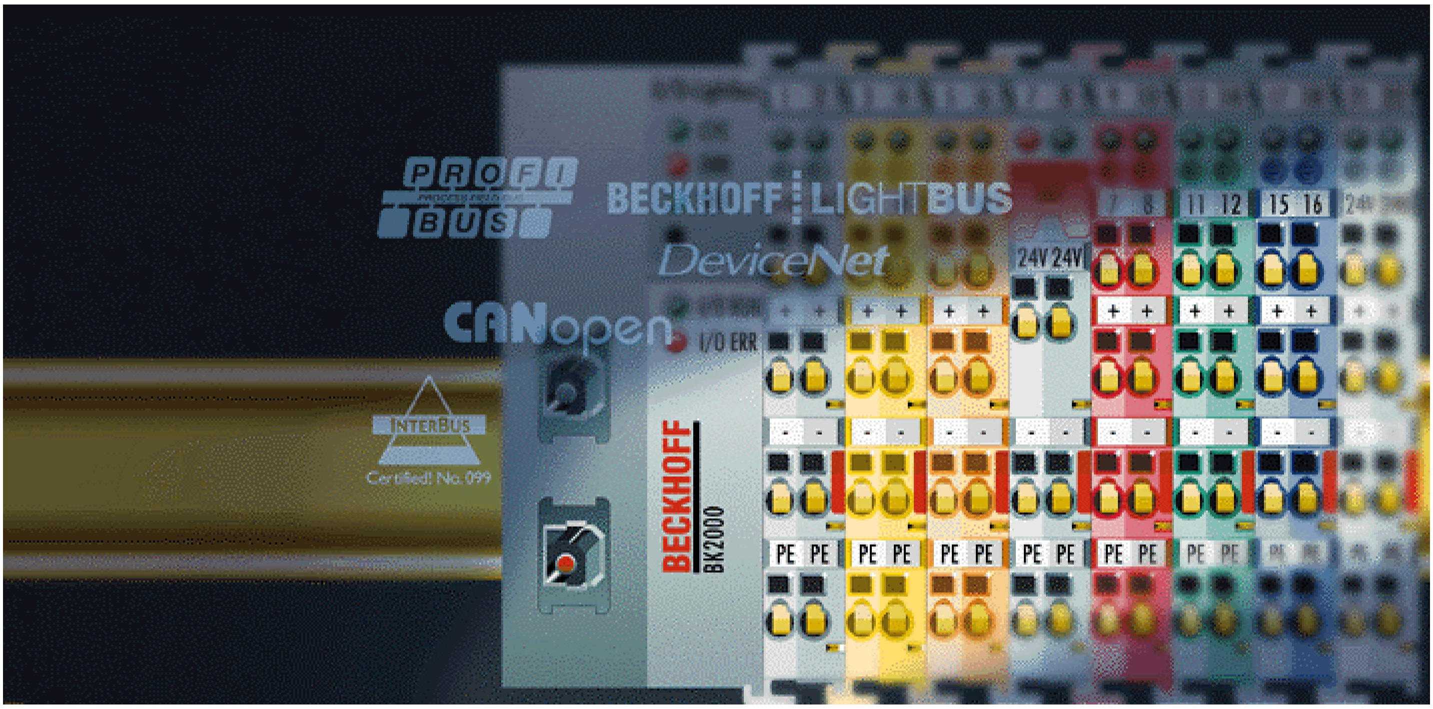


KS8000LV – LabVIEW© Communication Library for the BK8xxx Serial Bus Coupler

ActiveX Version 2.0.0.2
DLL-Version 2.0.0.2
LabView-Version 2.0.0.2

Last change: 26.03.99
Doc. Version 2.0

BECKHOFF
INDUSTRIE ELEKTRONIK

KS8000LV: BkComLV

LabVIEW©

LabVIEW©

The LabVIEW© graphical programming system from National Instruments supports the creation of applications without the need to write large amounts of program text. Programming is achieved through the selection, insertion and linking of graphical symbols into what are known as block diagrams, using a mouse. LabVIEW© programs are called **virtual instruments**, or VI for short.

More extensive information on use and programming can be found in the LabVIEW© manuals.

