The ActiveX Interface of Facon Communication Server

(Doc.V1.0 05/13/2003)

Methods	Description
OpenProject	Open the previous saved project file (with .fcs file extension). The function is the same as [open project] menu in the Facon Server Windows
SaveProject	Save the configuration data into the project file. The function is the same as [save project] menu in the Facon Server Windows
Connect	Start to connect the PLCs and retrieve the data continuously. The function is the same as [connect] menu in the Facon Server Windows
Disconnect	Terminate the connection with PLCs. The function is the same as [Disconnect] menu in the Facon Server Windows
AddGroup	Add a new data group. The function is the same as [Add group] menu in the Facon Server Windows
EditGroup	Edit the data group. The function is the same as [Edit group] menu in the Facon Server Windows
DeleteGroup	Delete the data group. The function is the same as [Delete group] menu in the Facon Server Windows
AddItem	Add a new data item for automatic retrieving. The function is the same as [Add item] menu in the Facon Server Windows
DeleteItem	Delete a data item. The function is the same as [Delete item] menu in the Facon Server Windows
GetItem	Get the value of a data item
SetItem	Write value into a data item

OpenProject

Open a project file, which was created by Facon Server manually or by the program thru the ActiveX interfaces. Based on the project's content (The Channel, controller, group and item specify), while calling the CONNECT method, the Facon server will connect and retrieve or write the data to/from the PLCs accordingly.

```
HRESULT _stdcall OpenProject(
    BSTR filename              // Project File
 );
```

Inputs

filename

[In] Project File name for opening

Returns

S_OK while return with success

S_FALSE while return with fail

Note

The application program must first open the project file before it can connect with PLCs.

Related methods

SaveProject,Connect,Disconnect.

SaveProject

Save the project data into file.

```
HRESULT _stdcall SaveProject(
    BSTR filename            //Project file name
);
```

Inputs

filename

[In] File name for storing project data.

Returns

S_OK while return with success

S_FALSE while return with fail

Related methods

OpenProject,Connect,Disconnect.

Connect

Start to establish the connections with PLC and retrieve the data periodically.

```
HRESULT _stdcall Connect();
```

Inputs

None

Returns

S_OK while return with success

S_FALSE while return with fail

Notes

Before calling COONECT method, the project should be opened.

Related methods

OpenProject,SaveProject,Disconnect.

Disconnect

Terminate the connections with PLCs

```
HRESULT _stdcall Disconnect();
```

Inputs

None

Returns

S_OK while return with success

S_FALSE while return with fail

Note

Only when the connection was established by the CONNECT method, can the DISCONNECT method be called

Related methods

OpenProject,SaveProject,Connect.

AddGroup

Add a new data group. The items under the same group can be managed together Which share the same attributes such as retrieve priority or enable control

```
HRESULT _stdcall AddGroup(
    BSTR path,              // The data path of added group
    BSTR groupname,         // The name of the group
    byte priority,          // The retrieving priority.
    byte enabled            //Enable/Disable control of the data group
    );
```

Inputs

path

[In] The data path of added group

groupname

[In] The name of the group

priority

[In] The retrieving priority

0 stands for the first priority. 1 stands for the medium priority while 2 is the lowest priority

enabled

[In] Initial state of data group

0 stands for disable, 1 stands for enable

Returns

S_OK while return with success operation

S_FALSE while return with fail operation

Note

Naming notation of group path: When add a “Group1” group under the devices[Channel0] and controller[Station1], the group path name is denoted by “Channel0.Station1”, please note that the device and the controller is separated by a dot character.

Related methods

OpenProject,SaveProject,EditGroup,DeleteGroup,AddItem,DeleteItem.

