test write.vi


ON/OFF

numéro de port[port number] numéro de port Les paramètres pour les **numéros de ports** série dépendent de la plate-forme que vous utilisez : Windows, Macintosh ou UNIX.

Nvoie

Case Constant (See Context Help) LabVIEW created this control from a constant that was wired to a Case structure. In LabVIEW 7.1 and earlier, subVIs could be kept in memory by wiring a constant to a Case structure and placing the subVI in a case that would not be executed. For example, you could wire a True constant to the Case structure and place a subVI in the False case of the Case structure, and LabVIEW would load the subVI along with the calling VI. This no longer works in LabVIEW 8.0 because the Case structure is optimized to remove code that will not be executed. Changing the constant to a control ensures that this VI behaves as it did in LabVIEW 7.1 and earlier. If the intent is not to keep subVIs in memory, you can change this control back to a constant by right-clicking the control and selecting Change to Constant from the shortcut menu. The proper way to keep a subVI in memory without calling it is to use a Static VI Reference.

chaîne écrite[string to write] "chaîne à écrire" représente les données à écrire sur le port série. Si le nombre de caractères dans chaîne à écrire est supérieur à la taille du buffer spécifiée dans Initialiser le port série, c'est le nombre de caractères égal à la taille du buffer qui sera écrit.

chaîne lue[string read] Le VI retourne les octets lus dans chaîne lue.

port number
I32

Nvoie
U16

True

False

Case Constant
(See Context Help)

ON/OFF
TF

port number
Value

STOP

string to write
abc

string read
abc

5
100
0

SA32
Write

False

Constant
(Context Help)

port number
Value

True

RUN

5
100
-256

SA32
Write

SA32
Write

sa32 write.vi
C:\Users\guillaume.divrechy\Desktop\3- SOLINOV\DOC SA32\SOFT\sa32 write.vi
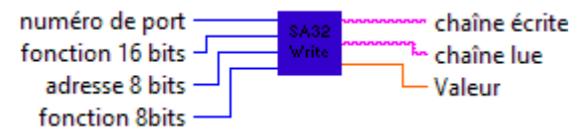
"test write.vi History"
Current Revision: 3

Position in Hierarchy

Iconified Cluster Constants

sa32 write.vi

numéro de port   SA32 Write   chaîne écrite
fonction 16 bits   chaîne lue
adresse 8 bits   Valeur
fonction 8bits

numéro de port
0

fonction 16 bits
0

adresse 8 bits
0

fonction 8bits
0

chaîne écrite
0103 001E 0003 65CD

chaîne lue
0103 0600 0000 0000 0361 74

Valeur
0,00000

LSB
65

MSB
CD

code d'erreur
×0

valeur paramètres
(16 bits)
(décimal)

Tableau

0

|	
23	0
0	0
0	3
0	0
0	0
0	0
0	0
0	0
0	0
0	0
0	
1	

Erreur
0

I32 **numéro de port[port number] numéro de port** Les paramètres pour les **numéros de ports** série dépendent de la plate-forme que vous utilisez : Windows, Macintosh ou UNIX.

U16 **adresse 8 bits**

U8 **fonction 16 bits**

I16 **fonction 8bits**

EXT **Valeur[Valeur]**

abc **chaîne écrite[string to write]** "chaîne à écrire" représente les données à écrire sur le port série. Si le nombre de caractères dans chaîne à écrire est supérieur à la taille du buffer spécifiée dans Initialiser le port série, c'est le nombre de caractères égal à la taille du buffer qui sera écrit.

abc **chaîne lue[string read 3]** Le VI retourne les octets lus dans chaîne lue.

U8 **LSB[LSB]**

U8 **MSB[MSB]**

I32 **code d'erreur[error code]** Si **code d'erreur** est non nul, une erreur a eu lieu.
Vous pouvez câbler **code d'erreur** à l'un des VIs gestionnaires d'erreur, qui décrivent l'erreur et vous donnent des options sur la manière de procéder lorsqu'une erreur survient.
Certains codes d'erreur renvoyés par les VIs de port série sont spécifiques à une plate-forme. Veuillez vous reporter à la documentation de votre système pour obtenir une liste des codes d'erreur.

