	5
Recommended core cross section	0.14 mm^2
Maximum cable length	< 30 m

Cable shielding

– Complete, not paired
– The shield must connect to the connector enclosures on both ends of the cable with the greatest possible surface area. Place the shield as follows:
  – Bundle the shielding.
  – Clamp it under a strain relief.
  – Wrap it with copper foil.

Wiring diagram

The interface cable KAY_0386-xxxx is wired as follows:

Controller	Shield		LCD 60
SER/X11 			**COM 2** 
	Connect shield with the greatest possible surface area Requires metallized housing		
Pin	**Signal**		**Pin**
2	GND		5
5	SDB	RDB	13
7	SDA	RDA	12
3	RDB	SDB	15
1	RDA	SDA	14
		Short-circuited	11
			10

Available lengths

Available cable length options of prefabricated interface cable assembly KAY_0386-xxx:

Component	Description	Item number
KAY_0386-0250	JC-xxx to LCD 60 with 15-pin Sub-D, length 2.5 m	60864359
KAY_0386-0500	JC-xxx to LCD 60 with 15-pin Sub-D, length 5 m	60864360

2.6 Interface cable KAY_0533-0025

Interface cable assembly KAY_0533-0025 connects HMI models LCD 52, LCD 54 and LCD-54Z to the controller.

Male connector specification (controller end)

Parameter	Description
Type	8-pin male MiniDIN connector
Manufacturer	KYCON
Item	KMDLA - 8P
Recommended core cross section	0.051 mm^2 ... 0.128 mm^2

Female connector specification (HMI end)

Parameter	Description
Type	15-pin female Sub-D connector in metallized housing (quality grade 3)
Manufacturer	Various manufacturers
Recommended core cross section	0.128 mm^2 ... 0.25 mm^2

Specifications of connecting cable

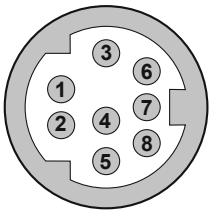
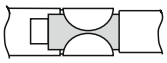
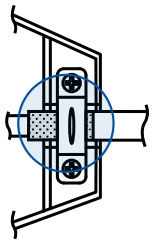
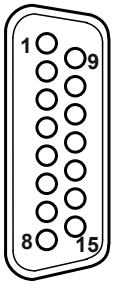
Category	Description
Number of cores	6
Recommended core cross section	0.14 mm^2
Cable length	0.25 m

Cable shielding

- Complete, not paired
- The shield must connect to the connector enclosures on both ends of the cable with the greatest possible surface area. Place the shield as follows:
  - Bundle the shielding.
  - Clamp it under a strain relief.
  - Wrap it with copper foil.

Wiring diagram

The interface cable KAY_0533-0025 is wired as follows:

Controller	Shield		LCD 52/54/54Z
SER/X11 			
	Connect shield with the greatest possible surface area! A metallized housing is required!		
Pin	**Signal**		**Pin**
6	DC +24 V		4
2	GND		7
5	SDB	RDB	10
7	SDA	RDA	11
3	RDB	SDB	12
1	RDA	SDA	13

Available lengths

Available cable length options of prefabricated interface cable assembly KAY_0533-0025:

Component	Description	Item number
KAY_0533-0025	JC-xxx to LCD 52/54 with 15-pin Sub-D, length 0.25 m	60864897

3 Control of alphanumeric HMIs

This chapter describes how to control Bucher Automation AG HMIs with text display from within the application program of a JC-3xx or JC-4xx control system. It also describes the registers used to parameterize the display functions. The controller provides the following display options:

– Displaying text strings
– Displaying the contents of variables
– Scanning the HMI keys
– Switching the HMI LEDs
– Monitor function

The display feature is part of the STX language set.

ⓘ Info

Further information

For a detailed description of the STX commands mentioned in this chapter, see the JetSym Help at *JetSym ST/STX projects* > *JetSym STX programming language* > *Display commands and user input*.

3.1 Registers

ⓘ Info

Limitations

The settings made in the given registers take global effect, meaning they impact all functions for controlling HMIs. If different settings are used in several tasks of the application program, then these settings may impact each other.

The registers are clustered into one register block. The basic register number of this block depends on the controller.

Register numbers

Basic register number	Register numbers
220000	222804 ... 222840

In this chapter, only the last 4 digits of a register number are specified, e.g. MR 2838. This is the module register. Add the basic register number of the corresponding device to determine the complete register number, for example 222838.

3.2 Configuring the screen size

During the boot process the HMI logs in to the controller and transmits its display size. This way, the controller can configure the size automatically. Therefore, there is no need configure the display size manually. But in some cases, this feature might make sense.

Correct configuration of the display size is required to ensure the controller correctly executes the special functions *Delete Screen*, and *Delete to End of Line* when displaying text strings (DisplayText commands).

Configuring the screen size manually

To configure the screen size manually, proceed as follows:

1. Enter the number of characters per line into MR 2805.
2. Multiply the value contained in MR 2805 by the number of lines and enter the result into MR 2804.

3.2.1 Register description

MR 2804

Number of characters on the screen

This module register contains the number of characters displayed on the screen.

Property	Description
Values	1 ... 128
Value after reset	48

Tab. 2: Number of characters on the screen

MR 2805

Number of characters per line

This module register contains the number of characters per line.

Property	Description
Values	1 ... 128
Value after reset	24

Tab. 3: Number of characters per line

3.3 Device numbers

The device number defines the output device. The device number is used as the **<DeviceNo>** parameter with the `DisplayText()`, `DisplayText2()`, `DisplayValue()` and `UserInput()` commands (see JetSym Help).

Overview

You may enter the following values for the parameter device number:

Number	Part	Description
0	Default device	The device number to be used is contained in MR 2824
1	HMI 1	Multi-display mode
2	HMI 2	Single-/Multi-display mode
3	HMI 3	Multi-display mode
4	HMI 4	Multi-display mode
5 ... 7	Reserved	Do not use
8	Printer module	Output to a printer module on the JX2 system bus
9, 10	Serial interface	Output to the user-programmable serial interface of the controller
11	Serial interface module	Output to a serial interface module on theJX2 or JX3 system bus

Tab. 4: Device number summary

Single-display mode

In single-display mode, an HMI always displays data from device no. **2**.

