File IO Plug-in

Requires VTS-Connect minimum version **4.0.0.81**

The ***File IO Plug-in*** allows an Expert Advisor to read and write any text file. The exact format of the file may be defined in any manner using the File IO Manager. The file is read from, or written to, using the functions from the Functions Toolbox *fnFileRead* and *fnFileWrite*.

What is a Plug-in?

VTS stands for ***Visual Traders Studio.***

The VTS *Expert Advisor* Builder is a Windows graphical application that enables non-programmers to build complex Expert Advisors by dragging, dropping and connecting logical elements.

The VTS application contains basic functionality to build almost any Expert Advisor.

A ***VTS Plug-in*** allows traders to easily implement advanced trading techniques using an add-on user interface.

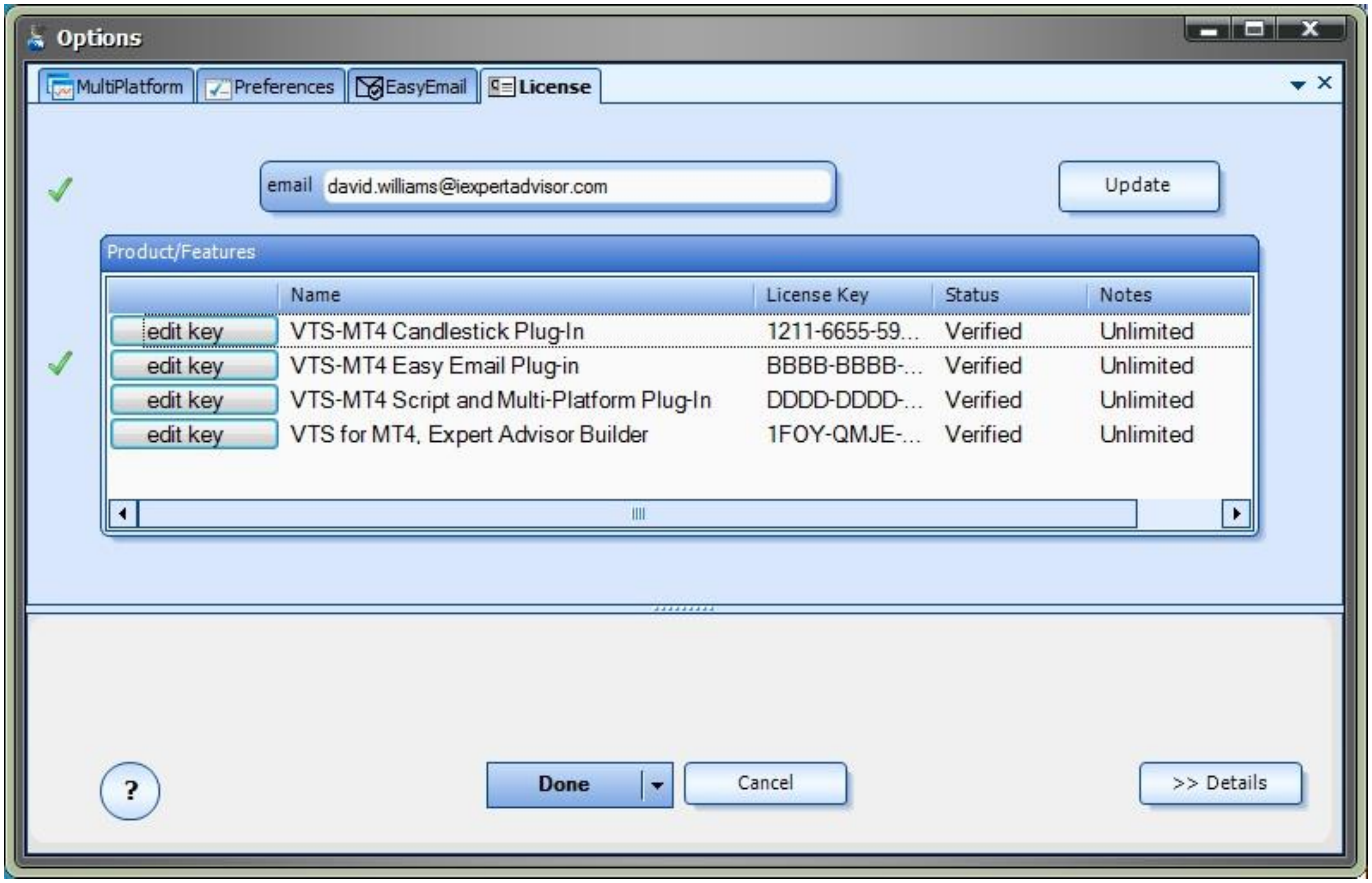
Contents

Enable the *File IO* Plug-in

You must enter your License key to enable the ***File IO Plug-in***.   Your license key for all of your VTS products can be found in the Members Area.

License keys are entered in Visual Traders Studio (VTS) from the License entry tab.

- The **email** address is the email address used to purchase VTS.
- The **License Key** is the key listed in the Members Area.
- The **Update** button is used to verify the email address and license key.
- The **edit key** button is used edit the key value.