[U8] **tableau d'octets non signés**

    U8

I32 **taille(s)[taille(s)]**
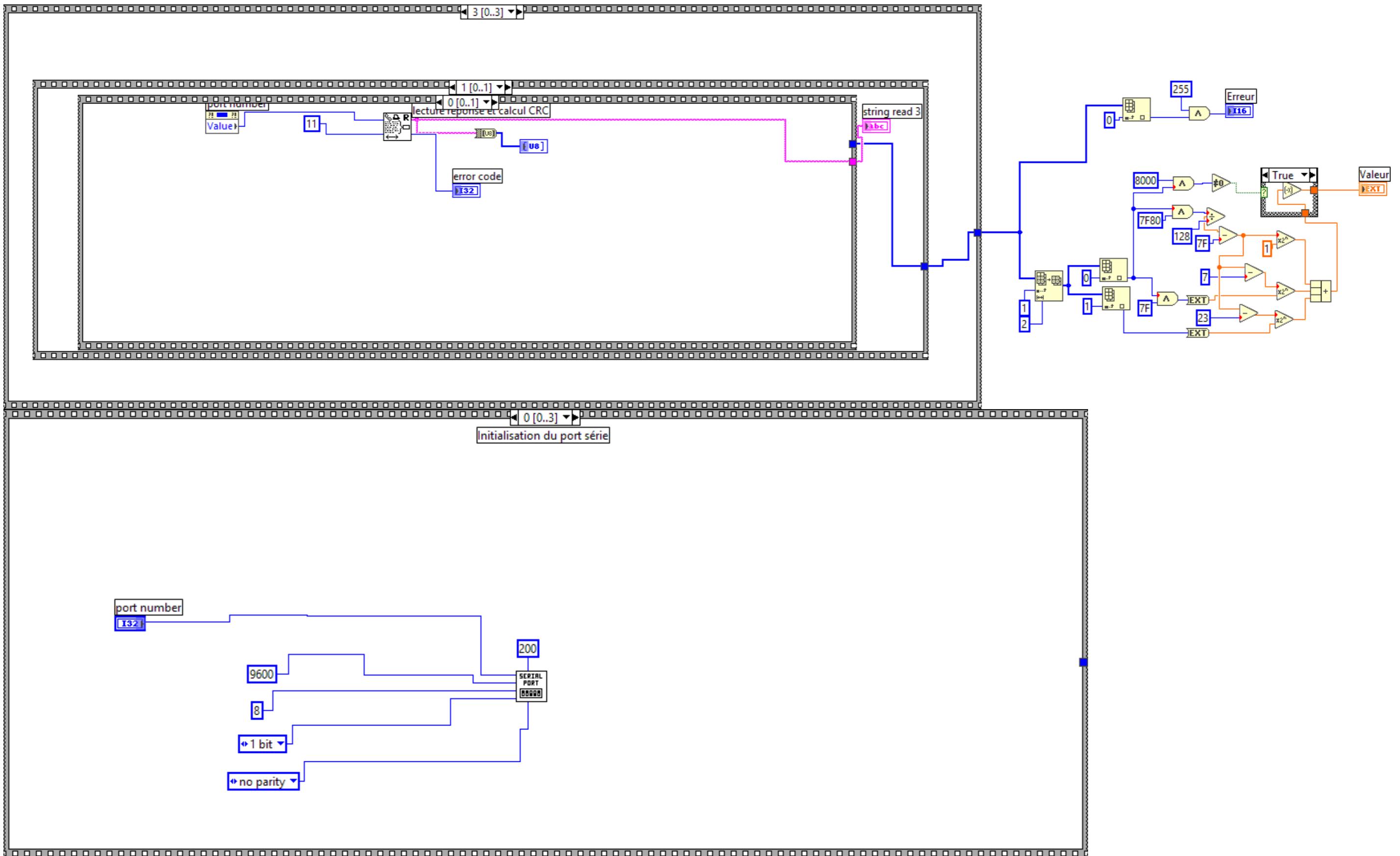
[U16] **Tableau[Tableau]**
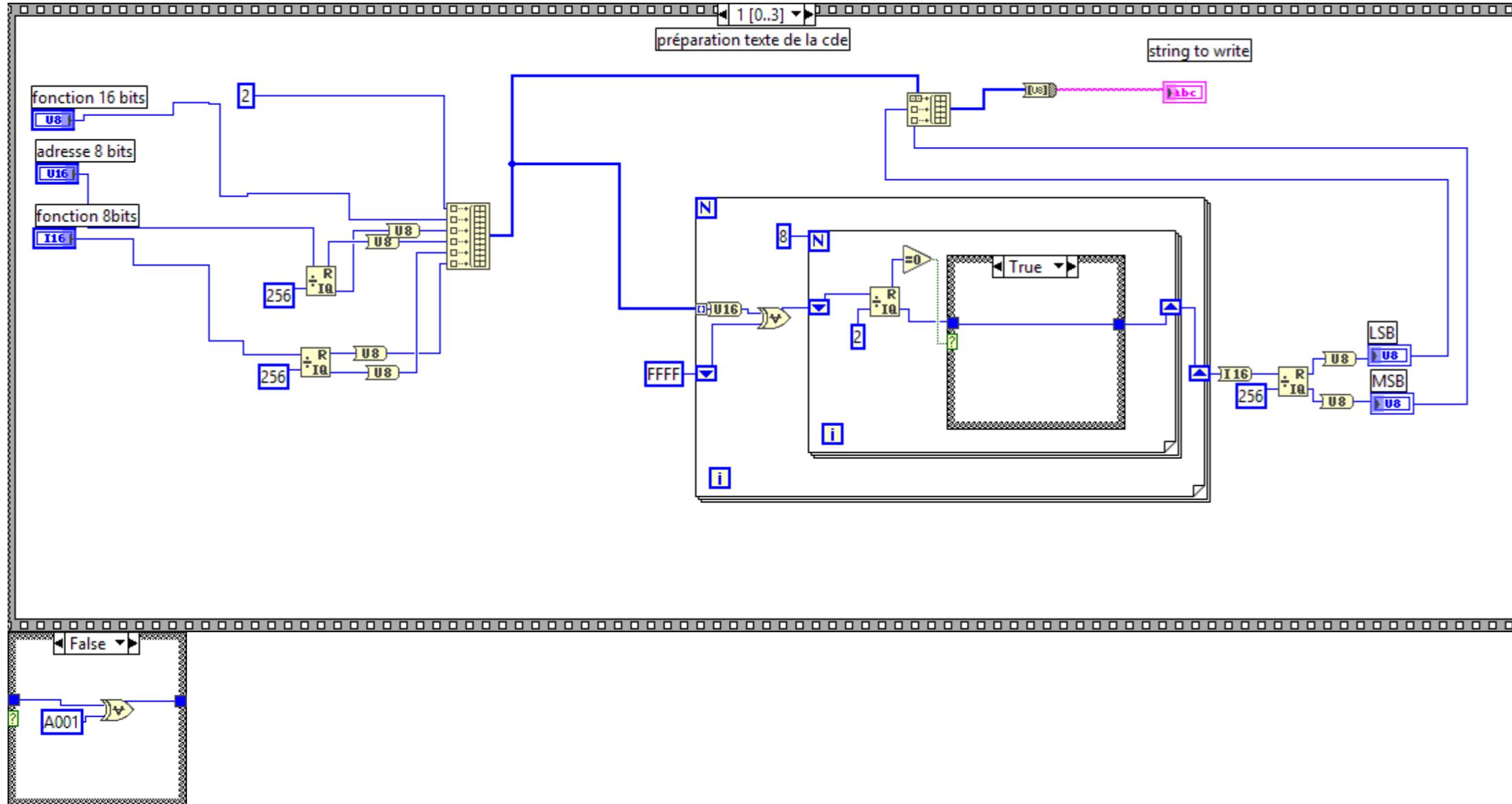
    U16 **Nouvel élément[Nouvel élément]**
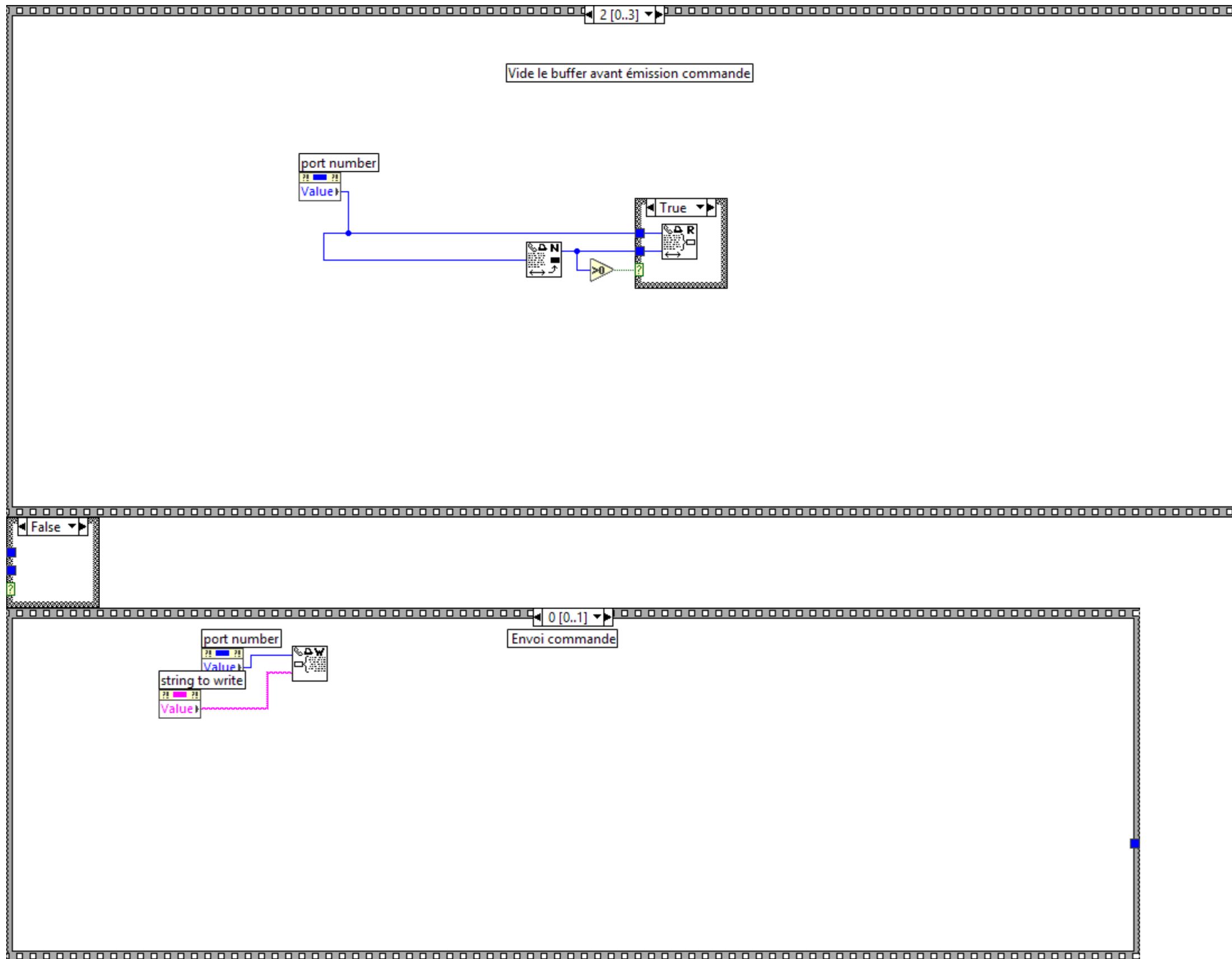
[I16] **valeur paramètres (16 bits) (décimal)[valeur paramètres (16 bits) (décimal)]**

    I16 **[]**

I16 **Erreur**

3 [0..3]

1 [0..1]