Multi-display mode

In multi-display mode, an HMI always displays data from the device the number of which is contained in the corresponding configuration register MR 2825 through MR 2828.

3.3.1 Register description

MR 2824

Device number of the default device

This module register contains the device number of the default device. If you always specify the default device in the application program (device number = 0), you can select during runtime which device is to be used.

Property	Description
Values	1 ... 11
Value after reset	2

Tab. 5: Device number of the default device

MR 2825

Device number of HMI 1 in multi-display mode

This module register contains the device number of HMI 1 in multi-display mode.

Property	Description
Values	1 ... 4
Value after reset	1

Tab. 6: Device number of HMI 1 in multi-display mode

MR 2826

Device number of HMI 2 in multi-display mode

This module register contains the device number of HMI 2 in multi-display mode.

Property	Description
Values	1 ... 4
Value after reset	2

Tab. 7: Device number of HMI 2 in multi-display mode

MR 2827

Device number of HMI 3 in multi-display mode

This module register contains the device number of HMI 3 in multi-display mode.

Property	Description
Values	1 ... 4
Value after reset	3

Tab. 8: Device number of HMI 3 in multi-display mode

MR 2828

Device number of HMI 4 in multi-display mode

This module register contains the device number of HMI 4 in multi-display mode.

Property	Description
Values	1 ... 4
Value after reset	4

Tab. 9: Device number of HMI 4 in multi-display mode

3.4 Cursor position

This parameter defines the position on the screen of the first character of a text string or variable to be displayed.

Info

Limitations

There is no cursor position evaluation for devices 8 through 11.

Cursor position 0

Cursor position **0** has a special meaning. The controller evaluates this parameter performing the following steps:

Step	Description	
1	The controller reads the content of MR 2814.	
	If...	**... then...**
	... MR 2814 is greater than 0,	... the value is used as cursor position.
	... MR 2814 equals 0,	... the message is displayed from the current cursor position onwards.

Tab. 10: Cursor position 0

Example: MR 2814 = 0

```
DisplayText(0, 1, 'Temp :');
DisplayValue(0, 0, Temperature);
```

Result: The temperature is displayed directly after the colon from cursor position 7 onwards.

LCD cursor positions

The following table shows the correlation between the cursor position as a command parameter and the position on the screen:

Designation	Line	Cursor position
LCD 16, LCD 110	1	1 ... 20
	2	21 ... 40
	3	41 ... 60
	4	61 ... 80
LCD 23, LCD 25, LCD 27, LCD 34	1	1 ... 24
	2	25 ... 48
LCD 52, LCD 54(Z)	1	1 ... 16
	2	17 ... 32
	3	33 ... 48
	4	49 ... 64

Designation	Line	Cursor position
LCD 60	1	1 ... 40
	2	41 ... 80

3.4.1 Register description

MR 2814

Indirect cursor position

If **0** was programmed as command parameter this module register specifies the cursor position.

Property	Description
Values	0 ... number of characters on the screen
Value after reset	0

Tab. 11: Indirect cursor position

3.5 Displaying text strings

STX commands

To display text strings, use the following STX commands (STX functions):

- `DisplayText()`
- `DisplayText2()`

3.5.1 DisplayText() STX command

Declaration

```
Function DisplayText(Dev:Int,
                     Pos:Int,
                     Const Ref Text:String);
```

Parameter

Parameter	Value	Description
Dev	0 ... 11	Number of the HMI where the text string is to be output
Pos	1 ... maximum number of characters shown on the screen	Cursor position defining the starting point for a text string to be displayed
Text	Text string to be displayed	Hard-coded text, or name of a string variable

Tab. 12: `DisplayText()` parameter

How to use this command

Command-line syntax for displaying a text string:

```
DisplayText(0,1, "Hello World!");
DisplayText(0, 25, StringVar);
```

Operating principle

The first STX command clears the entire screen content ('_' in text string). Then, it displays the text string 'Hello World!' from cursor position 1 onwards. The second STX command outputs the content of the string variable **StringVar** from cursor position 25 onwards. Both text strings are displayed on the default device (Dev = 0).

3.5.2 DisplayText2() STX command

Declaration

```
Function DisplayText2(Dev:Int,
                      Pos:Int,
                      Const Ref Text1:String)
                      Const Ref Text2:String);
```

Parameter

Parameter	Value	Description
Dev	0 ... 4	Number of the HMI where the text string is to be output
Pos	1 ... number of characters on the screen	Cursor position defining the starting point for a text string to be displayed
Text1	Text string to be displayed	Hard-coded text, or name of a string variable
Text2	Text string to be displayed	Hard-coded text, or name of a string variable

Tab. 13: `DisplayText2()` parameter

How to use this command

Command-line syntax for displaying one of two text strings:

```
DisplayText2(0, 25, 'Fehler:', 'Error:');
```

Operating principle

Calling up the STX command makes the text strings 'Fehler:' or 'Error:' appear on the default device (Dev = 0) from cursor position 25 onwards. MR 2806 controls which text string will be displayed.

3.5.3 Clearing the screen

If text strings are displayed, there are two control characters allowing to clear the screen content:

– Clearing the entire screen content
– Clearing the text string to the end of the line

ⓘ Info

Limitations

If display commands are addressed to devices 8 ... 11, these characters are not considered as control characters but displayed as text.

Clearing the screen

The default character for clearing the screen is the underline character '_'.

Using this character first deletes the displayed text strings, and then outputs any text following the underline character from cursor position 1 onwards.

Example

```
DisplayText(0, 10, 'H_ello');
```

Result: The screen is cleared and the word fragment "ello" appears from cursor position 1 onwards.

Delete text to the end of the line

The default character for deleting text up to the end of line is the dollar character '$'.

Using this character deletes the rest of the line from the present cursor position onwards.

Example

```
DisplayText(0, 25, 'Position :$');
```

Result: "Position :" is displayed from cursor position 25 onwards, and the rest of the line is deleted.

Changing control characters

To make the underline "_" and dollar "$" characters appear as regular characters, you must change the corresponding control character. Define the control characters in module registers MR 2839 and MR 2840.

3.5.4 Register description

MR 2806

Text selection for DisplayText2

The value in this module register specifies which one of the two text strings is to be output.