SubVI – LabVIEW© Subroutines

SubVI

The LabVIEW© subroutine concept allows one VI to be linked into another VI as a so-called SubVI. It is important to note the arbitrary possibilities for creating hierarchies, there being no quantitative limit on the nesting depth of VIs. Thus a SubVI can itself contain any number of other SubVIs.

SubVIs are analogous to the subroutines of classical programming languages, and can both receive values and return them (parameterisation).

When using them in a VI it must be noted, that SubVIs of the same name always represent just one instance, regardless of the number and location of VIs into which it is linked as a SubVI. This means that such linked SubVIs are in fact merely references to one original VI, and therefore has only one data region available to it.

If a SubVI needs to remember particular data or states, it is therefore necessary that each such SubVI is copied under a new name before linking. This procedure must be followed when using the BkComLV VIs in your own VIs.

The BkComLV VI

Integrate BkCom functionality by means of VI

The BkComLV consists of four independent VIs. They are

- Open-BkComLV
- Read-BkComLV
- Read Write-BkComLV
- Close-BkComLV

It is easy to make use of these VIs as SubVIs in your own VI, thus linking in the BkCom DLL functionality from a LabVIEW program.

More detailed information on the mode of operation and area of application of the BkCom DLL is found in the help files for BkComOcx.

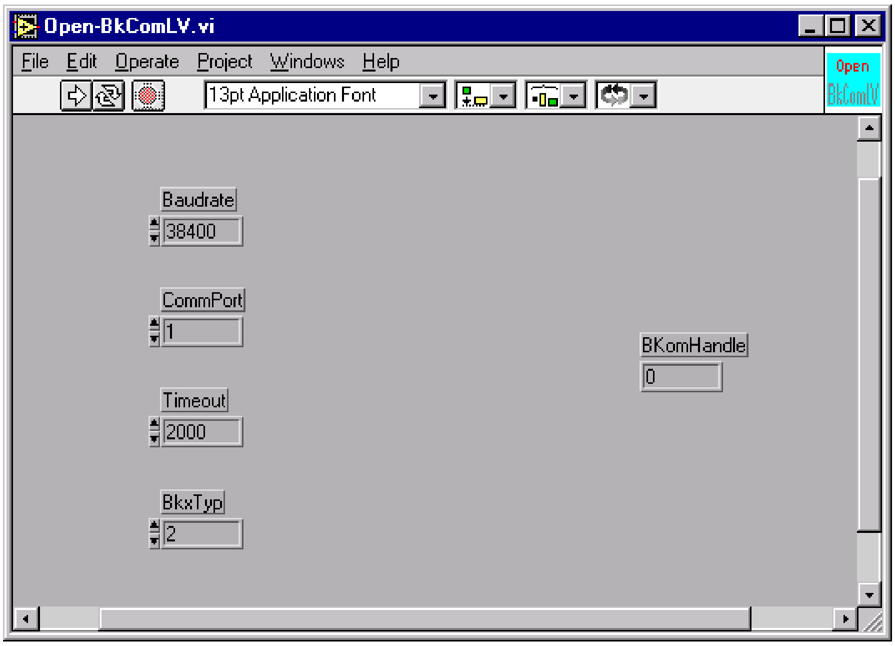
The BkComLV DLL supplied should be installed for the operation of BkComLV.

Open-BkComLV

Function

Opens the serial interface.

Display



Baudrate 38400

CommPort 1

Timeout 2000

BkxTyp 2

BKomHandle 0

Input data

Baud rate	-	Long Control
CommPort	-	Long Control
Timeout	-	Long Control
Bkx type	-	Long Control

Note: All input parameters are required.