EditGroup

Edit the attributes attached to a specific group. The attributes include retrieving priority and enable control.

```
HRESULT _stdcall EditGroup(
    BSTR path,              // Path name of the group
    BSTR groupname,         // Group name
    byte priority,          // Retrieving priority
    byte enabled            // Enable control
);
```

Inputs

path

[In] Path name of the group to be edited

groupname

[In] Group name to be edited

priority

[In] The retrieve priority

0 stands for the first priority. 1 stands for the medium priority while 2 is the lowest priority

enabled

[In] Enable Control

0 stands for disable, 1 stands for enable

Returns

S_OK while return with success operation

S_FALSE while return with fail operation

Note

Naming notation of group path: When a “Group1” group under the device[Channel0] and controller[Station1], the group path name is denoted by “Channel0.Station1”, please note that the device and the controller are separated by a dot character.

Related methods

OpenProject,SaveProject,AddGroup,DeleteGroup,AddItem,DeleteItem.

DeleteGroup

Delete the unwanted group defination

```
HRESULT _stdcall DeleteGroup(
   BSTR path,              // Path name of the group
   BSTR groupname,         // Group name
);
```

Inputs

path

[In] Path name of the group to be deleted

groupname

[In] Name of the group to be deleted.

Returns

S_OK while return with success operation

S_FALSE while return with fail operation

Note

While delete a group, all the data items under the group will also be deleted.

Related methodes

OpenProject,SaveProject,AddGroup,EditGroup,AddItem,DeleteItem.

AddItem

Add a data item or items into a group. The retrieving priority and the stop/start control of the data item can be set by editing the group attributes. The retrieving of the data items under the same group will be arranged in the same communication session as could as possible if the message size is not exceed the packet limitation. Care should be taken when the synchronization of the data is concerned.

```
HRESULT _stdcall AddItem(
    BSTR path,              // Full path name of the group that the item is under.
    BSTR itemname,          // Item name to be added into the group.
);
```

Inputs

path

[In] Full path name of the group that the item are going to add

itemname

[In] Name of the item(items) to be added into the group. There are some built in items associate with System, Channel and Group can be accessed. Please refer the appendix III for detail information.

Return

S_OK while return with success operation

S_FALSE while return with fail operation

Note

Naming notation while locate a certain group: When specify a “Group1” group under the device[Channel0] and controller[Station1], the full path name of the group is “Channel0.Station1.Group1”, please note that the device, the controller and group are separated by the dot character. The item name can be expressed as a single item(“R0”) or range of item(“R0-R8”) or list of item(“R0,R1,R5”).

Related methodes

AddGroup,EditGroup,DeleteGroup,DeleteItem.

DeleteItem

Remove the data items from within a group

```
HRESULT _stdcall DeleteItem(
   BSTR path,              // Full path name of the group
   BSTR itemname,          // Name of the item(items)
);
```

Inputs

path

[In] Full path name of the group that the item are going to remove.

itemname

[In] Name of the item(items) to be deleted from the group

Return

S_OK while return with success operation

S_FALSE while return with fail operation

Note

Naming notation while locate a certain group: When specify a “Group1” group under the device[Channel0] and controller[Station1], the full path name of the group is “Channel0.Station1.Group1”, please note that the device, the controller and group are separated by the dot character. The item name can be expressed as a single item(“R0”) or range of item(“R0-R8”) or list of item(“R0,R1,R5”).

Related methodes

AddGroup,EditGroup,DeleteGroup,AddItem.

GetItem

Get current value of specific item.

```
HRESULT _stdcall GetItem(
    BSTR path,              // Path of the item
    BSTR itemname,          // item name intent to read
    VARIANT* value          // Holding place for the item value
);
```

Inputs

path

[In] The path of the data item to be read

itemname

[In] The name of the data item to be read

value

[out, retval] Holding place for item value

Return

S_OK while return with success operation

S_FALSE while return with fail operation

Note

Naming notation while locate a certain group: When specify a “Group1” group under the device[Channel0] and controller[Station1], the full path name of the group is “Channel0.Station1.Group1”, please note that the device, the controller and group are separated by the dot character.

Related methodes

AddItem,DeleteItem,SetItem.