Property	Description	
Values	0	Text1
	1	Text2

Tab. 14: Text selection for DisplayText2

MR 2839

Control character for clearing the screen content

This module register contains the ASCII code of the control character clearing the screen content.

Property	Description
Values	0 ... 255
Value after reset	95 ('_')
Takes effect	Upon the next execution of the STX 'DisplayText()' or 'DisplayText2()' command

Tab. 15: Control character for clearing the screen content

MR 2840

Control character for deleting a text string up to the end of a line

This module register contains the ASCII code of the control character deleting a text string up to the end of a line.

Property	Description
Values	0 ... 255
Value after reset	36 ('$')
Takes effect	Upon the next execution of the STX ‘DisplayText()’ or ‘DisplayText2()’ command

Tab. 16: Control character for deleting a text string up to the end of a line

3.6 Displaying numerical values

STX command

To display numerical values, use the following STX command:

– `DisplayValue()`

Screen settings

The format of numerical values can be adjusted to individual needs. To do so, use the following parameters:

– Display field length
– Number of decimal places
– With or without sign place
– Displaying values in decimal or hexadecimal notation

Displaying numerical values

When displaying numerical values, the following formatting applies:

– The numerical value is displayed right-aligned.
– The first character to appear in the display field is the sign unless writing the sign character was disabled beforehand.
– Positive numericals are prefixed by a space character as sign. Negative numericals are prefixed by minus ‘-’.
– If the display field is too small, the leftmost digits are truncated.
– The value is rounded to the set decimal places.

3.6.1 DisplayValue() STX command

Declaration

```
Function DisplayValue(Dev:Int,
                      Pos:Int,
                      Value:Double);
```

Parameter

Parameter	Value	Description
Dev	0 ... 4	Number of the device where to output the value
Pos	1 ... maximum number of characters shown on the screen	Cursor position starting from where the value is to be displayed
Value	Value to be displayed	Constant value, name of a register or a variable

Tab. 17: `DisplayValue()` parameter

How to use this command

Command-line syntax for displaying a value:

```
DisplayValue(8, 0, -12,345);
DisplayValue(8, 0, Axis2.Position);
```

Operating principle

The first STX command outputs the value **-12,345** from cursor position 1 onwards. The second STX command outputs the content of variable **Axis2.Position** from cursor position 25 onwards.

Both numericals are output to the default device (Dev = 0).

3.6.2 Configuring the screen size

Setting the length of the display field

MR 2812 defines the length of the display field for numerical values.

```
MR 2812 = Number of digits + sign [+ decimal point]
```

Example

Parameter	Description
Number of digits	6
Show sign (MR 2816)	0 (Yes)
Decimal point	No
Field length (MR 2812)	7
Display	7 characters

Tab. 18: Setting the display field length – example

Setting the sign option

MR 2816 defines whether or not the sign is displayed.

To set the display field length in MR 2812 correctly, include the sign's place into the calculation, even if no sign is to be displayed.

Example

Parameter	Description
Number of digits	6
Show sign (MR 2816)	1 (No)
Decimal point	No
Field length (MR 2812)	7
Display	6 characters

Tab. 19: Setting the sign option – example

Setting the number of decimal places

MR 2810 defines the number of decimal places.

Displaying decimal places might require adjustment of the display field length in MR 2812.
The reason: The decimal point uses one place in the display field.

Setting the numerical value format

Numericals can be displayed in either decimal or hexadecimal format.

Flag 2060 sets the format of numerical values.

3.6.3 Register description

MR 2810

Number of decimal places for DisplayValue

This module register contains the number of decimal places to be displayed for numerical values.

Property	Description
Values	0 ... 4
Takes effect	Upon next execution of STX `DisplayValue()` command

Tab. 20: Number of decimal places for DisplayValue

MR 2812

Field length for DisplayValue

This module register contains the length of the display field.

Property	Description
Values	1 ... 12
Value after reset	11
Takes effect	Upon next execution of STX `DisplayValue()` command

Tab. 21: Field length for DisplayValue

MR 2816

Sign option for DisplayValue

The value in this module register specifies which one of two text strings is to be output.

Property	Description	
Values	0	Sign will be displayed
	1	Sign will not be displayed
Takes effect	Upon next execution of STX `DisplayValue()` command	

Tab. 22: Sign option for DisplayValue

Flag 2060

Numerical value format for DisplayValue

Flag 2060 sets the numerical format.

Property	Description	
Values	0	Decimal
	1	Hexadecimal
Takes effect	Upon next execution of STX `DisplayValue()` command	

Tab. 23: Numerical value format for DisplayValue

3.7 Entering numerical values

STX command

To enter register and variable contents via HMI, use the following STX command:

– `UserInput()`

Info

Limitations

The following limitations apply to the STX `UserInput()` command:

– While inputting numerical values, the keys used for this are not mapped to the key flags.

– The STX `UserInput()` command is not executed by the controller for as long as the monitor function is active.

Entered value format

The format of numerical values to be entered can be adjusted to suit application needs. To do so, use the following parameters:

– Input field length

– Maximum number of decimal places

– Suggest value

Keys used for entering numericals

Key	Description
[0] ... [9]	Enters a numerical value
[.] or [,]	Enters a decimal point or decimal comma
[-]	Enters a negative/positive numerical value; you can press the key any time during input.
[C]	Clears current entry; Displays suggested value again
ENTER ([↲])	Terminates the input process; confirms the input

Tab. 24: Keys used for entering numericals

Displaying input

During input, the following is displayed:

Step	Description	
1	The controller shows the suggested value. The formatting parameters are used the same way as for displaying numerical values.	
2	**If...**	**... then...**
	... You press the **ENTER** key	... the controller accepts the suggested value and completes executing the command.
	... you press another key usable for data input,	... the controller clears the suggested value and shows the value of the corresponding key instead.
3	The numerical value is left-aligned in the input field, until the command was successfully completed or aborted.	
4	**Result:** The input confirmed most recently remains on the screen.	