0 [0..1]

port number

Value

11

lecture réponse et calcul CRC

error code

I32

string read 3

abc

255

Erreur
I16

8000

7F80

128 7F

7

True

Valeur
EXT

7F

23

0 [0..3]

Initialisation du port série

port number

I32

9600

8

1 bit

no parity

200

SERIAL
PORT

préparation texte de la cde

string to write

fonction 16 bits

U8

adresse 8 bits

U16

fonction 8bits

I16

2

U8

U8

256

R
÷IQ

256

R
÷IQ

U8

U8

[U8]

abc

N

8 N

=0

True

i

i

U16

FFFF

R
÷IQ

2

I16

256

R
÷IQ

U8

U8

LSB

U8

MSB

U8

False

A001

Vide le buffer avant émission commande

port number

Value

True

Envoi commande

port number

Value

string to write

Value

0 [0..1]

2 [0..3]

False

1 [0..1]

tableau d'octets non signés

Value

taille(s)

valeur paramètres
(16 bits)
(décimal)

Tableau

Serial Port Write.vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr\serial.llb\Serial Port
Write.vi
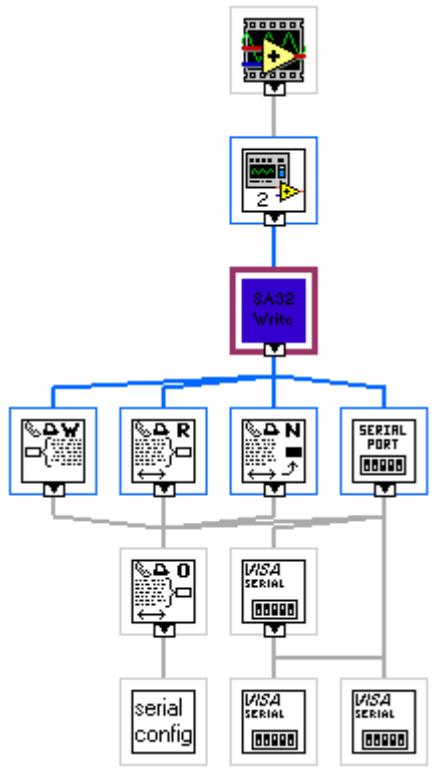
Serial Port Read.vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr\serial.llb\Serial Port
Read.vi

Bytes At Serial Port.vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr\serial.llb\Bytes At Serial
Port.vi

Serial Port Init.vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr\serial.llb\Serial Port
Init.vi

"sa32 write.vi History"
Current Revision: 1
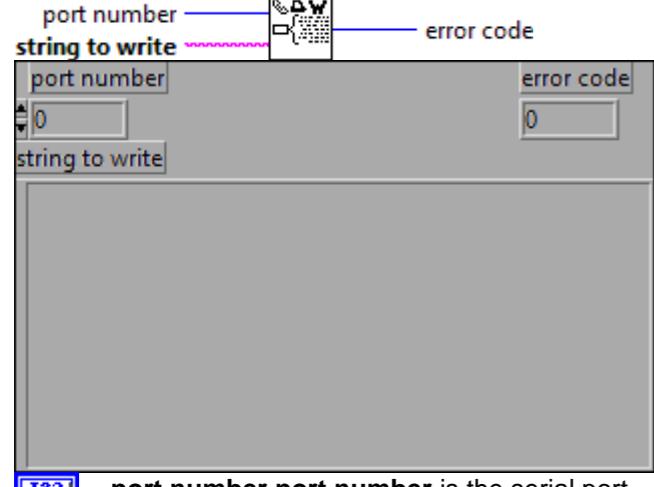
Position in Hierarchy

Iconified Cluster Constants
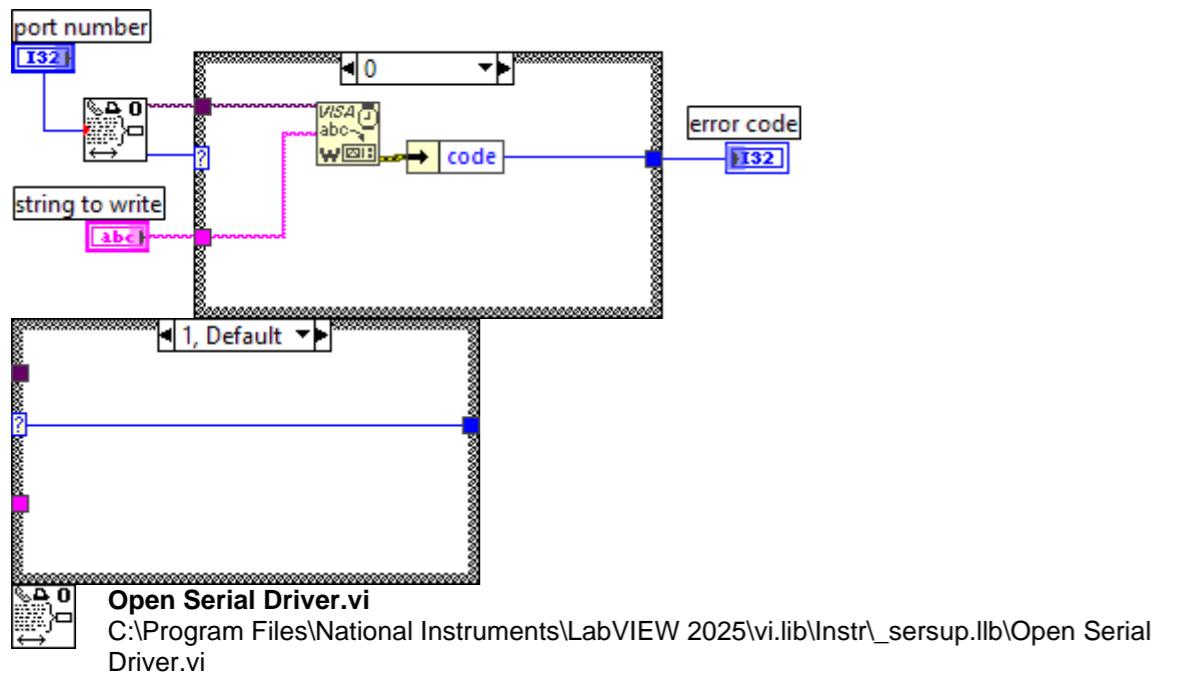
Serial Port Write.vi
Writes the data in **string to write** to the serial port indicated in **port number**.

| I32 | **port number port number** is the serial port. |

| abc | **string to write string to write** is the data to write to the serial port. |

| I32 | **error code error code** is the error or warning code. If **error code** is nonzero, an error occurred. You can wire **error code** to one of the error handler VIs, which describe the error and give you options on how to proceed when an error occurs. |

