

# VisualCard Control (Version 1.5.0)

## Information

File Name	VisualCard.OCX
Prog ID	SMJ.VisualCard.Control

## Properties

<i>Name</i>	<i>Type</i>	<i>Access</i>
Claimed	Boolean	Read
Result	Long	Read
Version	Long	Read
State	Long	Read
Description	String	Read
CommunicationResult	Long	Read
CommunicationID	Long	Read
Command	Long	Read
Status	Long	Read
TxString	String	Read
RxString	String	Read

## Methods

<i>Name</i>
Open
Close
Claim
Release
Communication
ClearCommunication

## Events

<i>Name</i>
CommunicationEvent

## Claimed Property

Syntax	BOOL Claimed;
Remarks	If TRUE, the device is claimed for the application. If FALSE, the device is released for other applications.

The application must be claimed before the Control will allow access to the methods and properties, and events.

This property is initialized to FALSE by the **Open** method.

## Result Property

Syntax	LONG Result;
Remarks	This property is set by each method. It is also set when a writable property is set.

<i>Value</i>	<i>Meaning</i>
SMJ_SUCCESS	Successful operation.
SMJ_E_CLOSED	Attempt was made to access a closed the Control.
SMJ_E_NOTCLAIMED	Attempt was made to access a not claimed the Control.
SMJ_E_ILLEGAL	Attempt was made to perform an illegal or unsupported operation with the device, or an invalid parameter value was used.
SMJ_E_NOCONTROL	The Control cannot communicate.
SMJ_E_NOHARDWARE	The device is not connecting to the system.
SMJ_E_CLEAR	The Control cleared communication with the device.
SMJ_E_TIMEOUT	The Control was timeout communication with the device.
SMJ_E_BUSY	The Control is communication with the device.
SMJ_E_NOBUSY	The Control is not communication with the device.
SMJ_E_DSR_BUSY	The device is not connecting to the system.

## Version Property

Syntax	LONG Version;
Remarks	Control version number. This property holds the Control version number.

Three version levels are specified, as follows:

### ***Version LevelDescription***

Major	The "millions" place.
Minor	The "thousands" place.
Build	The "units" place.

A sample version number is:

1003027	This value may be displayed as version "1.3.27", and interpreted as major version 1, minor version 3, build 27 of the Control.
---------	--

This property is always readable.

## State Property

Syntax	LONG State;
Remarks	This property is the current state of the Control.

<b><i>Value</i></b>	<b><i>Meaning</i></b>
SMJ_S_CLOSED	The Control is closed.
SMJ_S_IDLE	The Control is in a good state and is not busy.
SMJ_S_BUSY	The Control is in a good state and is busy performing communication.
SMJ_S_ERROR	An error has been reported, and the application must recover the Control to a good state before normal communication can resume.
SMJ_S_CLEAR	The Control cleared communication with the device.
SMJ_S_TIMEOUT	The Control was timeout communication with the device.

This property is always readable.

## Description Property

Syntax      BSTR Description;  
Remarks     String identifying the Control.

The property identifies the Control and any pertinent information about it.

A sample returned string is:  
"SMJ VisualCard Control"

This property is always readable.

## CommunicationResult Property

Syntax      LONG CommunicationResult;  
Remarks     This property is Communication Result with the device.

<i><b>Value</b></i>	<i><b>Meaning</b></i>
SMJ_SUCCESS	Successful communication.
SMJ_E_CLOSED	Attempt was made to access a closed the Control.
SMJ_E_CLEAR	The Control clear communication with the device.
SMJ_E_TIMEOUT	The Control was timeout communication with the device.
SMJ_E_BUSY	The Control is communication with the device.
SMJ_E_DSR_BUSY	The device is not connecting to the system.

## CommunicationID Property

Syntax      LONG CommunicationID;  
Remarks     The Control will fire a CommunicationEvent passing this property as a parameter.

## Command Property

Syntax      LONG Command;  
Remarks     This property is a parameter at the Communication Method request.

## Status Property

Syntax	LONG Status;
Remarks	This property is a status at the Communication Method finish.

## TxString Property

Syntax	BSTR TxString;
Remarks	This property is a parameter at the Communication Method request.

## RxString Property

Syntax	BSTR RxString;
Remarks	This property is a received string at the Communication Method finish.

## Open Method

Syntax      LONG Open (LONG *nPort*, LONG *nBaudRate*, LONG *nByteSize*,  
LONG *nParity*);

The *nPort* parameter gives open to the communication port (IP-address).

<i>Value</i>	<i>Meaning</i>
SMJ_VISUALCARD_PORTCOM1	COM1
SMJ_VISUALCARD_PORTCOM2	COM2
SMJ_VISUALCARD_PORTCOM3	COM3
...	...
SMJ_VISUALCARD_PORTCOM254	COM254
SMJ_VISUALCARD_PORTCOM255	COM255
SMJ_VISUALCARD_PORTCOM256	COM256

The *nBaudRate* parameter gives open to the communication baud rate.