Tab. 25: Displaying input

3.7.1 UserInput() STX command

Declaration

```
Function UserInput(Dev:Int,
                   Pos:Int):Double;
```

Parameter

Parameter	Value	Description
Dev	0 ... 4	Number of the device where to input the value
Pos	1 ... maximum number of characters shown on the screen	Cursor position starting from where the input field is to be displayed

Tab. 26: `UserInput()` parameter

Result

Property	Description
Type	Double
Value	Value that has been input

How to use this command

Command-line syntax for assigning the return value to a variable:

```
AutoSet[Index].Destination := UserInput(0, 10);
```

Operating principle

The controller processes command as follows:

Step	Description	
1	Addressing device **0**, the controller displays the suggested value from cursor position **10** onwards, prepares an input field and activates the cursor.	
2	The task stops on the STX `UserInput()` command, until being aborted or completed by the user pressing the **ENTER** ([↵]) key.	
3	**If...**	**... then...**
	... You press the **ENTER** ([↵]) key	... the controller assigns the entered value to the variable and resumes the task executing the next command.
	... the STX `UserInput()` command is aborted,	... an exception is thrown and step 4 is carried out.
4	**If...**	**... then...**
	... an exception handling has been programmed,	... the task proceeds with the exception handling.
	... no exception handling has been programmed,	... the task is aborted and the error is displayed in the error register.

Tab. 27: `UserInput()` operating principle

Polling the state

MR 2817 polls the state of the STX `UserInput()` command.

Aborting the command

To abort an active STX `UserInput()` command, enter the value **0** in MR 2817.

Operating principle

The controller performs the following steps to abort an active STX `UserInput()` command:

Step	Description
1	The controller disables the blinking cursor on the HMI.
2	The controller throws the *USER_INPUT_BREAK* exception.

Step	Description	
3	**If...**	**... then...**
	... an exception handling has been programmed,	... the task proceeds with the exception handling.
	... no exception handling has been programmed,	... the task is aborted and the error is displayed in the error register.
4	**Result:** The variable, where function result was addressed to, will not be changed.	

Tab. 28: Operating principle of aborting the `UserInput()` command

How to use this command

```
Try
        Value := UserInput (0, 25);
    Catch USER_INPUT_BREAK:
        Trace ('UserInput aborted !!');
End_Try;
```

3.7.2 Configuring the input

Setting the length of the input field

MR 2813 defines the input field length for numerical values.

```
MR 2813 = Number of digits + sign [+ decimal point]
```

Example

Parameter	Description
Number of digits	6
Decimal point	No
Field length (MR 2813)	7

Tab. 29: Setting the display field length – example

Setting and polling the number of decimal places

MR 2811 defines the number of decimal places.

Entering decimal places might require adjustment of the display field length in MR 2813.
The reason: The decimal point uses one place in the display field.

MR 2808 contains the number of decimal places entered.

Setting the suggested value

MR 2815 defines the suggested value for the STX `UserInput()` command.

The controller displays the suggested value for the STX `UserInput()` command using the same format settings as for displaying numerical values.

3.7.3 Register description

MR 2808

Number of decimal places entered with UserInput

This module register contains the number of decimal places entered by the user.

Property	Description
Values	0 ... [MR 2811]

Tab. 30: Number of decimal places entered with UserInput

MR 2811

Setting the maximum number of decimal places for UserInput

This module register contains the maximum permissible number of decimal places for inputting numericals.

Property	Description
Values	1 ... 12
Value after reset	11
Takes effect	Upon next execution of STX `UserInput()` command

Tab. 31: Number of decimal places for UserInput

MR 2813

Field length for UserInput

This module register contains the length of the input field.

Property	Description
Values	1 ... 12
Value after reset	11
Takes effect	Upon next execution of STX `UserInput()` command

Tab. 32: Field length for UserInput

MR 2815

Suggested value for UserInput

This module register contains the suggested value to be displayed after the STX `UserInput()` command was executed, and the clear key **[C]** was pressed.

Property	Description
Values (Int)	-2,147,483,648 ... 2,147,483,647
Values (Float)	+/-(1.2×10^{-38} ... 3.4×10^{38})
Type	Int or Float depending on the value most recently written
Takes effect	Upon next execution of STX `UserInput()` command

Tab. 33: Suggested value for UserInput

MR 2817

UserInput state

This module register contains the state of the `UserInput()` command. To abort an active STX `UserInput()` command, enter the value **0** in MR 2817.

Property	Description	
Reading values	0	No UserInput active
	1	UserInput active
Writing values	0	Aborting UserInput

Tab. 34: UserInput state

3.8 Querying the keys

Mapping of keys

The controller maps the keys of the HMIs to the following variables:

– Special flags
– Bits in registers that overlay special flags

Flags and register bits can have the following states:

Key	Special flag/register bit
Pressed	TRUE/1
Not pressed	FALSE/0

ⓘ Info

Limitations

While inputting numerical values, the keys used for this are not mapped to the key flags and register bits.

Keys used for entering numericals

Key	Description
[0] ... [9]	Enters a numerical value
[.] or [,]	Enters a decimal point or decimal comma
[-]	Enters a negative/positive numerical value; you can press the key any time during input.
[C]	Clears current entry; Displays suggested value again
ENTER ([↵])	Terminates the input process; confirms the input

Tab. 35: Keys used for entering numericals

3.9 Assigning keys

HMI keys are assigned to an array of special flags and registers overlaying the flags.

Flag numbers

Key flags are addressed relative to a basic flag number. In multi-display you can set the basic flag number via registers.

HMI	Registers	Default basic flag number	Flag numbers
Single display	-	2000	2160 ... 2223
1	MR 2829	2000	2160 ... 2223
2	MR 2830	2000	2160 ... 2223
3	MR 2831	2000	2160 ... 2223
4	MR 2832	2000	2160 ... 2223

Tab. 36: Flag numbers

Assignment

Below, the assignment of keys, special flags and overlaid registers are listed. The given assignments apply in case of single-display mode and when using the default settings in multi-display mode.