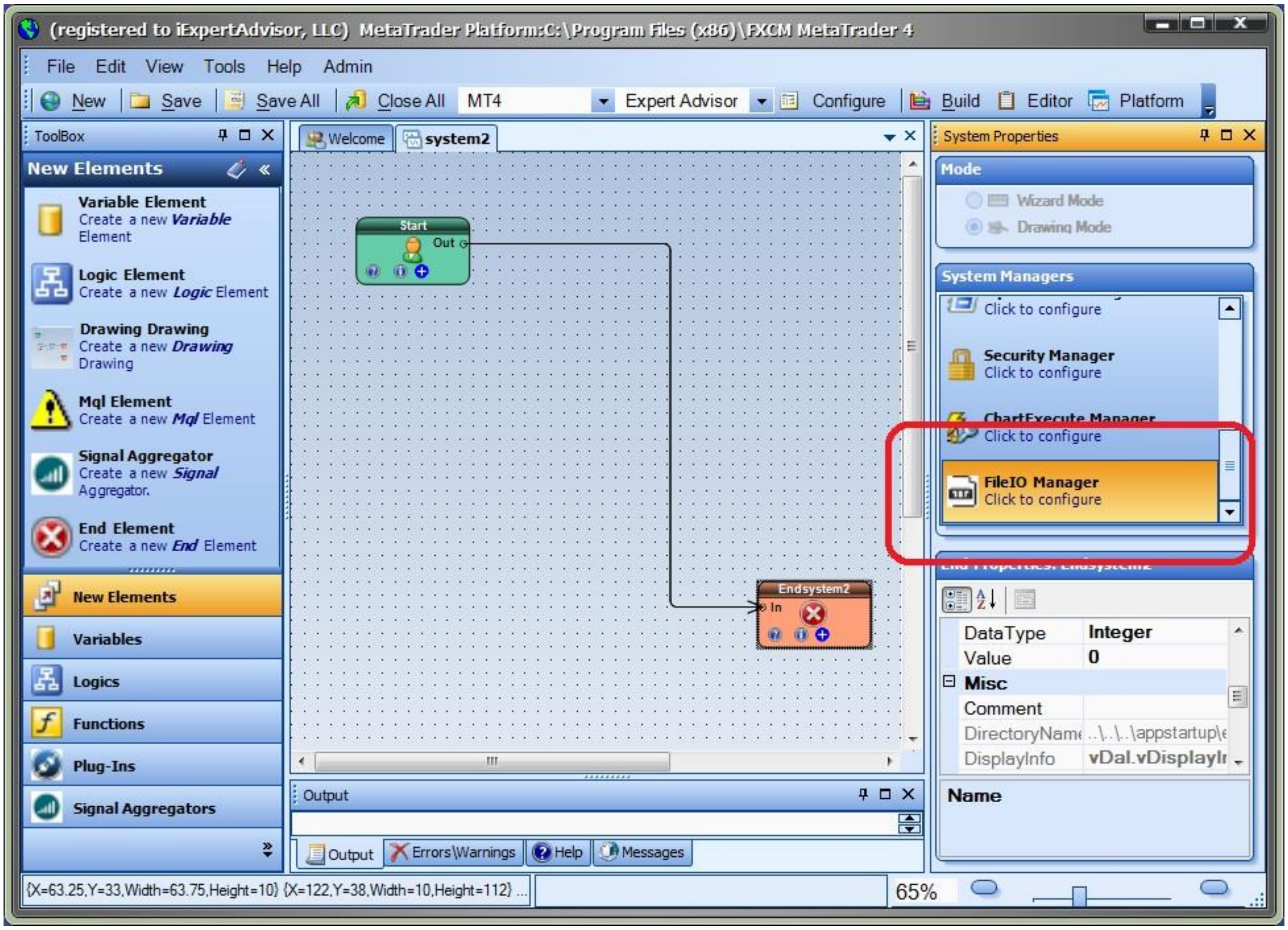


email: david.williams@iexpertadvisor.com

	Name	License Key	Status	Notes
edit key	VTS-MT4 Candlestick Plug-In	1211-6655-59...	Verified	Unlimited
edit key	VTS-MT4 Easy Email Plug-in	BBBB-BBBB-...	Verified	Unlimited
edit key	VTS-MT4 Script and Multi-Platform Plug-In	DDDD-DDDD-...	Verified	Unlimited
edit key	VTS for MT4, Expert Advisor Builder	1FOY-QMJE-...	Verified	Unlimited

File IO in the System Manager

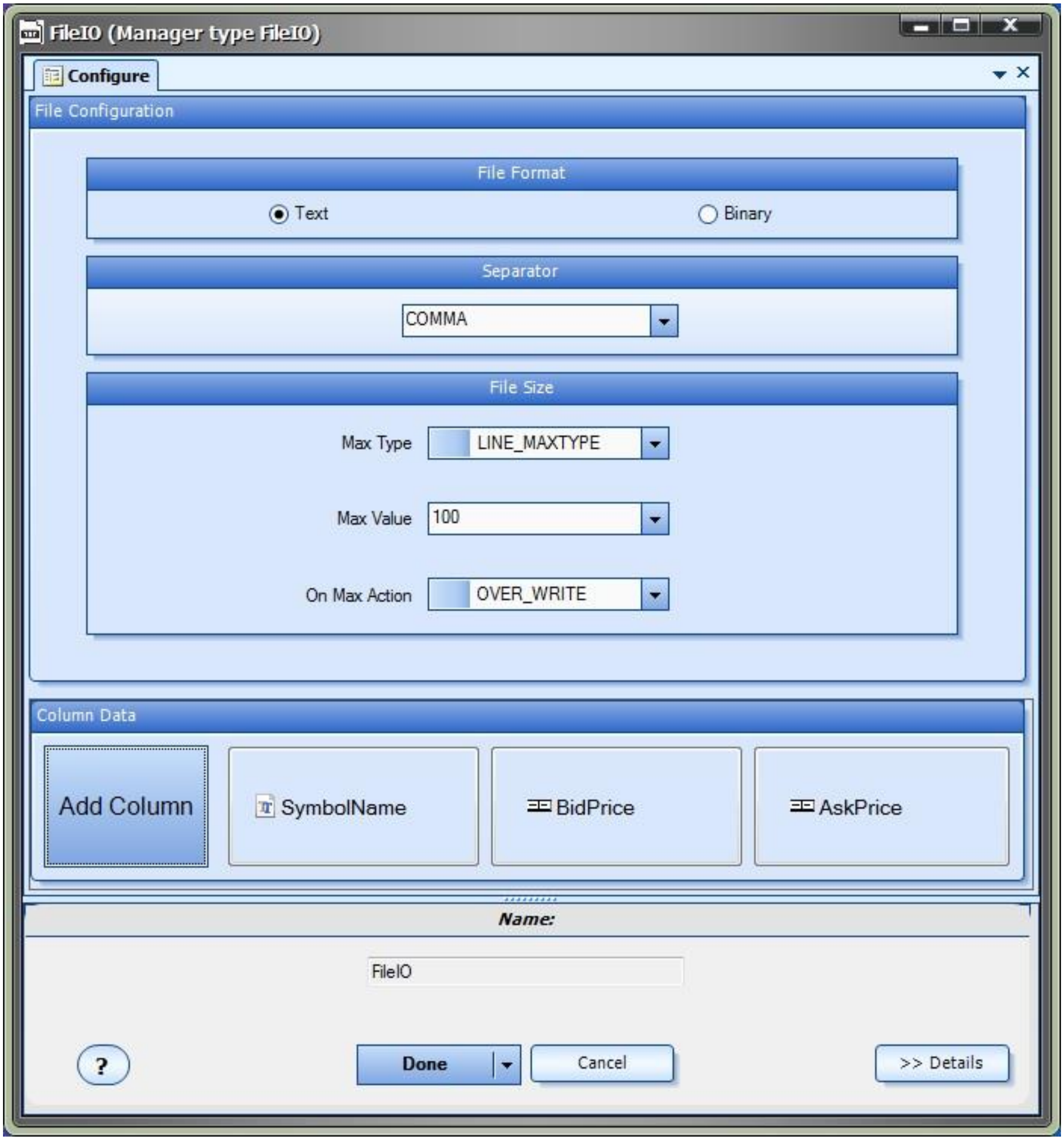
- **File IO** configuration is accessed through the ***File IO*** System Manager.
- System Managers are found on the right side of the VTS main application.
- Depending on your screen resolution, you may need to scroll down to view the ***File IO*** icon.
- To open the ***File IO*** Manager, double-click the icon.