<i>Value</i>	<i>Meaning</i>
SMJ_VISUALCARD_BAUDRATE19200BPS	19200BPS
SMJ_VISUALCARD_BAUDRATE9600BPS	9600BPS
SMJ_VISUALCARD_BAUDRATE4800BPS	4800BPS
SMJ_VISUALCARD_BAUDRATE2400BPS	2400BPS
SMJ_VISUALCARD_BAUDRATE38400BPS	38400BPS
SMJ_VISUALCARD_SOCKET	Socket Communication

The *nByteSize* parameter gives open to the communication byte size.

<i>Value</i>	<i>Meaning</i>
SMJ_VISUALCARD_BYTESIZE8BIT	8BIT
SMJ_VISUALCARD_BYTESIZE7BIT	7BIT
SMJ_VISUALCARD_SOCKET	Socket Communication

The *nParity* parameter gives open to the communication parity.

<i>Value</i>	<i>Meaning</i>
SMJ_VISUALCARD_PARITYNONE	None Parity
SMJ_VISUALCARD_PARITYODD	Odd Parity

SMJ\_VISUALCARD\_PARITYEVEN

Even Parity

SMJ\_VISUALCARD\_SOCKET

Socket Communication

Remarks      This method is open a device for the communication.  
When successful, the many property is initialized.

Return        One of the following values is returned by the method and placed in the **Result** property:

<i>Value</i>	<i>Meaning</i>
SMJ_SUCCESS	Successful operation.
SMJ_E_ILLEGAL	The Control is already open.
SMJ_E_NOCONTROL	Could not establish a connection to the device.
Other Values	See Result.

## Close Method

Syntax        LONG Close ();

Remarks      This method is close a device for the communication.  
If the **Claimed** property is TRUE, then exclusive access to the device is first released.  
When successful, the many property is initialized.

Return        One of the following values is returned by the method and placed in the **Result** property:

<i>Value</i>	<i>Meaning</i>
SMJ_SUCCESS	Successful operation.
Other Values	See Result.

## Claim Method

Syntax      LONG Claim (LONG *nTimeout*);

The *nTimeout* parameter gives the maximum number of milliseconds to wait for exclusive access to be satisfied.

If zero, this method attempts to claim the device, then returns the appropriate status immediately.

Remarks      This method is request exclusive access to the device. Many devices require an application to claim them before they can be used.

When successful, the **Claimed** property is changed to TRUE.

Return      One of the following values is returned by the method and placed in the **Result** property:

<i>Value</i>	<i>Meaning</i>
SMJ_SUCCESS	Successful operation. The <b>Claimed</b> property is TRUE. Also returned if the Control has already claimed the device.
SMJ_E_ILLEGAL	The Control cannot be claimed The <i>nTimeout</i> parameter is invalid.
SMJ_E_TIMEOUT	Another application has exclusive access to the device, and did not relinquish control before <i>nTimeout</i> milliseconds expired.
Other Values	See Result.

## Release Method

Syntax      LONG Release ();

Remarks      This method is release access to the device.

Return      One of the following values is returned by the method and placed in the **Result** property:

<i>Value</i>	<i>Meaning</i>
SMJ_SUCCESS	Successful operation. The Claimed property is now FALSE.
SMJ_E_ILLEGAL	The Control does not have exclusive access to the device.
Other Values	See Result.



## Communication Method

Syntax      LONG Communication (LONG *nCommand*, BSTR *lpzTxString*,  
LONG *nTimeout*, BOOL *bDsrBusy*);

The *nCommand* parameter gives the command code for the communication.

The *lpzTxString* parameter gives the transmit string for the communication.

The *nTimeout* parameter gives the maximum number of milliseconds to wait for communication to be finished.

If zero, this method does not time out.

The *bDsrBusy* parameter the value to check the DSR signal.

If FALSE, this method does not check the DSR signal.

Remarks      This method is communication to the device.

Return      One of the following values is returned by the method and placed in the **Result** property:

<i>Value</i>	<i>Meaning</i>
SMJ_SUCCESS	Successful operation.
SMJ_E_BUSY	The Control does communication to the device.
Other Values	See Result.

## ClearCommunication Method

Syntax      LONG ClearCommunication ();

Remarks      This method is communication clear to the device.

Return      One of the following values is returned by the method and placed in the **Result** property:

<i>Value</i>	<i>Meaning</i>
SMJ_SUCCESS	Successful operation.
SMJ_E_NOBUSY	The Control does not communication to the device.
Other Values	See Result.

## CommunicationEvent Event

Syntax      void CommunicationEvent (LONG nResult, LONG nCommunicationID);

The ***nResult*** parameter is Result Code of the Communication finish.

The ***nCommunicationID*** parameter is ID number of the Communication finish.

Remarks      This Event is Communication finish.