Numerical keys

Key	Flag	Register bit (16-bit overlaying)	Register bit (32-bit overlaying)
[0]	2160	203139.0	203127.16
[1]	2161	203139.1	203127.17
[2]	2162	203139.2	203127.18
[3]	2163	203139.3	203127.19
[4]	2164	203139.4	203127.20
[5]	2165	203139.5	203127.21
[6]	2166	203139.6	203127.22
[7]	2167	203139.7	203127.23
[8]	2168	203139.8	203127.24
[9]	2169	203139.9	203127.25
[SHIFT]+[0]	2170	203139.10	203127.26
[SHIFT]+[1]	2171	203139.11	203127.27
[SHIFT]+[2]	2172	203139.12	203127.28
[SHIFT]+[3]	2173	203139.13	203127.29
[SHIFT]+[4]	2174	203139.14	203127.30
[SHIFT]+[5]	2175	203139.15	203127.31
[SHIFT]+[6]	2176	203140.0	203128.0

Key	Flag	Register bit (16-bit overlaying)	Register bit (32-bit overlaying)
[SHIFT]+[7]	2177	203140.1	203128.1
[SHIFT]+[8]	2178	203140.2	203128.2
[SHIFT]+[9]	2179	203140.3	203128.3

Tab. 37: Numerical keys

Function keys

Key	Flag	Register bit (16-bit overlaying)	Register bit (32-bit overlaying)
[F1]	2201	203141.9	203128.25
[F2]	2202	203141.10	203128.26
[F3]	2203	203141.11	203128.27
[F4]	2204	203141.12	203128.28
[F5]	2205	203141.13	203128.29
[F6]	2206	203141.14	203128.30
[F7]	2207	203141.15	203128.31
[F8]	2208	203142.0	203129.0
[F9]	2209	203142.1	203129.1
[F10]	2210	203142.2	203129.2
[F11]	2211	203142.3	203129.3
[F12]	2212	203142.4	203129.4
[SHIFT]+[F1]	2181	203140.5	203128.5
[SHIFT]+[F2]	2182	203140.6	203128.6
[SHIFT]+[F3]	2183	203140.7	203128.7
[SHIFT]+[F4]	2184	203140.8	203128.8
[SHIFT]+[F5]	2185	203140.9	203128.9
[SHIFT]+[F6]	2186	203140.10	203128.10
[SHIFT]+[F7]	2187	203140.11	203128.11
[SHIFT]+[F8]	2188	203140.12	203128.12
[SHIFT]+[F9]	2189	203140.13	203128.13
[SHIFT]+[F10]	2190	203140.14	203128.14
[SHIFT]+[F11]	2191	203140.15	203128.15
[SHIFT]+[F12]	2192	203141.0	203128.16

Tab. 38: Function keys

Special keys (does not apply to LCD 27)

Key	Flag	Register bit (16-bit overlaying)	Register bit (32-bit overlaying)
[SHIFT]+[←]	2193	203141.1	203128.17
[SHIFT]+[→]	2194	203141.2	203128.18
[SHIFT]+[R]	2195	203141.3	203128.19
[SHIFT]+[I/O]	2196	203141.4	203128.20
[SHIFT]+[=]	2197	203141.5	203128.21
[SHIFT]+[C]	2198	203141.6	203128.22
[SHIFT]+ [ENTER] ([↲])	2199	203141.7	203128.23
[SHIFT]	2200	203141.8	203128.24
[→]	2213	203142.5	203129.5
[←]	2214	203142.6	203129.6
[R]	2215	203142.7	203129.7
[I/O]	2216	203142.8	203129.8
[=]	2217	203142.9	203129.9
[C]	2218	203142.10	203129.10
[ENTER] ([↲])	2219	203142.11	203129.11
[-]	2220	203142.12	203129.12
[SHIFT]+[-]	2221	203142.13	203129.13
[.]	2222	203142.14	203129.14
[SHIFT]+[.]	2223	203142.15	203129.15

Tab. 39: Special keys

LCD 27 special keys

Key	Flag	Register bit (16-bit overlaying)	Register bit (32-bit overlaying)
[↑]	2209	203142.1	203129.1
[↓]	2210	203142.2	203129.2
[C]	2211	203142.3	203129.3
[↲]	2212	203142.4	203129.4

Tab. 40: LCD 27 special keys

3.9.1 Register description

MR 2829 … 2832

Basic flag number for device 1 ... 4

This module register contains the basic flag number of HMI 1 ... 4.

Property	Description
Values	-160 ... 2080
Value after reset	2000
Takes effect	Upon next operation of a key

Tab. 41: Basic flag number for device 1 ... 4

3.10 Enabling and disabling LEDs

Mapping of LEDs

The controller reads the state of LEDs located in HMI keys from the least significant 12 bits of the corresponding register:

Register bit	LED
1	ON
0	OFF

3.11 Assigning LEDs

By default, LEDs located in HMI keys are assigned to a register that is overlaid by special flags.

Register and flag numbers

Multi-display mode allows you to set the number of the register from where to read the LED state via registers.

HMI	Registers	Default LED register number	Flag numbers
Single display	-	203143	2224 ... 2235
1	MR 2833	203143	2224 ... 2235
2	MR 2834	203143	2224 ... 2235
3	MR 2835	203143	2224 ... 2235
4	MR 2836	203143	2224 ... 2235

Tab. 42: LED register and flag numbers

Assignment

Here, the assignments between keys, special flag and overlaid register are listed. The given assignments apply in case of single-display mode and when using the default settings in multi-display mode.

LED in the key	Flag	Register bit
[F1]	2224	203143.0
[F2]	2225	203143.1
[F3]	2226	203143.2
[F4]	2227	203143.3
[F5]	2228	203143.4
[F6]	2229	203143.5
[F7]	2230	203143.6
[F8]	2231	203143.7
[F9]	2232	203143.8
[F10]	2233	203143.9
[F11]	2234	203143.10
[F12]	2235	203143.11

Tab. 43: LED assignment

3.11.1 Register description

MR 2833 … 2836

LED register number for device 1 ... 4

This module register contains the LED register number of HMI 1 ... 4.

Property	Description
Values	100000 ... 1059999
Value after reset	203143

Tab. 44: LED register number for device 1 ... 4

3.12 Monitor function

This chapter describes how to use an HMI to display and change variables independently of the application program.

Requirements

Using the monitor functions requires the following:

– An HMI with numeric keypad is connected to the controller.
– The monitor functions are not blocked in the configuration registers.
– The STX `UserInput()` command for entering numerical values is disabled.

ⓘ Info

Limitations

The monitor function can only access controller variables that are assigned permanent address.

– Registers (%VL)
– Flags (%MX)
– Inputs (%IX)
– Outputs (%QX)

ⓘ Info

Multi-display mode

When using the monitor function in multi-display mode, observe the following:

– To view the monitor function screen, you need to enable it on the respective HMI by pressing the **[R]** or **[I/O]** key.
– The controller cannot identify the HMI where subsequent keystrokes are carried out.