Value range

Baud rate in baud:
Baud rate = {
9600,
19200,
38400
}

Communication port number:
CommPort = {
1,
: ,
n
}

Time in milliseconds:
Timeout = {
0,

N
}

Constant:
Bkx type = {
1, // BK8000
2 // BK8100
}

Output data BkcomHandle - Long Indicator

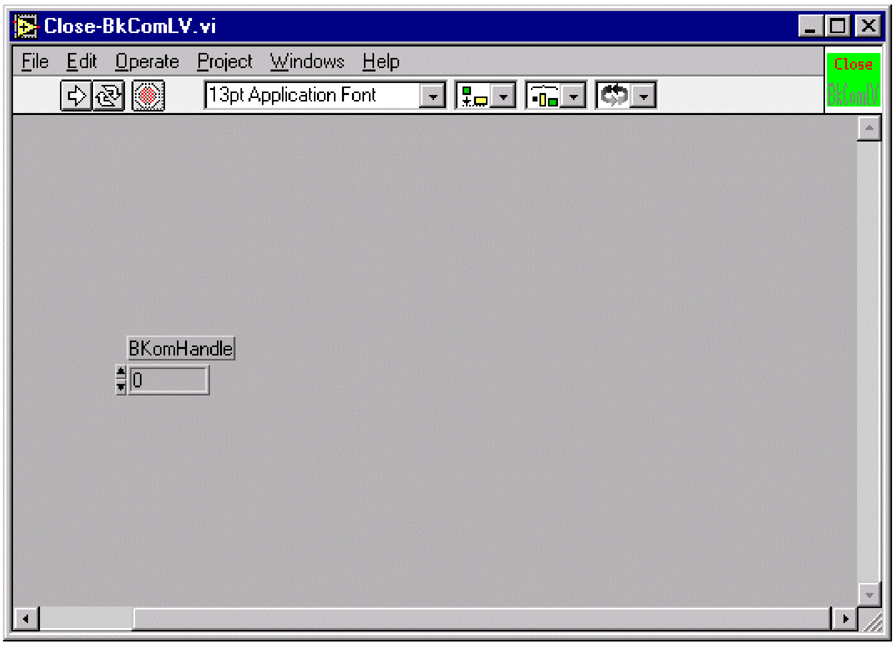
Value range BkcomHandle - Unique number <> 0

Explanation The Open-BkComLV.vi opens the given communication port and returns a handle to it.

Close-BkComLV

Function Closes the serial interface.

Display



BKomHandle
0

Input data BkcomHandle - Long Indicator

Note: All input parameters are required.

Value range BkcomHandle - Unique number <> 0

Output data None

Explanation

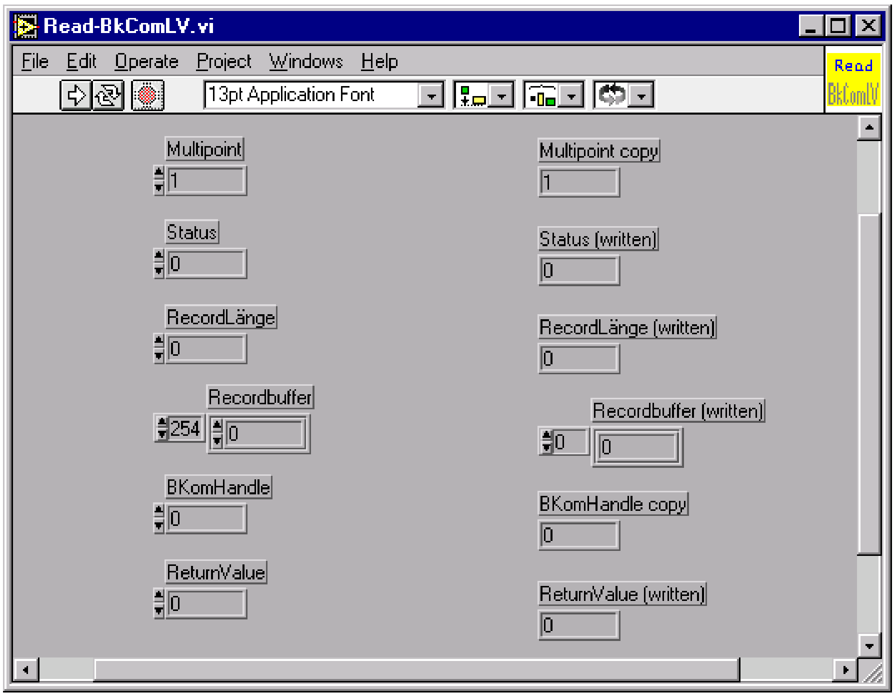
The Close-BkComLV.vi closes the given communication port.

Read-BkComLV

Function

Reads process data from the serial interface.

Display



Multipoint 1
Status 0
RecordLänge 0
Recordbuffer 254 0
BKomHandle 0
ReturnValue 0
Multipoint copy 1
Status (written) 0
RecordLänge (written) 0
Recordbuffer (written) 0 0
BKomHandle copy 0
ReturnValue (written) 0

Input data

Multipoint	-	Long Control
Status	-	Long Control
Record length	-	Long Control
Record buffer	-	One-dimensional array of long controls
BkcomHandle	-	Long Control
ReturnValue	-	Long Control

Note: All input parameters are required.