port number

I32

string to write

abc

0

error code

I32

code

1, Default

Open Serial Driver.vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr_sersup.llb\Open Serial Driver.vi
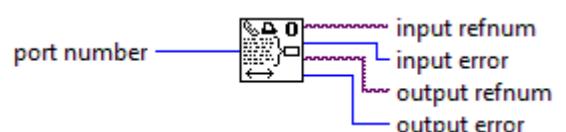
"Serial Port Write.vi History"
Current Revision: 24
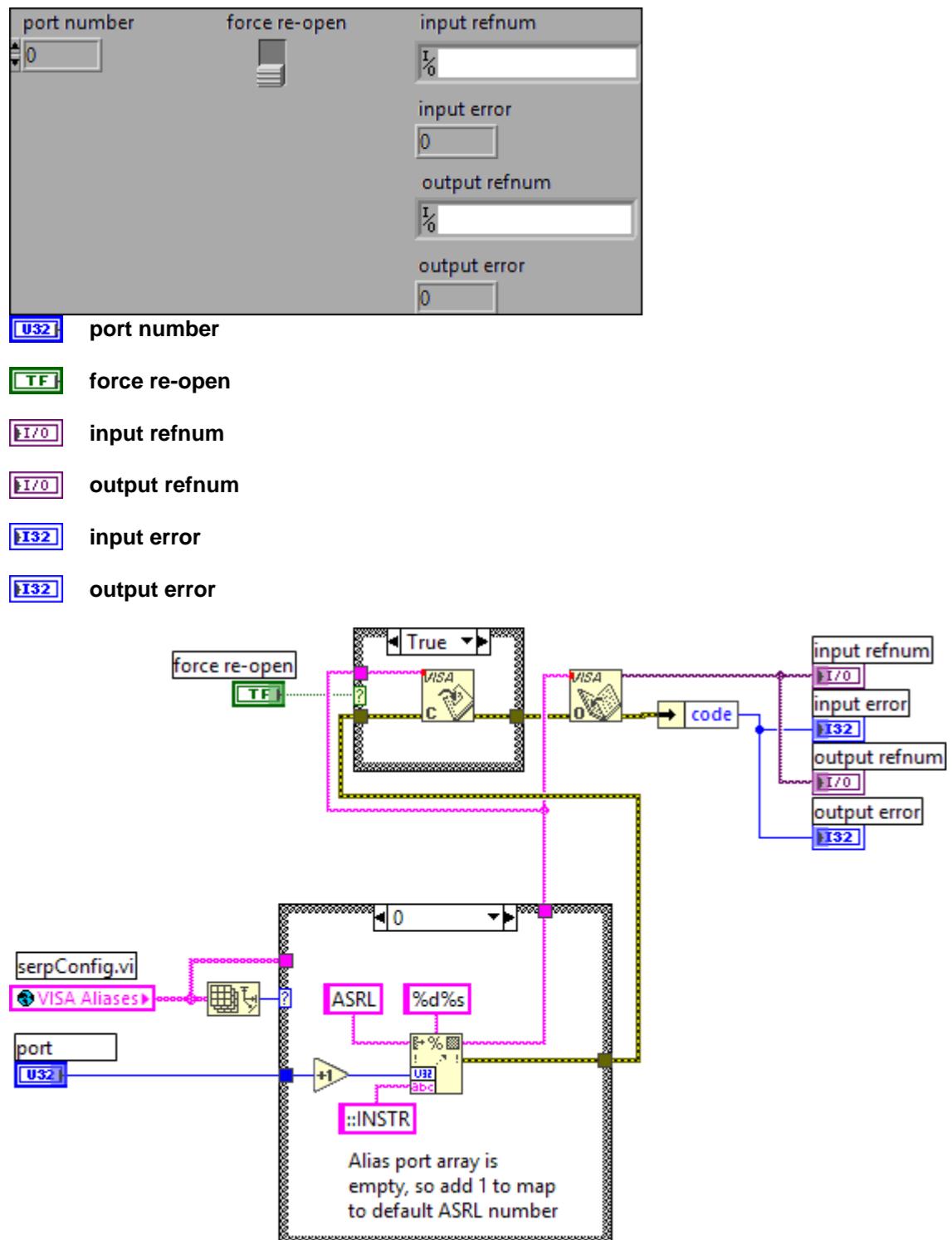
Position in Hierarchy
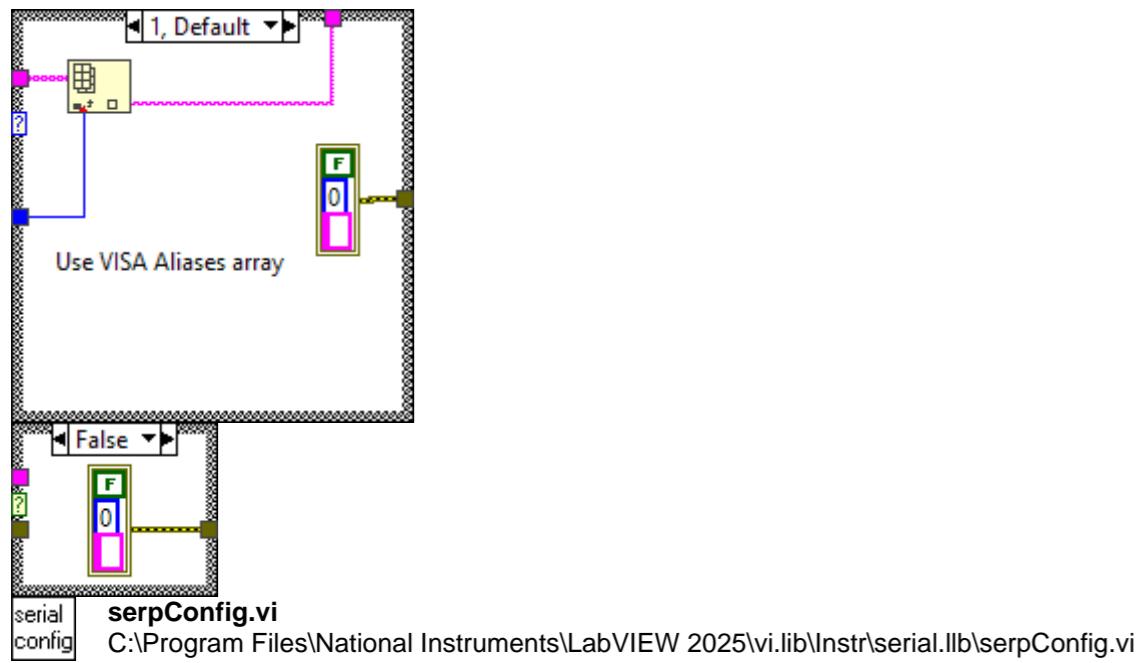
serial
config

Iconified Cluster Constants

Open Serial Driver.vi

port number

input refnum
input error
output refnum
output error

port number	force re-open	input refnum
0		I/0

input error

0

output refnum

I/0

output error

0

U32	port number
TF	force re-open
I/0	input refnum
I/0	output refnum
I32	input error
I32	output error

force re-open

TF

True

VISA
C

VISA
0

code

input refnum

I/0

input error

I32

output refnum

I/0

output error

I32

serpConfig.vi

VISA Aliases

0

ASRL

%d%s

port

U32

+1

::INSTR

Alias port array is
empty, so add 1 to map
to default ASRL number

serpConfig.vi
serial config

serpConfig.vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr\serial.llb\serpConfig.vi

"Open Serial Driver.vi History"
Current Revision: 17

Position in Hierarchy
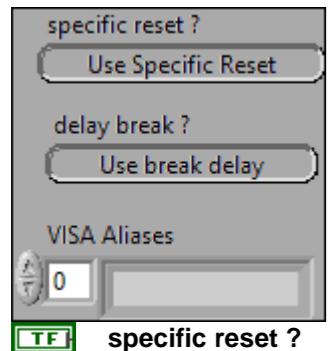
Iconified Cluster Constants

serpConfig.vi
This VI was removed from the Functions palette. Use the VIs and functions on the VISA palette to control serial devices. If you use the LabVIEW PDA Module, use this VI for serial communication.

serial config

specific reset ?
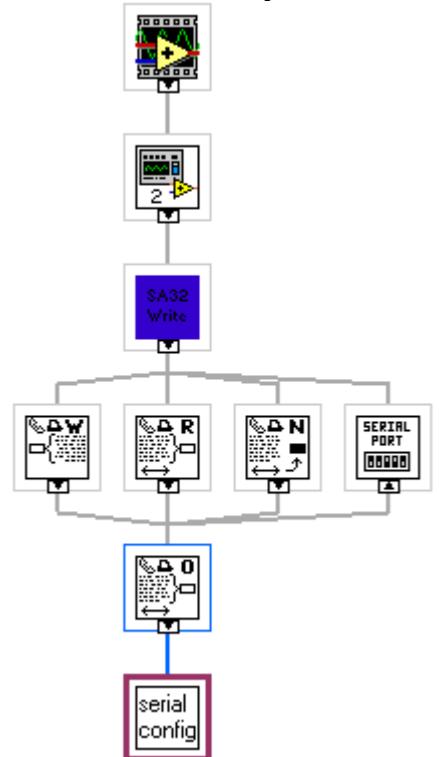
delay break ?