SetItem

Write the new value into a specific item

```
HRESULT _stdcall GetItem(
   BSTR path,              //Path
   BSTR itemname,          //Item name
   BSTR value              //New value of item
);
```

Inputs

path

[In] The path of the data item to be written

itemname

[In] The name of the data item to be written

value

[In] The new value of the data item to be written

Return

S_OK while return with success operation

S_FALSE while return with fail operation

Note

Naming notation while locate a certain group: When specify a “Group1” group under the device[Channel0] and controller[Station1], the full path name of the group is “Channel0.Station1.Group1”, please note that the device, the controller and group are separated by the dot character.

Related methodes

AddItem,DeleteItem,GetItem.

Appendix I

Sample VB program

Description:

While startup the sample program, the FaconServer will execute automatically and load the project file(D:\Demo.fcs). After that the sample program will connect the PLC and retrieve the content of R0 periodically with timer function and show the value on the window's caption.

```
Dim server As Object      'FaconServer object declaration

Private Sub Form_Load()
  Set server = CreateObject("FaconSvr.FaconServer")       'Create the FaconServer object
  server.OpenProject ("D:\DEMO.fcs")                      ' Open the project file(D:\Demo.fcs)
                                                          ' which was created previously
  ConnectBtn_Click
End Sub

Private Sub ConnectBtn_Click()
  server.Connect                          'Connect with FaconServer in order to read back the PLC
                                          data
  Timer1.Enabled = True                   'Enable timer
End Sub

Private Sub DisconnectBtn_Click()
  Timer1.Enabled = False                  'Stop timer
  server.Disconnect                       'Disconnect with FaconServer
End Sub

Private Sub Timer1_Timer()
  Label1.Caption = server.GetItem("Chennel0.Station0.Group0", "R0")       ' Read back R0 value

  If server.GetItem("Chennel0.Station0.Group0", "M1922") > 0 Then         ' Read back M1922
    Shape1.FillColor = &HFF&
  Else
    Shape1.FillColor = &HFF00&
  End If
End Sub

Private Sub AddGroup_Click()
  a = server.AddGroup("Chennel0.Station0", "Group1", 0, 1)
                        'Dynamic create Group1 under Chennel0.Station0 with high priority
  a = server.AddItem("Chennel0.Station0.Group1", "R0-R10")
                                    'Create Item R0-R10 under Chennel0.Station0.Group1
End Sub

Private Sub DeleteGroup_Click()  ' Dynamic remove group
  a = server.DeleteGroup("Chennel0.Station0", "Group1")
                                    '  Remove Group0 under Chennel0.Station0
End Sub

Private Sub AddItem_Click()' Dymanic add items
  a = server.AddItem("Chennel0.Station0.Group0", "R1-R5,R8,M1922")
'Add R1~R5and R8,M1922 items under Chennel0.Station0.Group1

End Sub
```

```
Private Sub DeleteItem_Click()
  a = server.DeleteItem("Chennel0.Station0.Group0", "R1")
                                        'Remove R1 item under Chennel0.Station0.Group1
End Sub

Private Sub ItemWrite_Click()
  a = server.SetItem("Chennel0.Station0.Group0", "R0", 100)     'write 100 into R0
End Sub
```