Value range

Multipoint	-	Station address of the bus coupler
Status	-	Irrelevant, standard value 0
Record length	-	Irrelevant, standard value 0
Record buffer	-	Irrelevant, standard value 0
BkcomHandle	-	Unique number <> 0
ReturnValue	-	Irrelevant, standard value 0

Note: Irrelevant means that the value supplied has no effect on the function. It is however necessary to supply it, in order to make the type and size known, so that the output data are returned correctly.

Output data

Multipoint copy	-	Long Indicator
Status (written)	-	Long Indicator
Record length (written)	-	Long Indicator
Record buffer (written)	-	One-dimensional array of long indicators
BkCom handle copy	-	Long Indicator
ReturnValue (written)	-	Long Indicator

Value range	Multipoint copy	-	Station address of the bus coupler
	Status (written)	-	State of the bus coupler
	Record length (written)	-	Number of process values read
	Record buffer (written)	-	Process values read
	BkCom handle copy	-	Unique number <> 0, must have been obtained from an Open-BkComLV.vi
	ReturnValue (written)	-	0 : ok, <> 0 error

Explanation

Read-BkComLV.vi is a call for synchronous communications to read out the entire input process image of a BK8x00 bus coupler. The size of the process image depends on the number and type of terminals plugged into the coupler.

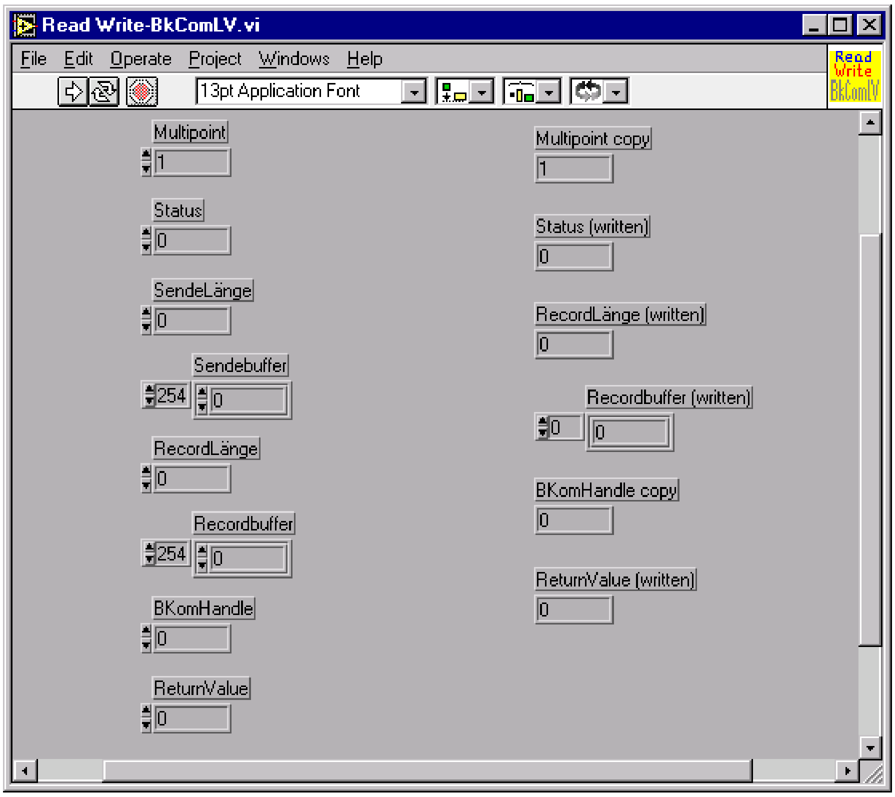
Although the receive buffer is of type long (32 bits), the coupler only transfers data in the low word (see also the BkComOCX example).

Read Write-BkComLV

Function

Writes and reads process data through the serial interface.