VISA Aliases

String

"serpConfig.vi History"
Current Revision: 5
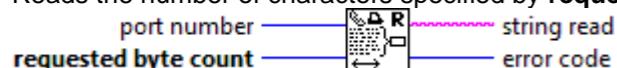
Position in Hierarchy
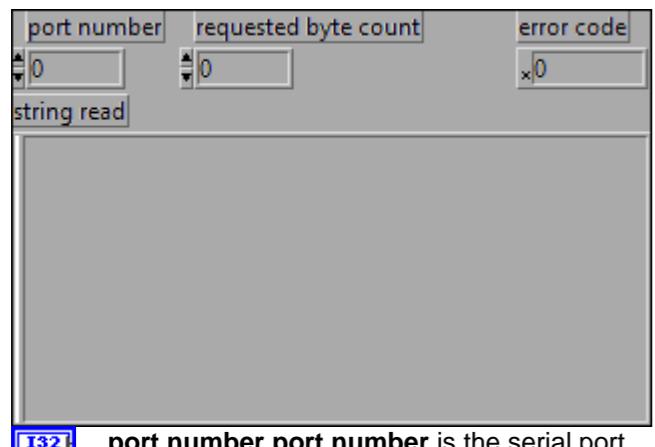
Iconified Cluster Constants

Serial Port Read.vi
Reads the number of characters specified by **requested byte count** from the serial port indicated in **port number**.

port number ———— string read
requested byte count ———— error code

port number	requested byte count		error code
0	0		0
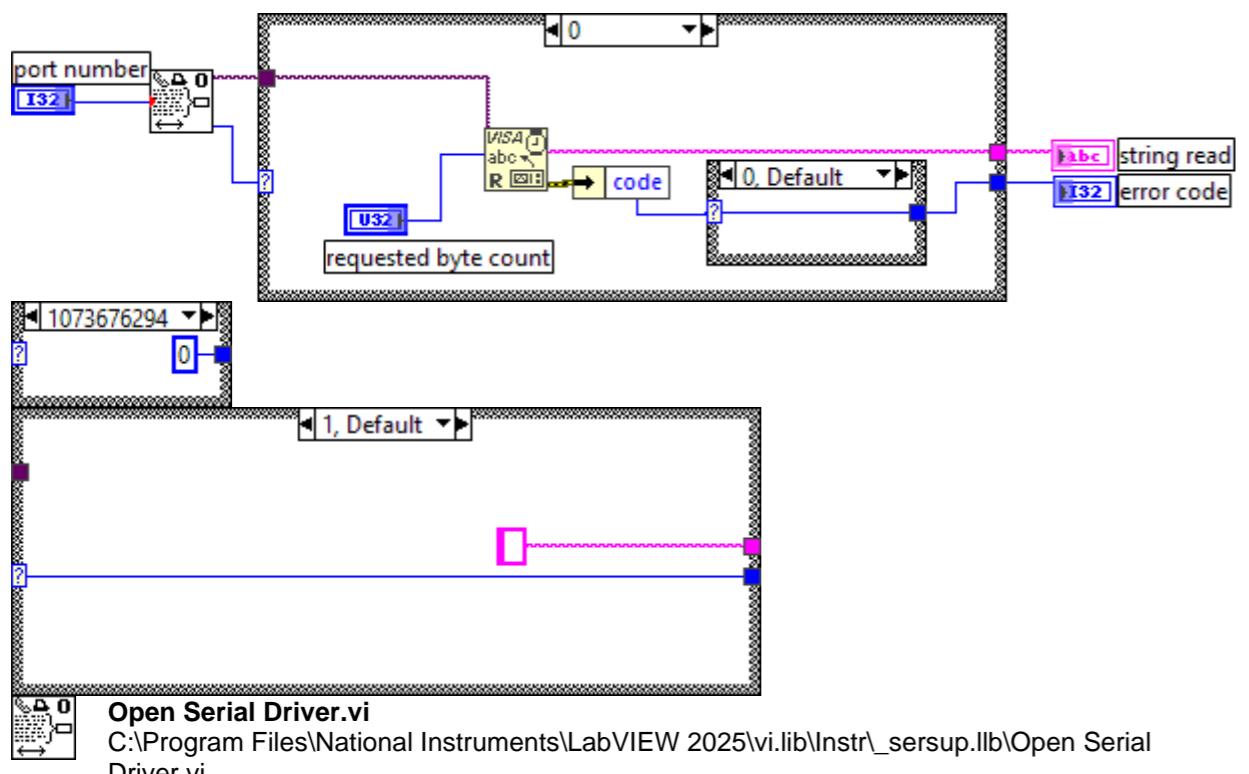
string read

I32 **port number** port number is the serial port.

U32 **requested byte count** requested byte count specifies the number of characters to read.

abc **string read** string read is the bytes read from the serial port.
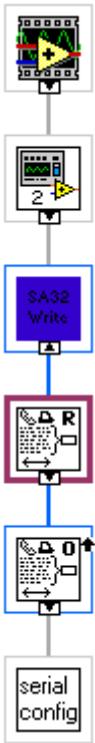
I32 **error code** error code is the error or warning code. If **error code** is nonzero, an error occurred. You can wire **error code** to one of the error handler VIs, which describe the error and give you options on how to proceed when an error occurs.

Open Serial Driver.vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr_sersup.llb\Open Serial Driver.vi

"Serial Port Read.vi History"
Current Revision: 24
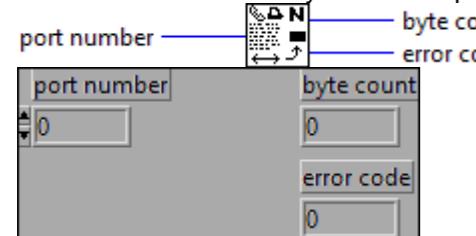
Position in Hierarchy

Iconified Cluster Constants
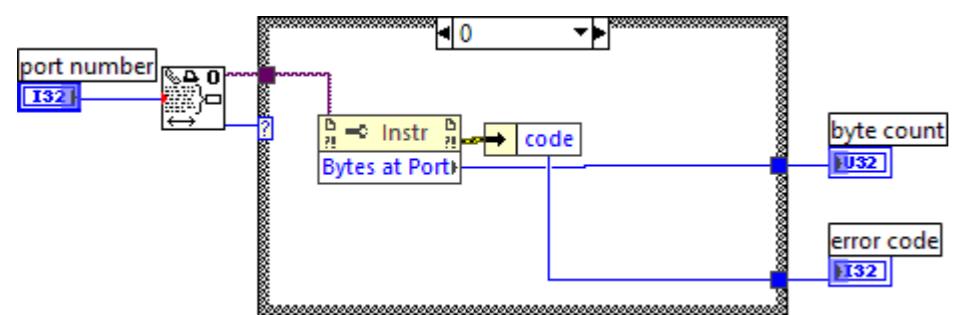
Bytes At Serial Port.vi
Returns the number of bytes in the input buffer of the serial port indicated in **port number**.
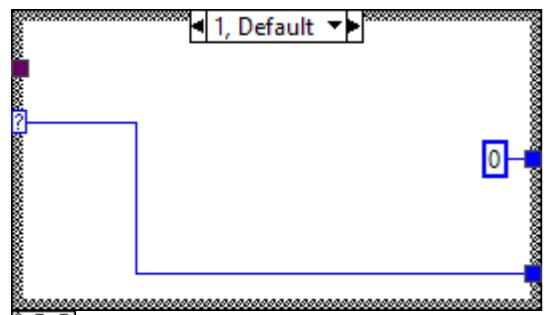
`I32` **port number port number** is the serial port.

`U32` **byte count byte count** is the number of bytes currently in the serial port buffer.

`I32` **error code error code** is the error or warning code. If **error code** is nonzero, an error occurred. You can wire **error code** to one of the error handler VIs, which describe the error and give you options on how to proceed when an error occurs.