(registered to iExpertAdvisor, LLC) MetaTrader Platform:C:\Program Files (x86)\FXCM MetaTrader 4
File Edit View Tools Help Admin
New Save Save All Close All MT4 Expert Advisor Configure Build Editor Platform
ToolBox
New Elements
Variable Element Create a new Variable Element
Logic Element Create a new Logic Element
Drawing Drawing Create a new Drawing Drawing
Mql Element Create a new Mql Element
Signal Aggregator Create a new Signal Aggregator.
End Element Create a new End Element
New Elements
Variables
Logics
Functions
Plug-Ins
Signal Aggregators
Welcome
system2
Start
Out
Endsystem2
In
Output
Output Errors\Warnings Help Messages
System Properties
Mode
Wizard Mode
Drawing Mode
System Managers
Click to configure
Security Manager Click to configure
ChartExecute Manager Click to configure
FileIO Manager Click to configure
DataType Integer
Value 0
Misc
Comment
DirectoryName ..\..\..\appstartup\
DisplayInfo vDal.vDisplayI
Name
{X=63.25,Y=33,Width=63.75,Height=10} {X=122,Y=38,Width=10,Height=112} ...
65%

File IO Configuration

- **File IO** configuration is accessed through the ***File IO*** System Manager.
- To open the ***File IO*** Manager, double-click the icon.

This screen is used to configure the format of the file that is read or written to using the functions fnFileRead and fnFileWrite.



File Format: (●) Text ( ) Binary

Separator: COMMA

File Size

Max Type: LINE_MAXTYPE

Max Value: 100

On Max Action: OVER_WRITE

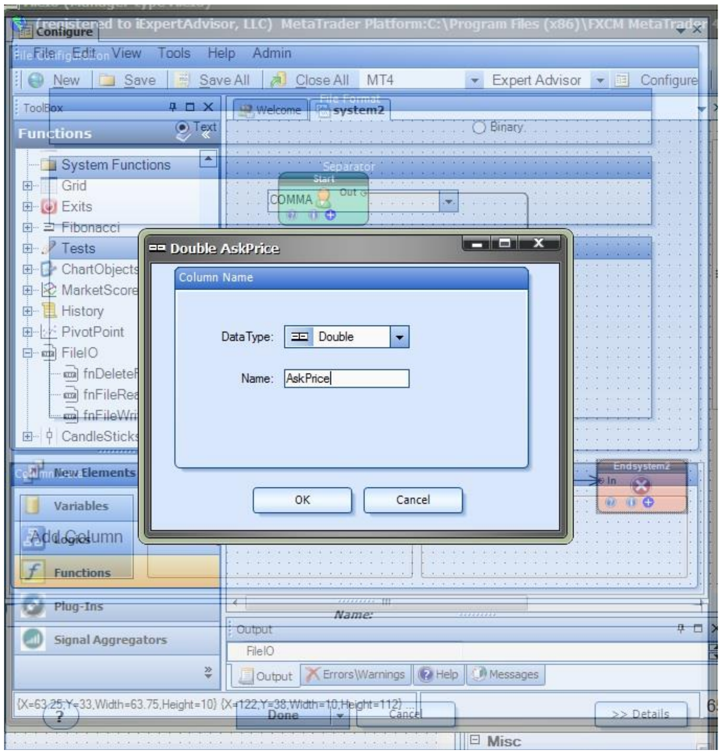
Column Data: SymbolName, BidPrice, AskPrice

Name: FileIO

Name	Options	Description
File format	Text Binary	Simple text file Binary read/write mode (without string to

		string conversion).
Separator	COMMA TAB SPACE PIPE	Character used to separate data values
File Size, Max Type	LINE_MAXTYPE BYTE_MAXTYPE	Mac Value in bytes or lines.
File Size, Max Value	Integer number (0-max)	A value 0 is unlimited.
File Size, Max Action	OVER_WRITE ROLL_OVER	Overwrite the last line of the file. Delete the file and start over.
Add Column	Add a Data column	Add a column of any data type. As columns are added they appear to the left. Columns can be dragged to be rearranged. Columns can be edited by double-clicking. Columns can be deleted by right-clicking

Clicking the "**Add Column**" button displays the following window to create or edit a column:



Data Type: Double

Name: AskPrice

Example:

To create a file that writes the currency symbol, bid and ask price on each tick:

EURUSD,1.1229300000,1.1232600000

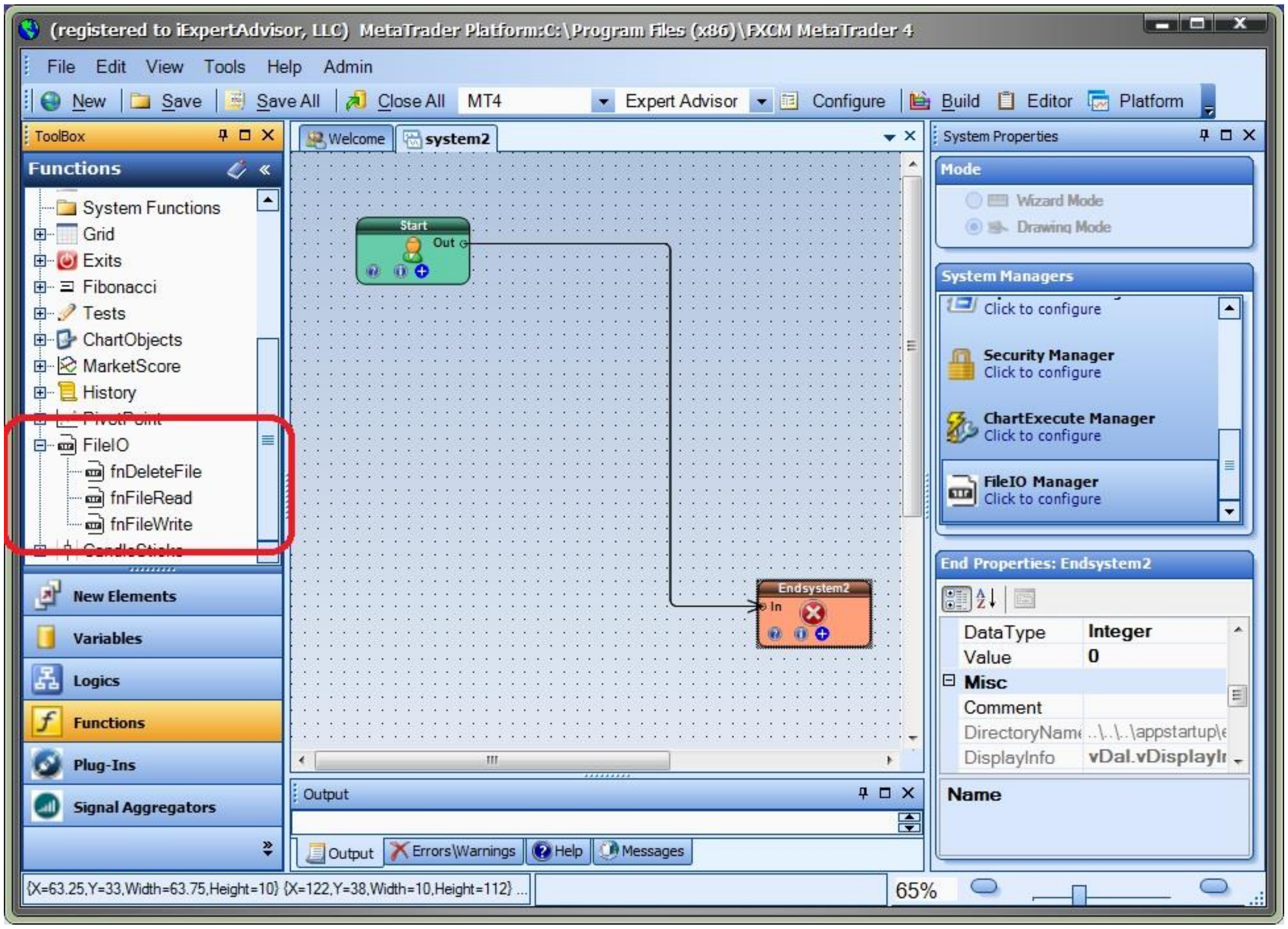
- The first column is a string data type and can be named anything, for example **SymbolName**
- The second column is a double data type and be named anything, for example **BidPrice**
- The third column is a double data type and be named anything, for example **AskPrice**
- The columns are separated by a comma: the Separator value is selected as **COMMA**.

Refer to the sections on fnFileRead and fnFileWrite to read and write lines of data to this file format.

File IO Functions in the Toolbox

Once enabled, the File IO functions are available in the Toolbox Function tab under the ***FileIO*** menu.

These functions are dragged and dropped from the Toolbox onto the Drawing Pad like any other functions.