Display



Multipoint 1
Status 0
SendeLänge 0
Sendebuffer 254 0
RecordLänge 0
Recordbuffer 254 0
BKomHandle 0
ReturnValue 0

Multipoint copy 1
Status (written) 0
RecordLänge (written) 0
Recordbuffer (written) 0 0
BKomHandle copy 0
ReturnValue (written) 0

Input data	Multipoint	-	Long Control
	Status	-	Long Control
	Send length	-	Long Control
	Send buffer	-	One-dimensional array of long controls
	Record length	-	Long Control
	Record buffer	-	One-dimensional array of long controls
	BkcomHandle	-	Long Control
	ReturnValue	-	Long Control

Note: All input parameters are required.

Value range	Multipoint	-	Station address of the bus coupler
	Status	-	irrelevant, standard value 0
	Send length	-	Number of words to be written (a word in a long value) from offset 0 in the process output image
	Send buffer	-	Data words to be sent
	Record length	-	irrelevant, standard value 0
	Record buffer	-	irrelevant, standard value 0
	BkcomHandle	-	Unique number <> 0, must have been obtained from an Open-BkComLV.vi
	ReturnValue	-	irrelevant, standard value 0

Note: Irrelevant means that the value supplied has no effect on the function. It is however necessary to supply it, in order to make the type and size known, so that the output data are returned correctly.

Output data	Multipoint copy	-	Long Indicator
	Status (written)	-	Long Indicator
	Record length (written)	-	Long Indicator
	Record buffer (written)	-	One-dimensional array of long indicators
	BkCom handle copy	-	Long Indicator
	ReturnValue (written)	-	Long Indicator

Value range	Multipoint copy	-	Station address of the bus coupler
	Status (written)	-	State of the bus coupler
	Record length (written)	-	Number of process values read
	Record buffer (written)	-	Process values read
	BkCom handle copy	-	Unique number <> 0
	ReturnValue (written)	-	0 : ok, <> 0 error

Explanation

Read Write-BkComLV.vi is a call for synchronous communications to write the whole output process image and to read out the entire input process image of a BK8x00 bus coupler. The size of the process image read depends on the number and type of terminals plugged into the coupler.

The entire process output image MUST be written. It is not possible to write a partial segment.

Although the send and receive buffers are of type long (32 bits), the coupler only transfers data in/from the low word (see also the BkComOCX example).

Creating a VI with BkCom-VIs

Procedure

The general procedure for creating a LabVIEW© VI including a BkComLV-VI is illustrated in what follows.

* Depending on the number of communication ports which are to be opened, the Open-BkComLV and Close-BkComLV VIs must be copied correspondingly often, renaming them in accordance with their purpose. Original copies are located in the appropriate directory (..\LabView).

* Depending on the number of read and write functions to be located in the process images of BK8x00 bus couplers, the corresponding number of copies of the 'Read-BkComLV' and 'Read Write-BkComLV' VIs must be made, renamed again in accordance with their purpose.

* From the function palette, using 'Select User VI', select the copied BkComLV VI and place it into your own block diagram.

* Create suitable LabVIEW© variables (controls and indicators) on the front panel of your own VI (see the BkComLV-VI input and output data), and wire them to the nodes of the BkComLV SubVI.

Example VI with BkComLV VI

There are some example VIs for BkComLV VI. In preparation, either one or two Bk8x00 bus couplers must have at least 3 digital output terminals (e.g. KL2012) inserted, and other input terminals can optionally be inserted behind them. The bus couplers are to be connected to the COM port(s). The correct port number and the station number of the couplers (Multipoint) must be entered into the example VI. All examples are started from LabVIEW© by 'Run' (Ctrl+r), and are properly ended by turning the program switch to 'End'. If the test is successful, a 'running light' effect is to be seen at the output terminals. Signals asserted at the input terminals must lead to an alteration in the values in the corresponding record buffer arrays. The following examples are available for download from the FTP server at ftp://ftp.beckhoff.com /Software/KS8000Bs:

- 'OnePort': The effect operates via a communication port and a bus coupler in one VI: **BkcomLVCompleteSubVI.vi**.
- ‚TwoPorts': The effect operates via two communication ports and two bus couplers in one VI: **DblBKcomLVCompleteSubVI.v*i*.**
- ‚OneAndOnePort': The effect operates via two communication ports, two bus couplers in two VIs to be opened simultaneously: **BKcomLVCompleteSubVI_1.vi** and **BKcomLVCompleteSubVI_2.vi**.