Appendix II
IDL defination

```
[
  uuid(39EC6DB2-DE51-4843-8629-5BA95C57C192),
  version(1.0),
  helpstring("FaconSvr Library")

]
library FaconSvr
{

  importlib("stdole2.tlb");
  importlib("stdvcl40.dll");

  [
    uuid(E33C7B53-90BB-41E9-88E5-3DD8D485E95B),
    version(1.18),
    helpstring("Dispatch interface for FaconServer Object"),
    dual,
    oleautomation
  ]
   interface IFaconServer: IDispatch
  {
    [
    id(0x00000001)
    ]
    HRESULT _stdcall OpenProject([in] BSTR filename );
    [
    id(0x00000002)
    ]
    HRESULT _stdcall SaveProject([in] BSTR filename );
    [
    id(0x00000003)
    ]
    HRESULT _stdcall Connect( void );
    [
    id(0x00000004)
    ]
    HRESULT _stdcall Disconnect( void );
    [
    id(0x00000005)
    ]
    HRESULT _stdcall AddGroup([in] BSTR path, [in] BSTR groupname, [in] byte priority, [in] byte
enabled );
    [
    id(0x00000006)
    ]
    HRESULT _stdcall EditGroup([in] BSTR path, [in] BSTR groupname, [in] byte priority, [in] byte
enabled );
    [
    id(0x00000007)
    ]
    HRESULT _stdcall DeleteGroup([in] BSTR path, [in] BSTR groupname );
    [
    id(0x00000008)
    ]
    HRESULT _stdcall AddItem([in] BSTR path, [in] BSTR itemname );
    [
    id(0x00000009)
    ]
    HRESULT _stdcall DeleteItem([in] BSTR path, [in] BSTR itemname );
    [
    id(0x0000000A)
    ]
    HRESULT _stdcall GetItem([in] BSTR path, [in] BSTR itemname, [out, retval] VARIANT * value );
    [
    id(0x0000000B)
    ]
    HRESULT _stdcall SetItem([in] BSTR path, [in] BSTR itemname, [in] BSTR value );
  };
```

```
  [
    uuid(6FC145E9-A43F-4A71-89B5-9D7284540D5B),
    version(1.7),
    helpstring("Events interface for FaconServer Object")
  ]
   dispinterface IFaconServerEvents
  {
    properties:
    methods:
  };

  [
    uuid(56A9ADF8-604D-40B7-A696-990FCFAD1E46),
    version(1.0),
    helpstring("FaconServer Object")
  ]
  coclass FaconServer
  {
    [default] interface IFaconServer;
    [default, source] dispinterface IFaconServerEvents;
  };

};
```

Appendix III
Built in data items

System.Connect

This item can be both read and written. When read, which reflects the online status of server. If the value is '1' then Facon server is online otherwise is offline. Write this item can control the facon server into offline or online state.

Chennel.Active

This type of item can only be read, which reflects the current status of specific channel device. If the device is successful online then this value is '1' otherwise means connect error or not online.

Chennel.Type

This type of item can only be read, which indicates the device name of specific channel device.

Device Type	Device Name
RS232	"RS232"
MODEM	"MODEM"
UDP	"UDP"
TCP	"TCP"

Chennel.Parameter

This type of item can be both read and written, which indicates the parameter of specific channel device.
The content of this item is device dependent. When the device is RS232 then the content is the port number of PC to connect. Write a new value to this item can change the parameter as new one.

Note: When change the new parameter, the server will first disconnect the channel then connect with new parameter.

Device Type	parameter	Description of parameter
RS232	"COM1"	Communication port
MODEM	"56784321"	Dial out phone number for modem
UDP	"192.168.1.1"	IP of connected channel
TCP	"192.168.1.1"	IP of connected channel

Channel.Station.Sleep

This type of item can only be read, which indicates if the station is connected. If the value is '0' means OK otherwise the station is not accessible.

Channel.Station.WriteCount

This type of item can only be read, which indicates if the write requests to the station were all processed.
If the value is 0 then all previous write attempts to the specific station was processed OK.

Channel.Station.Group.Priority

This type of item can only be read, which indicates the priority of specific data group.

Priority	value
Highest	0
Normal	1
Lowest	2

Channel.Group.Update

This type of item can only be read, which indicates the refresh time (unit in ms) of specific data group.

Channel.Group.UpdateTime

This type of item can only be read, which indicates the latest data updated time of specific data group.

Channel.Group.Status

This type of item can be both read and written, which indicates if the specific data group is in the refresh list. If the value is '1', means this data group is in the list and its value will be updated periodically.
Otherwise it will not. Write this item can control if the data group refresh or not.

Channel.Group.RefreshCount

This type of item can only be read, which indicates the total refresh times of specific data group since it started to refresh.