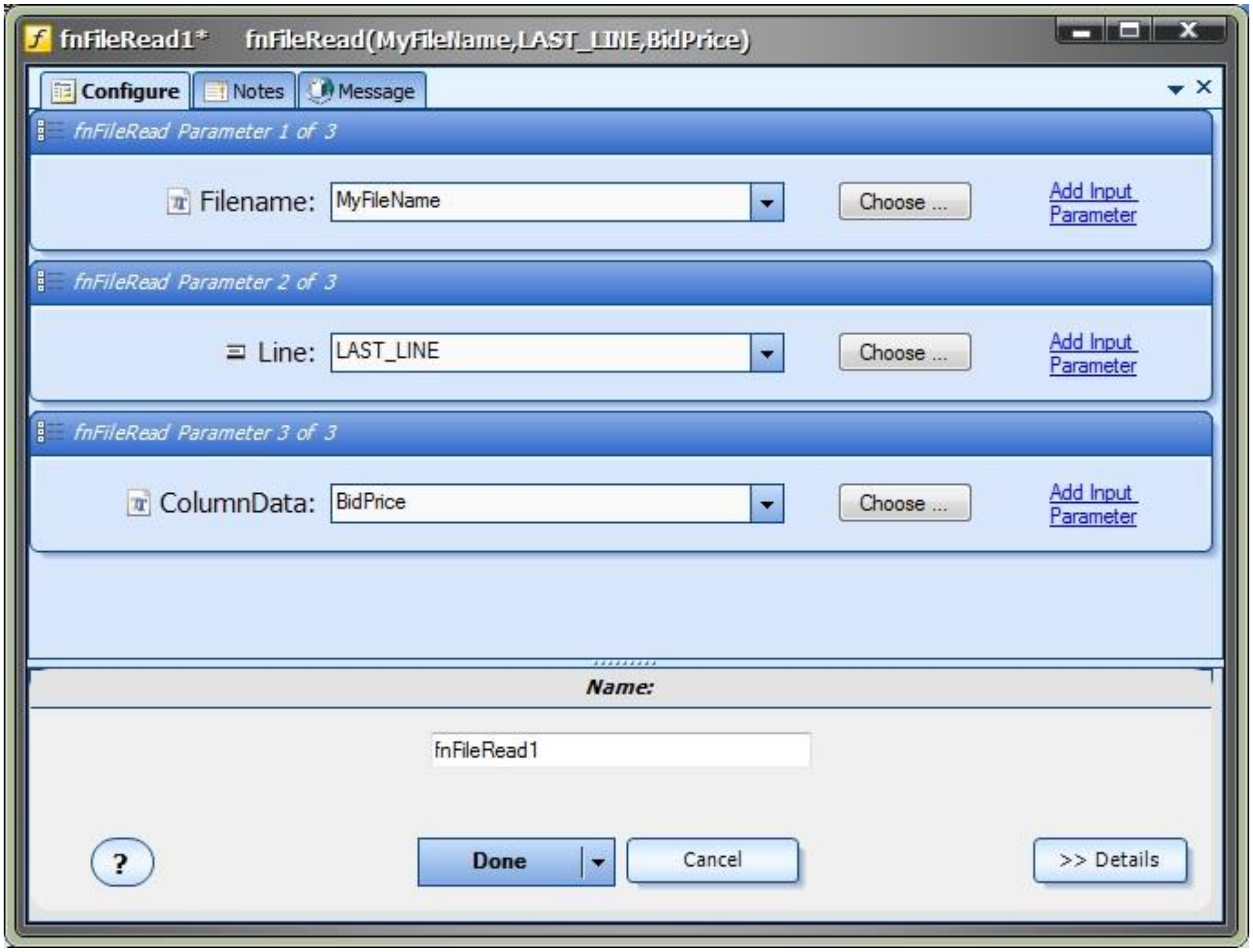
DataType	Integer
Value	0
Misc	
Comment	
DirectoryName	..\..\..\appstartup\e
DisplayInfo	vDal.vDisplayIn

Name

{X=63.25,Y=33,Width=63.75,Height=10} {X=122,Y=38,Width=10,Height=112} ...

fnFileRead

The function ***fnReadFile*** is used to read a single data value from a file. The configuration of the file is defined in the *File IO* System Manager



fnFileRead1* fnFileRead(MyFileName,LAST_LINE,BidPrice)