3.12.1 HMIs supporting the monitor function

The following table lists the range of Bucher Automation AG alphanumeric HMIs supporting the monitor function:

Designation	Keys	Variables
LCD 16 + NUM 25	[R] and [I/O]	– Registers – Flag – Inputs – Outputs
LCD 34	[R]	– Registers – Flag
LCD 52	[R] and [I/O]	– Registers – Flag – Inputs – Outputs

Designation	Keys	Variables
LCD 54(Z)	[R] and [I/O]	– Registers – Flag – Inputs – Outputs
LCD 60	[R] and [I/O]	– Registers – Flag – Inputs – Outputs
LCD 110	[R] and [I/O]	– Registers – Flag – Inputs – Outputs

3.12.2 Description of the keys

Keys in use

The monitor function uses following keys:

Key	Description
[R]	Initiates the monitor function for registers or flags
[I/O]	Initiates the monitor function for outputs, inputs or flags
[0] ... [9]	Enters a variable number or numerical
[.] or [,]	Enters a decimal point or decimal comma
[-]	Enters a negative/positive numerical value; you can press the key any time during input.
[C]	– Clears current entry – Aborts the monitor function
ENTER ([↲])	– Confirms the input; displays the variable or accepts the value – Switches between monitor function screen and default screen

Tab. 45: Monitor function – keys in use

ⓘ Info

Limitations

While the monitor function is active, keys used by this function are not mapped to the key flags.

3.12.3 Displaying and changing variables

Initiating the monitor function

To initiate the monitor function, press the key **[R]** or the key **[I/O]**. Using these keys, you can change the variable type as long as the entry of the variable number has not been completed by pressing the **ENTER** ([↲]) key.

Displaying register content

To display content of a register, proceed as follows:

1. Press the **[R]** key.
2. Enter the register number.
3. Press the **ENTER** ([↲]) key.

   ⇨ The HMI switches to the monitor function screen.

⇨ The screen shows the register content for the period set in MR 2819 *Monitor functions - Display duration*. After expiry of the set period, the HMI returns to the default screen.

Displaying the flag state

To display a flag state, proceed as follows:

1. Press the **[R]** key twice or the **[I/O]** key three times.
2. Enter the flag number.
3. Press the **ENTER** ([↲]) key.

   ⇨ The HMI switches to the monitor function screen.

⇨ The screen shows the flag state for the period set in MR 2819 *Monitor functions - Display duration*. After expiry of the set period, the HMI returns to the default screen.

Displaying the output state

To display an output state, proceed as follows:

1. Press the **[I/O]** key.
2. Enter the output number.
3. Press the **ENTER** ([↲]) key.

   ⇨ The HMI switches to the monitor function screen.

⇨ The screen shows the output state for the period set in MR 2819 *Monitor functions - Display duration*. After expiry of the set period, the HMI returns to the default screen.

Displaying the input state

To display an input state, proceed as follows:

1. Press the **[I/O]** key twice.
2. Enter the input number.
3. Press the **ENTER** ([↲]) key.

   ⇨ The HMI switches to the monitor function screen.

⇨ The screen shows the input state for the period set in MR 2819 *Monitor functions - Display duration*. After expiry of the set period, the HMI returns to the default screen.

Modifying a variable value

To modify a variable value, proceed as follows:

1. Have the value of the variable displayed (see above).
2. Press the **[=]** key.
3. Enter a new value.
4. Press the **ENTER** ([↲]) key.
   ⇨ The system prompts you to enter a new value for this variable. Meanwhile, the current value is displayed.

⇨ The new value is written to the variable. The variable value is displayed for the period of time set in MR 2819 *Monitor functions - Display duration*. After expiry of the set period, the HMI returns to the default screen.

Aborting the monitor screen

If you wish to abort the display of a variable value ahead of the set period (default: 3.5 s) and return to the default screen, press the **ENTER** ([↲]) key.

Displaying a variable again

To switch back to the variable screen from the default screen, press the **ENTER** ([↲]) key again. The monitor function screen is shown for another 3.5 s.

3.12.4 Register description

MR 2818

Disabling/enabling the monitor function

This module register is bit-coded. These bits can be used to disable/enable individual monitor functions. Keys are also mapped to flags with the monitor function being disabled.

Property	Description	
Values	0 ... 255	
Value after reset	255	
Description of the bits		
Bit 0	**[R] key**	
	0 =	**[R]** key has no monitor function
	1 =	**[R]** key has monitor function
Bit 1	**Displaying the flag state**	
	0 =	**[R]** and **[I/O]** keys without monitor function **Display flag state**
	1 =	**[R]** and **[I/O]** keys with monitor function **Display flag state**
Bit 2	**Displaying the output state**	
	0 =	**[I/O]** key without monitor function **Display output state**
	1 =	**[I/O]** key with monitor function **Display output state**

Property	Description	
Bit 3	**Displaying the input state**	
	0 =	**[I/O]** key without monitor function **Display input state**
	1 =	**[I/O]** key with monitor function **Display input state**
Bit 4	**Changing register contents**	
	0 =	**[=]** key without monitor function **Change register contents**
	1 =	**[=]** key with monitor function **Change register contents**
Bit 5	**Changing the flag state**	
	0 =	**[=]** key without monitor function **Change flag state**
	1 =	**[=]** key with monitor function **Change flag state**
Bit 6	**Changing the output state**	
	0 =	**[=]** key without monitor function **Change output state**
	1 =	**[=]** key with monitor function **Change output state**
Bit 7	**Displaying the input state permanently**	
	0 =	**[=]** key without monitor function
	1 =	**[=]** key with monitor function

Tab. 46: Disabling/enabling monitor functions

MR 2819

Display time for monitor functions

This module register contains the display time in multiples of 100 ms.

Property	Description
Values	0 ... 65,535
Value after reset	35 (3.5 s)
Takes effect	Upon next switch-over to the monitor function screen

Tab. 47: Display time for monitor functions

MR 2820

Switch-over to monitor function screen

This module register configures the function of the **ENTER** ([↵]) key.

Property	Description	
Values	0	Switching between monitor function screen and default screen is enabled
	1	Switching between monitor function screen and default screen is disabled

Tab. 48: Switch-over to monitor function screen

MR 2821

Dialog language

This module register configures the dialog language of the monitor function.