Open Serial Driver.vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr_sersup.llb\Open Serial Driver.vi

"Bytes At Serial Port.vi History"
Current Revision: 28
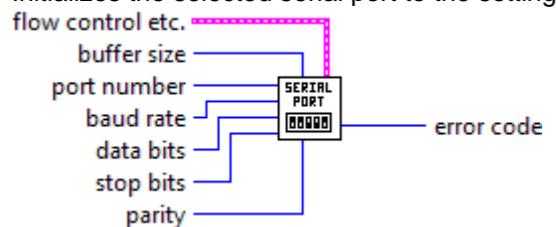
Position in Hierarchy
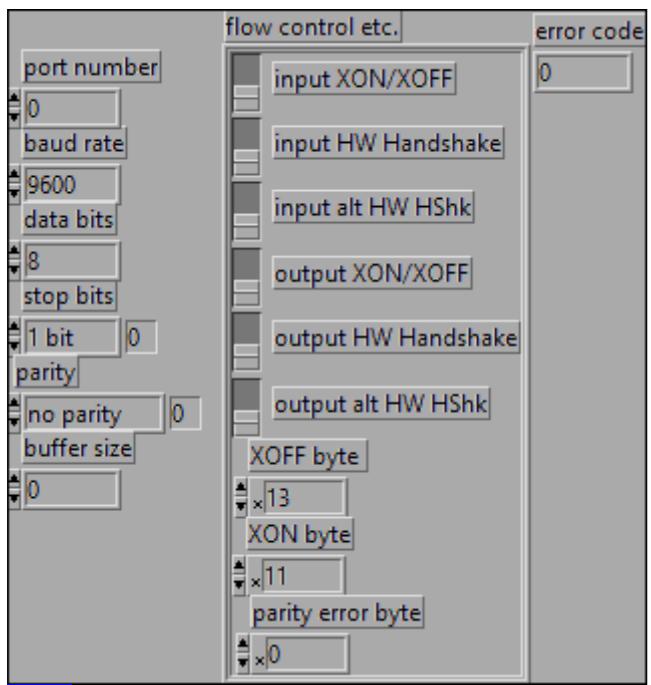
Iconified Cluster Constants

Serial Port Init.vi
Initializes the selected serial port to the settings you specify.

	flow control etc.	error code
port number	input XON/XOFF	0
0	input HW Handshake	
baud rate	input alt HW HShk	
9600	output XON/XOFF	
data bits		
8	output HW Handshake	
stop bits	output alt HW HShk	
1 bit 0	XOFF byte	
parity	×13	
no parity 0	XON byte	
buffer size	×11	
0	parity error byte	
	×0	

U32 **baud rate baud rate** is the rate of transmission. The default is 9600.

⟨⟩ **stop bits stop bits** specifies the number of stop bits used to indicate the end of a frame.

⟨⟩ **parity parity** specifies the parity used to transmit or receive every frame.

U16 **data bits data bits** is the number of bits in the incoming data and can be between 5 and 8. The default is 8.

U16 **buffer size buffer size** indicates the size of the input and output buffers the VI allocates for communication through the specified port.

I32 **port number port number** is the serial port.

⊞ **flow control etc. flow control etc.** enables you to set options for the serial port.

    TF **input XON/XOFF input XON/XOFF** is a software handshaking protocol you can use to avoid overflowing the serial port buffers. The default is FALSE.

    TF **input HW Handshake input HW Handshake** corresponds to Request To Send (RTS) handshaking. The default is FALSE.

    TF **input alt HW HShk input alt HW HShk** corresponds to Data Terminal Ready (DTR) handshaking. The default is FALSE.

    TF **output XON/XOFF output XON/XOFF** is a software handshaking protocol you can use to avoid overflowing the serial port buffers. The default is FALSE.

   **TF** **output HW Handshake output HW Handshake** corresponds to Clear to Send (CTS) handshaking. The default is FALSE.
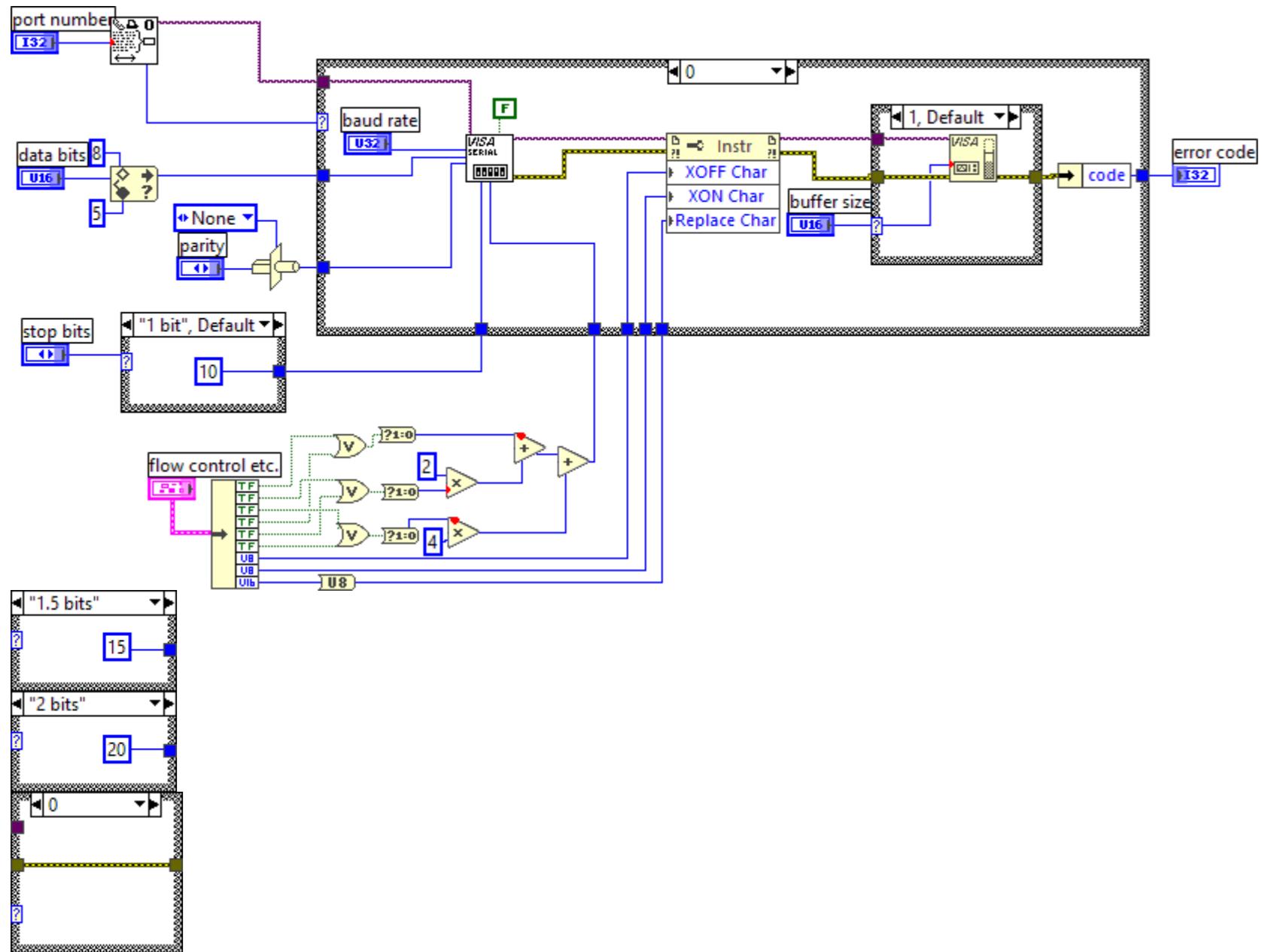
    TF **output alt HW HShk output alt HW HShk** corresponds to Data Set Ready (DSR) handshaking. The default is FALSE.
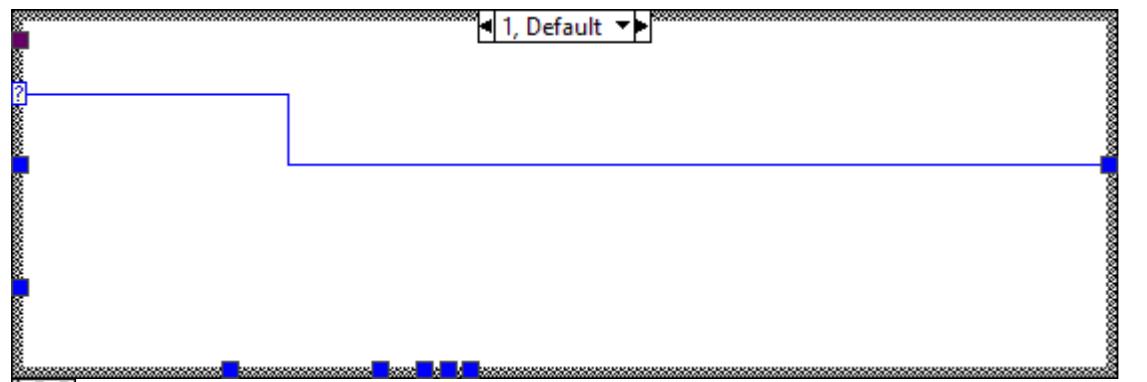
    U8 **XOFF byte   XOFF byte** is the byte used for XOFF (^S). The default is 0x13.