fnFileRead Parameter 1 of 3

Filename: MyFileName

fnFileRead Parameter 2 of 3

Line: LAST_LINE

fnFileRead Parameter 3 of 3

ColumnData: BidPrice

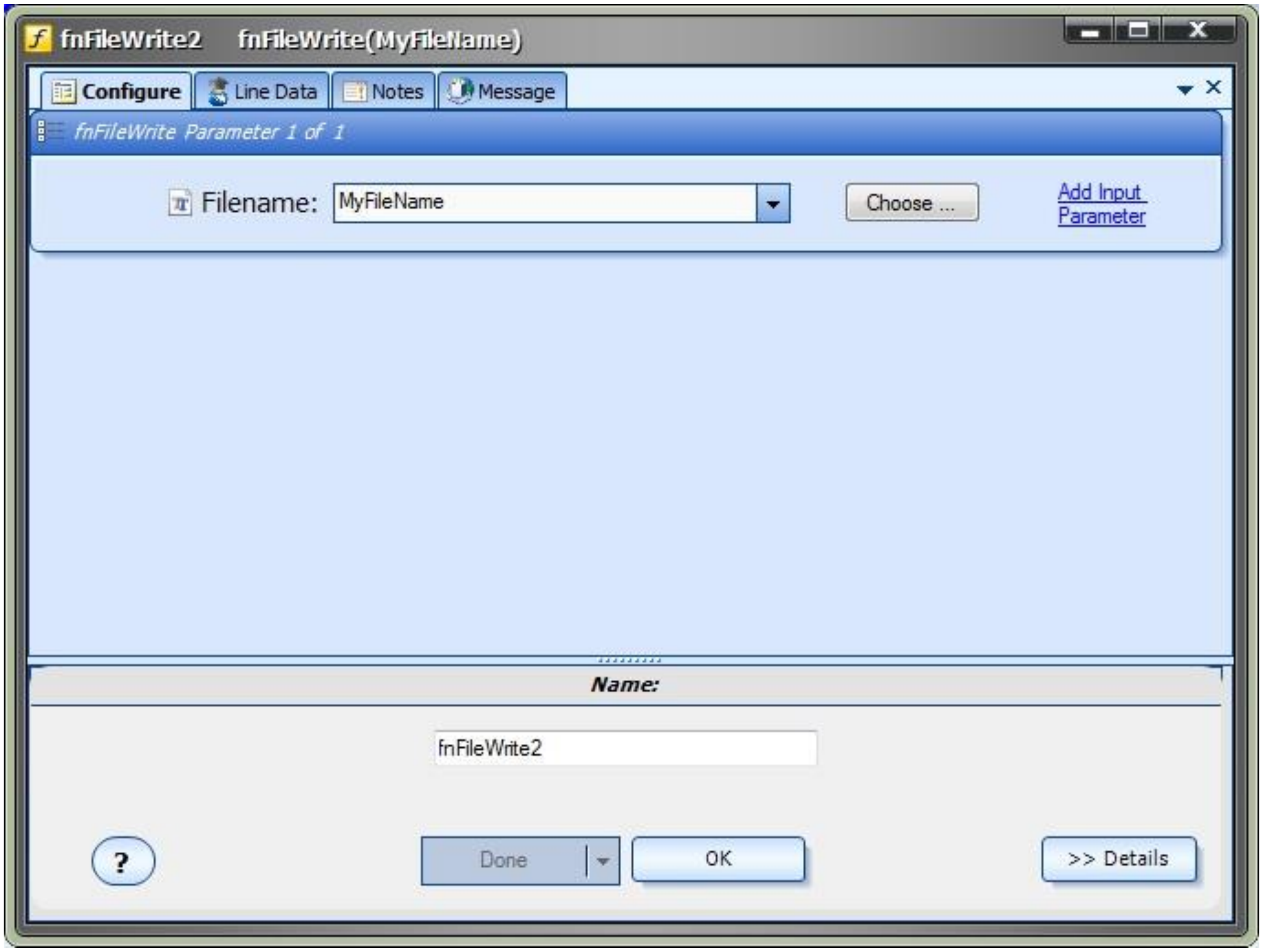
Name: fnFileRead1

Parameter Name	**Data type**	**Description**
FileName	string	The name of the file to read. MetaTrader will only read files in the MT platform "Files" folder, for example: C:\Program Files (x86)\FXCM MetaTrader 4\MQL4\Files
Line (number)	integer	The line number of the file to read. The special value LAST_LINE always reads the last line of the file.
ColumnData	Any data type	The data column to read, as defined in the File IO Configuration. The selection of the ColumnData parameter determines the data type returned from this function. The values available for selection are determine by the data columns defined in the File IO Configuration. For example, if three data columns are defined as: SymbolName (string) BidPrice (double) AskPrice (double) Those three values will be available for the selection of the ColumnData parameter. If the "SymbolName" value is selected, the function will return a **string** data type. If the "BidPrice" or "AskPrice" value is selected, the function will return a **double** data type. The returned data type may be important if the value is used in a comparison statement of a logical condition.

fnFileWrite

The function ***fnFileWrite*** is used to write a single line of to a file. The configuration of the file is defined in the *File IO* System Manager.

The **Configure** tab contains the parameter for the file name:

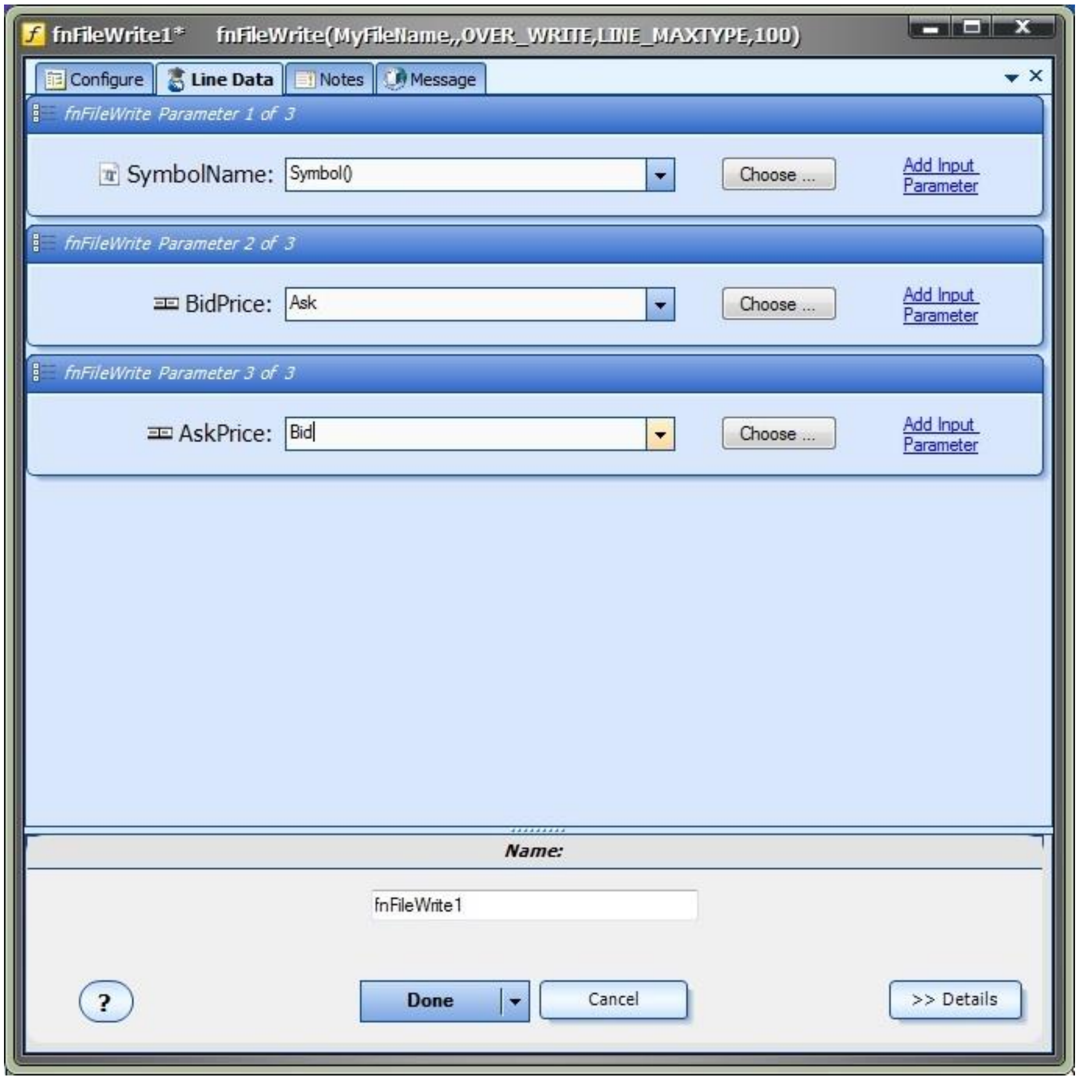


Filename: MyFileName

Name: fnFileWrite2

Parameter Name	Data type	Description
FileName	string	The name of the file to read. MetaTrader will only read files in the MT platform "Files" folder, for example: C:\Program Files (x86)\FXCM MetaTrader 4\MQL4\Files

The **Line Data** tab contains parameters for the Column Data defined in the *File IO* System Manager. For this example, we have added Columns named SymbolName, BidPrice and AskPrice.