Property	Description	
Values	0	German
	1	English
Takes effect	Next time the monitor function is launched	

Tab. 49: Dialog language

4 Controlling printer and serial interfaces

This chapter describes how to control printer and serial interfaces from within the application program of a controller.

Controlling the interfaces

Printer and serial interfaces can be controlled in 2 ways:

– Direct access to registers of the interface
– Using display functions included in the STX language set

Supported interfaces

The following table lists the printer and serial interfaces supported by the controller. It also indicates the device number the display command needs to address in order to output information to the interface.

Module	Interface	Device number
JX2-PRN1	Centronics printer module	8
User-programmable serial interface	Serial interface of the CPU	9
JX2-SER1	Serial interface module	11
JX3-MIX2	Serial interface on the module	11

Configuring interfaces

For more information on how to configure and program interfaces refer to the documentation on the corresponding module.

ⓘ Info

Further information

For more information on this subject refer to the application-oriented manual *User-Programmable Interfaces* available for download from our homepage.

Direct interface access

To output special or control characters or to retrieve the state of the external device, direct access to the registers of the interface is required. For more information on how to access registers refer to the documentation on the corresponding module.

Display functions

The controller provides the following display options:

– Displaying text strings
– Displaying the contents of variables

ⓘ Info

Further information

For a detailed description of the STX commands mentioned in this chapter, see the JetSym Help at *JetSym ST/STX projects* > *JetSym STX programming language* > *Display commands and user input*.

4.1 Registers

Info

Limitations

The settings made in the given registers take global effect, meaning they will impact all functions for controlling printer and serial interfaces. If different settings are used in several tasks of the application program, then these settings may impact each other.

The registers are clustered into one register block.

Register numbers

Basic register number	Register numbers
220000	222806 ... 222838

In this chapter, only the last 4 digits of a register number are specified, e.g. MR 2838. This is the module register. Add the basic register number of the corresponding device to determine the complete register number, for example 222838.

4.2 Device numbers

The device number defines the output device. The device number is used as the **<DeviceNo>** parameter with the `DisplayText()`, `DisplayText2()`, and `DisplayValue()` commands (see JetSym Help).

Overview

You may enter the following values for the parameter device number:

Number	Part	Description
0	Default device	The device number to be used is contained in MR 2824
1	HMI 1	Multi-display mode
2	HMI 2	Single-/Multi-display mode
3	HMI 3	Multi-display mode
4	HMI 4	Multi-display mode
5 ... 7	Reserved	Do not use
8	Printer module	Output to a printer module on the JX2 system bus
9, 10	Serial interface	Output to the user-programmable serial interface of the controller
11	Serial interface module	Output to a serial interface module on theJX2 or JX3 system bus

Tab. 50: Device number summary

4.2.1 Register description

MR 2824

Device number of the default device

This module register contains the device number of the default device. If you always specify the default device in the application program (device number = 0), you can select during runtime which device is to be used.

Property	Description
Values	1 ... 11
Value after reset	2

Tab. 51: Device number of the default device

4.3 Interface module numbers

Redirecting display commands to a printer or serial interface module connected to the JX2 or JX3 system bus requires configuration of the module number.

Redirection to an internal user-programmable serial interface is unambiguously defined by the device number. Therefore, no additional configuration is required.

Determining module numbers

The module number to be entered is calculated based on the number of the module on the system bus plus a system bus constant:

```
Module number = number of the module + system bus constant
```

System bus	System bus constant
JX3	100
JX2	200

Tab. 52: System bus constant

4.3.1 Register description

MR 2837

Printer module number

This module register contains the number of the module where to redirect the output of the display command for device 8.

Property	Description
Values (JX3 system bus)	102 ... 117
Values (JX2 system bus)	202 … 224
Takes effect	Upon the next execution of the STX `DisplayText()` or `DisplayValue()` command

Tab. 53: Printer module number

MR 2838

Serial interface module number

This module register contains the number of the module where to redirect the output of the display command for device 11.

Property	Description
Values (JX3 system bus)	102 ... 117
Values (JX2 system bus)	202 … 224
Takes effect	Upon the next execution of the STX `DisplayText()` or `DisplayValue()` command

Tab. 54: Serial interface module number

4.4 Outputting text strings

STX commands

To output text strings, use the following STX commands (STX functions):

- `DisplayText()`
- `DisplayText2()`

4.4.1 DisplayText() STX command

Declaration

```
Function DisplayText(Dev:Int,
                     Pos:Int,
                     Const Ref Text:String);
```

Parameter

Parameter	Value	Description
Dev	8 ... 11	Number of the device where to output the text strings
Pos	Not relevant	Is not evaluated
Text	Text string to be output	Hard-coded text, or name of a string variable

Tab. 55: `DisplayText()` parameter

How to use this command

Command-line syntax for outputting text strings to a printer module:

```
DisplayText(8,0, "Hello World!");
DisplayText(8, 0, StringVar);
```

Operating principle

The first command outputs text string 'Hello World!' the printer module. The second STX command outputs the content of the string variable **StringVar**.

The application program task stops at the `DisplayText()` command waiting for the entire text to be output.

4.4.2 DisplayText2() STX command

Declaration

```
Function DisplayText2(Dev:Int,
                      Pos:Int,
                      Const Ref Text1:String)
                      Const Ref Text2:String);
```

Parameter

Parameter	Value	Description
Dev	8 ... 11	Number of the device where to output the text strings
Pos	Not relevant	Is not evaluated
Text1	Text string to be output	Hard-coded text, or name of a string variable
Text2	Text string to be output	Hard-coded text, or name of a string variable

Tab. 56: `DisplayText2()` parameter

How to use this command

Command-line syntax for outputting one out of two text strings to a serial interface module:

```
DisplayText2(11, 0, 'Fehler:', 'Error:');
```

Operating principle

The STX command outputs the text string 'Fehler:', or 'Error:' to a serial interface module. MR 2806 controls which text string will be displayed.

The application program task stops at the `DisplayText2()` command waiting for the entire text to be output.

4.4.3 Register description

MR 2806

Text selection for DisplayText2

The value in this module register specifies which one of the two text strings is to be output.