    U8 **XON byte XON byte** is the byte used for XON (^Q). The default is 0x11.

U16 **parity error byte parity error byte** determines the behavior when a parity error occurs. If the high byte is nonzero, the low byte is the character that replaces any parity errors found when you enable parity. The default is 0.

I32 **error code error code** is the error or warning code. **error code** is -1 if **baud rate**, **data bits**, **stop bits**, **parity**, or **port number** are out of range or if the VI could not initialize the serial port.

1, Default
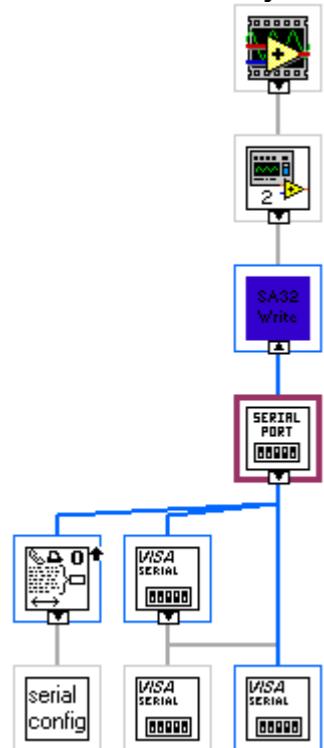
Open Serial Driver.vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr_sersup.llb\Open Serial
Driver.vi

VISA Configure Serial Port (Instr).vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr_visa.llb\VISA Configure
Serial Port (Instr).vi

VISA Configure Serial Port
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr_visa.llb\VISA Configure
Serial Port

"Serial Port Init.vi History"
Current Revision: 47
rev. 6 lun. 24 févr. 1997 18:01:30 steve
The baud rate parameter has been changed to uInt32 for fast ports.   This may still be restricted on some platforms by lower level limitations.

Position in Hierarchy

Iconified Cluster Constants

VISA Configure Serial Port (Instr).vi
<p>Initializes the serial port specified by VISA resource name to the specified settings. Wire data to the VISA resource name input to determine the polymorphic instance to use or manually select the instance.</p>

VISA resource name **VISA resource name** specifies the resource to be opened. The VISA resource name control also specifies the session and class.

timeout (10sec) **timeout** specifies the time, in milliseconds, for the write and read operations.

baud rate (9600) **baud rate** is the rate of transmission.

data bits (8) **data bits** is the number of bits in the incoming data.

parity (0:none) **parity** specifies the parity used for every frame to be transmitted or received.

termination char (0xA = '\n' = LF) **termination char** calls for termination of the read operation. The read operation terminates when the **termination char** is read from the serial device.

error in (no error) **error in** describes error conditions that occur before this node runs. This input provides st andard error in functionality.

status status is TRUE (X) if an error occurred before this node ran or FALSE (checkmark) to indicate a warning or that no error occurred before this node ran. The default is FALSE.

code code is the error or warning code. The default is 0.

source source specifies the origin of the error or warning and is, in most cases,

the name of the node that produced the error or warning. The default is an empty string.

U16 **stop bits (10: 1 bit)** stop bits specifies the number of stop bits used to indicate the end of a frame.

TF **Enable Termination Char (T)** Enable Termination Char prepares the serial device to recognize termination char.

**U16** **flow control (0:none)** flow control sets the type of control used by the transfer mechanism.

error out error out contains error information. This output provides standard error out functionality.

> **TF** **status status** is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

> **I32** **code code** is the error or warning code.

> **abc** **source source** describes the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning.

I/O **VISA resource name out** VISA resource name out is a copy of the VISA resource name that VISA functions return.

"VISA Configure Serial Port (Instr).vi History"
Current Revision: 113

Position in Hierarchy

Iconified Cluster Constants

VISA Configure Serial Port

<p>Initializes the serial port specified by VISA resource name to the specified settings. Wire data to the VISA resource name input to determine the polymorphic instance to use or <a href="nihelplauncher://docs/csh?context=lvcore_lvhowto_selectingdefaultinstpolyvi">manually select the instance.</p>

VISA Configure Serial Port (Instr).vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr_visa.llb\VISA Configure Serial Port (Instr).vi
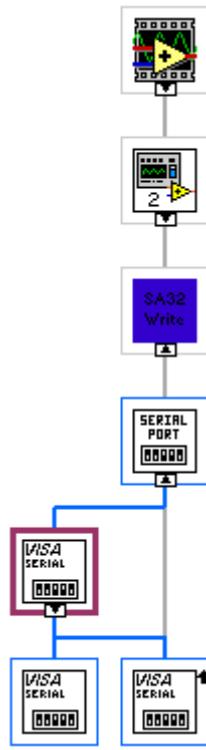
VISA Configure Serial Port (Serial Instr).vi
C:\Program Files\National Instruments\LabVIEW 2025\vi.lib\Instr_visa.llb\VISA Configure Serial Port (Serial Instr).vi

"VISA Configure Serial Port History"
Current Revision: 23
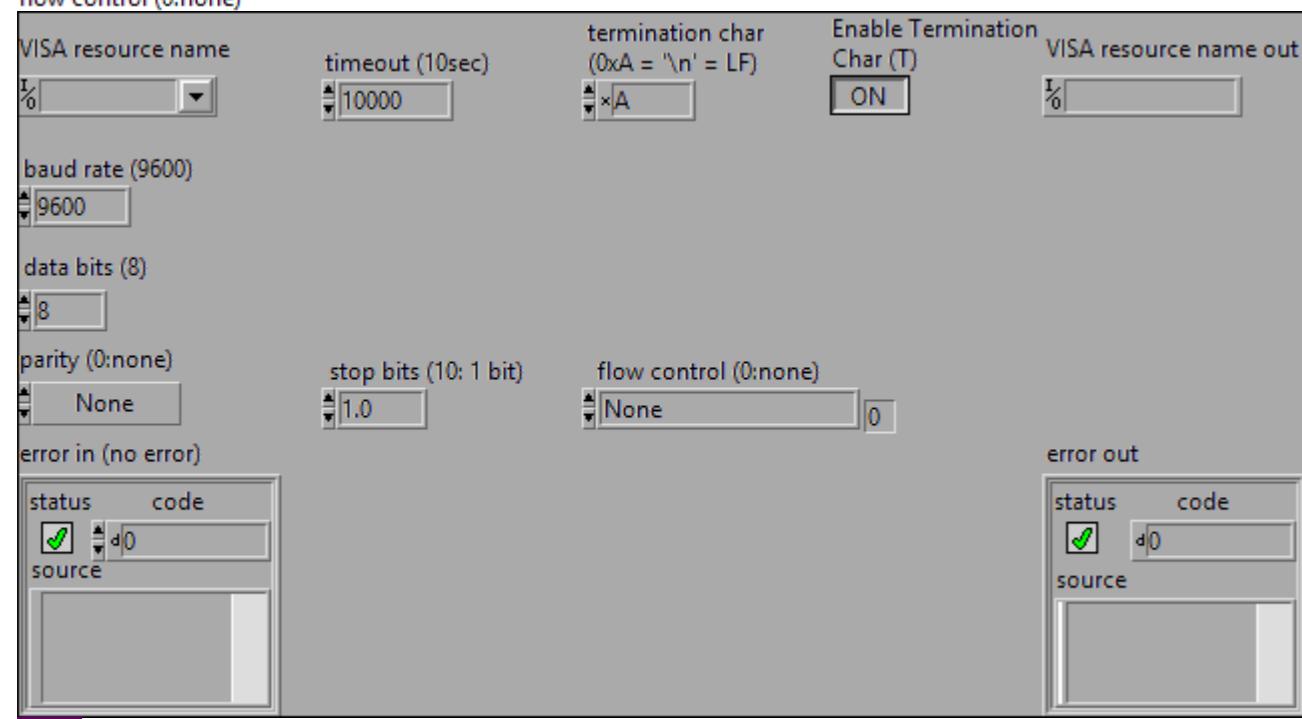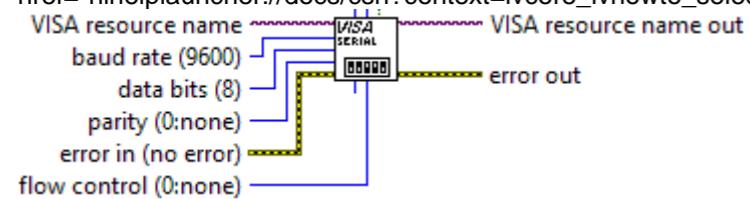
Position in Hierarchy