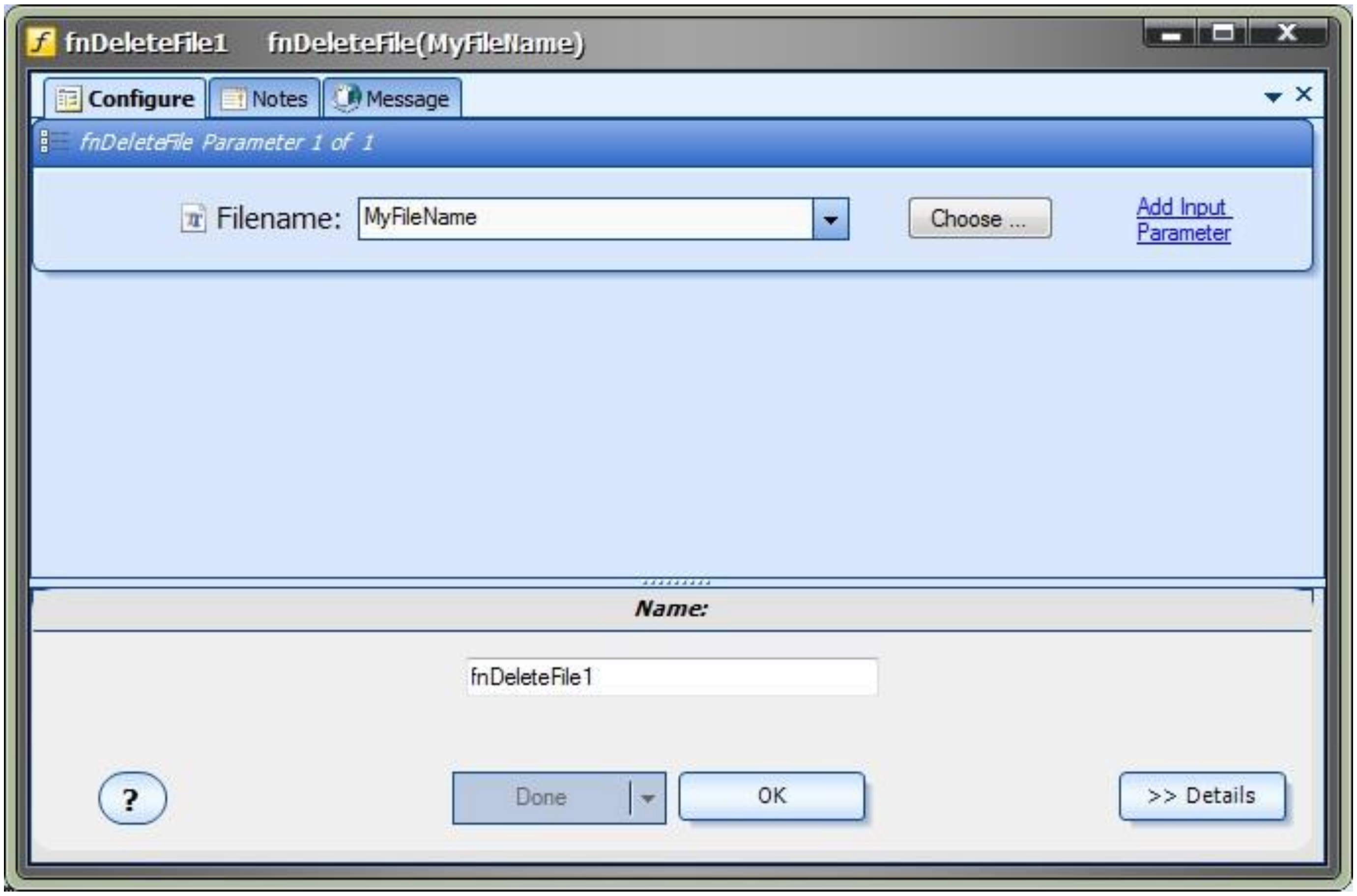


Note: There will be no parameters shown if no Columns have been added in the *File IO* System Manager.

Parameter Name	Data type	Description
SymbolName	string	As defined by the user in in the *File IO* System Manager.
BidPrice	double	As defined by the user in in the *File IO* System Manager.
AskPrice	double	As defined by the user in in the *File IO* System Manager.

fnDeleteFile

The function ***fnDeleteFile*** is used to delete a file from the file system. (Note: a file is opened automatically when using the function fnFileWrite)


fnDeleteFile1 fnDeleteFile(MyFileName)
Configure
Notes
Message
fnDeleteFile Parameter 1 of 1
Choose ...
Add Input Parameter
?
Done
OK
>> Details

Filename: MyFileName

Name: fnDeleteFile1

MetaTrader will only read or write files in the MT platform "Files" folder.
For example: C:\Program Files (x86)\FXCM MetaTrader 4\MQL4\Files