Property	Description	
Values	0	Text1
	1	Text2

Tab. 57: Text selection for DisplayText2

4.5 Outputting numerical values

STX command

To display numerical values, use the following STX command:

– `DisplayValue()`

Format of the output value

The format of numerical values to be output can be adjusted to suit application needs. To do so, use the following parameters:

– Display field length
– Number of decimal places
– With or without sign place
– Decimal or hexadecimal notation

Outputting numerical values

When outputting numerical values, the following formatting rules apply:

– The numerical value is displayed right-aligned.
– The first character to appear in the display field is the sign unless writing the sign character was disabled beforehand.
– The first character of a numerical value to be written is the leading space or the leftmost digit.
– Positive numericals are prefixed by a space character as sign. Negative numericals are prefixed by minus '-'.
– If the display field is too small, the leftmost digits are truncated.
– The value is rounded to the set decimal places.

4.5.1 DisplayValue() STX command

Declaration

```
Function DisplayValue(Dev:Int,
                      Pos:Int,
                      Value:Double);
```

Parameter

Parameter	Value	Description
Dev	8 ... 11	Number of the device where to output the value
Pos	Not relevant	Is not evaluated
Value	Value to be output	Constant value, name of a register or a variable

Tab. 58: `DisplayValue()` parameter

How to use this command

Command-line syntax for outputting a numerical value to a printer module:

```
DisplayValue(8, 0, -12,345);
DisplayText(8, 0, '$t');
DisplayValue(8, 0, Axis2.Position);
DisplayText(8, 0, '$n');
```

Operating principle

The first STX command outputs the value **-12,345**. The second STX command inserts a tab ($t). The third STX command outputs the content of variable **Axis2.Position**. The fourth STX command triggers a carriage return and line feed ($n).

The application program task stops at the `DisplayText()` or `DisplayValue()` command waiting for all characters to be output.

4.5.2 Configuring the screen size

Setting the length of the display field

MR 2812 defines the length of the display field for numerical values.

```
MR 2812 = Number of digits + sign [+ decimal point]
```

Example

Parameter	Description
Number of digits	6
Show sign (MR 2816)	0 (Yes)
Decimal point	No
Field length (MR 2812)	7
Display	7 characters

Tab. 59: Setting the display field length – example

Setting the sign option

MR 2816 defines whether or not the sign is displayed.

To set the display field length in MR 2812 correctly, include the sign's place into the calculation, even if no sign is to be displayed.

Example

Parameter	Description
Number of digits	6
Show sign (MR 2816)	1 (No)
Decimal point	No
Field length (MR 2812)	7
Display	6 characters

Tab. 60: Setting the sign option – example

Setting the number of decimal places

MR 2810 defines the number of decimal places.

Displaying decimal places might require adjustment of the display field length in MR 2812.
The reason: The decimal point uses one place in the display field.

Setting the numerical value format

Numericals can be displayed in either decimal or hexadecimal format.

Flag 2060 sets the format of numerical values.

4.5.3 Register description

MR 2810

Number of decimal places for DisplayValue

This module register contains the number of decimal places to be displayed for numerical values.

Property	Description
Values	0 ... 4
Takes effect	Upon next execution of STX `DisplayValue()` command

Tab. 61: Number of decimal places for DisplayValue

MR 2812

Field length for DisplayValue

This module register contains the length of the display field.

Property	Description
Values	1 ... 12
Value after reset	11
Takes effect	Upon next execution of STX `DisplayValue()` command

Tab. 62: Field length for DisplayValue

MR 2816

Sign option for DisplayValue

The value in this module register specifies which one of two text strings is to be output.

Property	Description	
Values	0	Sign will be displayed
	1	Sign will not be displayed
Takes effect	Upon next execution of STX `DisplayValue()` command	

Tab. 63: Sign option for DisplayValue

Flag 2060

Numerical value format for DisplayValue

Flag 2060 sets the numerical format.

Property	Description	
Values	0	Decimal
	1	Hexadecimal
Takes effect	Upon next execution of STX `DisplayValue()` command	

Tab. 64: Numerical value format for DisplayValue

5 Register overview

The following table provides a summary of existing module registers.

MR	Description
2804	Number of characters on the screen
2805	Number of characters per line
2806	Text selection (`DisplayText2()`)
2808	Number of decimal places (`UserInput()`)
2810	Number of decimal places (`DisplayValue()`)
2811	Maximum number of decimal places (`UserInput()`)
2812	Field length (`DisplayValue()`)
2813	Field length (`UserInput()`)
2814	Indirect cursor position
2815	Suggested value (`UserInput()`)
2816	Displaying the sign
2817	State of the UserInput (`UserInput()`)
2818	Monitor functions (enable/disable)
2819	Display time for monitor functions
2820	Switch-over to monitor function screen
2821	Dialog language of the monitor function
2824	Indirect device number (default device)
2825	Device number of HMI 1 (multi-display mode)
2826	Device number of HMI 2 (multi-display mode)
2827	Device number of HMI 3 (multi-display mode)
2828	Device number of HMI 4 (multi-display mode)
2829	Basic flag number of HMI 1 (multi-display mode)
2830	Basic flag number of HMI 2 (multi-display mode)
2831	Basic flag number of HMI 3 (multi-display mode)
2832	Basic flag number of HMI 4 (multi-display mode)
2833	Register number for LEDs on HMI 1 (multi-display mode)
2834	Register number for LEDs on HMI 2 (multi-display mode)
2835	Register number for LEDs on HMI 3 (multi-display mode)
2836	Register number for LEDs on HMI 4 (multi-display mode)
2837	Printer module number
2838	Serial interface module number
2839	Control character for clearing the screen content
2840	Control character for deleting a text string up to the end of a line

6 Service

6.1 Technical support

In case of questions, suggestions, or issues, please contact our experts from Technical Support. You may reach out by phone or through the contact form on our homepage:

Technical Support | Bucher Automation - We automate your success.

Or email us:

support@bucherautomation.com

Please supply the following information when contacting Technical Support:

– Hardware revision and serial number
  The hardware revision and serial number is printed on the nameplate of the product.

Bucher Automation AG
Thomas-Alva-Edison-Ring 10
71672 Marbach/Neckar, Germany
T +49 7141 2550-0
info@bucherautomation.com

www.bucherautomation.com