Iconified Cluster Constants

VISA Configure Serial Port (Serial Instr).vi
<p>Initializes the serial port specified by VISA resource name to the specified settings. Wire data to the VISA resource name input to determine the polymorphic instance to use or manually select the instance.</p>
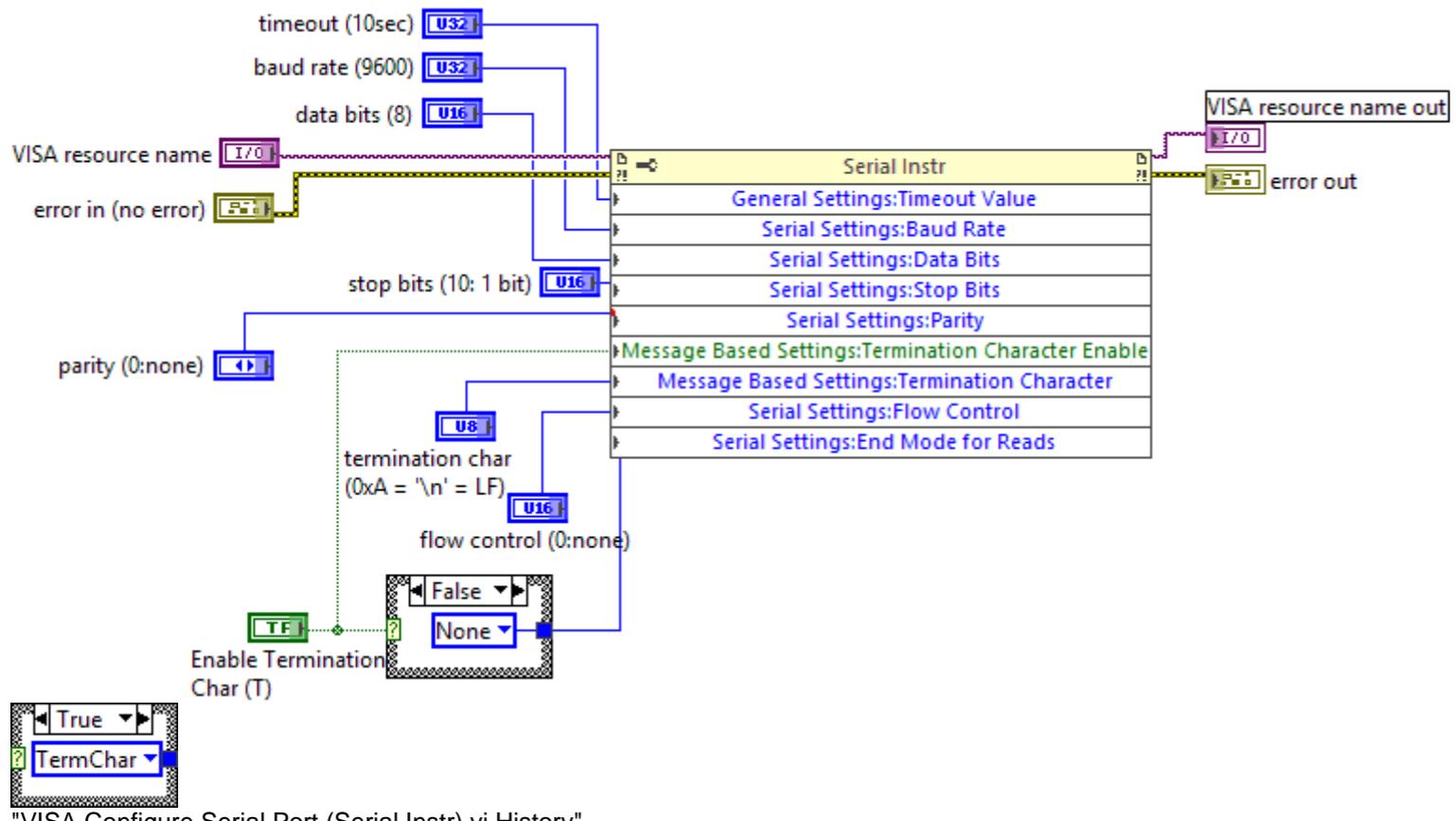
VISA resource name VISA resource name specifies the resource to

be opened. The VISA resource name control also specifies the session and class.

U32 **timeout (10sec)** timeout specifies the time, in milliseconds, for the write and read operations.

U32 **baud rate (9600)** baud rate is the rate of transmission.

U16 **data bits (8)** data bits is the number of bits in the incoming data.

◄► **parity (0:none)** parity specifies the parity used for every frame to be transmitted or received.

U8 **termination char (0xA = '\n' = LF)** termination char calls for termination of the read operation. The read operation terminates when the termination char is read from the serial device.

error in (no error) error in describes error conditions that occur before this node runs. This input provides standard error in functionality.

TF **status** status is TRUE (X) if an error occurred before this node ran or FALSE (checkmark) to indicate a warning or that no error occurred before this node ran. The default is FALSE.

I32 **code** code is the error or warning code. The default is 0.

abc **source** source specifies the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning. The default is an empty string.

U16 **stop bits (10: 1 bit)** stop bits specifies the number of stop bits used to indicate the end of a frame.

TF **Enable Termination Char (T)** Enable Termination Char prepares the serial device to recognize termination char.

U16 **flow control (0:none)** flow control sets the type of control used by the transfer mechanism.

error out error out contains error information. This output provides standard error out functionality.

TF **status** status is TRUE (X) if an error occurred or FALSE (checkmark) to indicate a warning or that no error occurred.

I32 **code** code is the error or warning code.

abc **source** source describes the origin of the error or warning and is, in most cases, the name of the node that produced the error or warning.

I/O **VISA resource name out** VISA resource name out is a copy of the VISA resource name that VISA functions return.

timeout (10sec) `U32`

baud rate (9600) `U32`

data bits (8) `U16`

VISA resource name `I/O`

error in (no error)

stop bits (10: 1 bit) `U16`

parity (0:none)

termination char
(0xA = '\n' = LF) `U8`

flow control (0:none) `U16`

VISA resource name out `I/O`

error out

Serial Instr

General Settings:Timeout Value
Serial Settings:Baud Rate
Serial Settings:Data Bits
Serial Settings:Stop Bits
Serial Settings:Parity
Message Based Settings:Termination Character Enable
Message Based Settings:Termination Character
Serial Settings:Flow Control
Serial Settings:End Mode for Reads

◄ False ►

None ►

Enable Termination
Char (T) `TF`

◄ True ►

TermChar ►

"VISA Configure Serial Port (Serial Instr).vi History"
Current Revision: 115

Position in Hierarchy

Iconified Cluster Constants