Using the File IO Plug-in

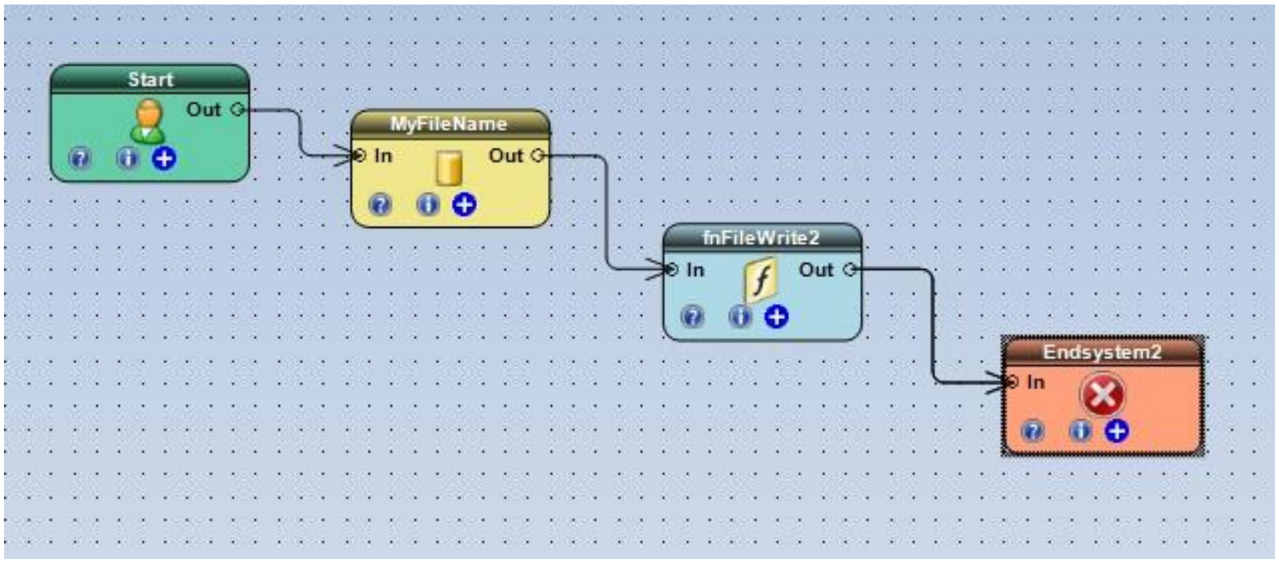
Writing Data

This drawing will write data to a file. The function **fnFileWrite2** is configured as shown in the fnWriteFile section.

On each tick, the Symbol, Bid price and Ask price are written to the file. (The file name is defined in the variable MyFileName).

This is a sample of a single line:

EURUSD,1.1229200000,1.1232500000

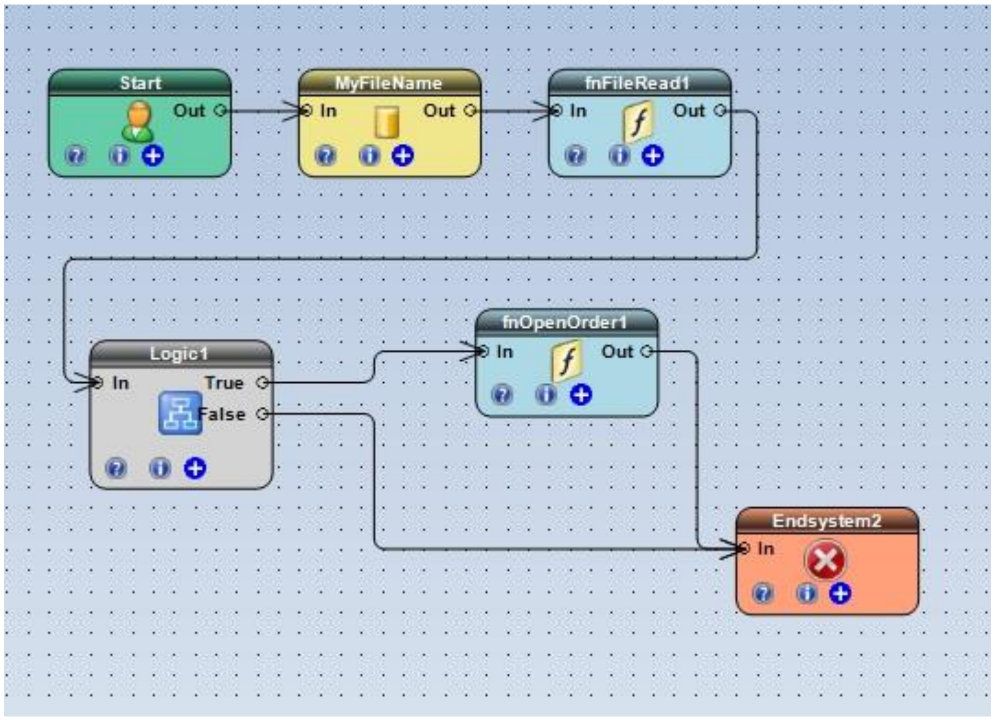


Reading Data

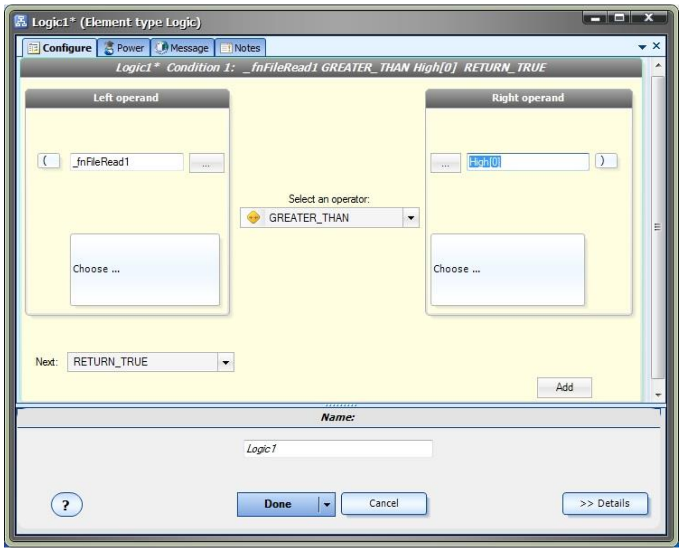
This drawing will read data from a file. The function **fnFileRead** is configured as shown in the fnReadFile section.

The last line of the file is read, and the column that holds the **BidPrice** is returned as a double.

The value is stored in a variable named **_fnFileRead1**.



The variable **_fnFileRead1** is used in the Logic condition and is compared to the High price:



Logic1* Condition 1: _fnFileRead1 GREATER_THAN High[0] RETURN_TRUE

Left operand: (_fnFileRead1

Select an operator: GREATER_THAN

Right operand: High[0])

Next: RETURN_TRUE

Name: Logic1

If the **BidPrice** (as stored in the **_fnFileRead1** variable) is greater than the High price of the current candle, the ***true*** path of the Logic Element is followed and a trade is opened by the function fnOpenOrder.