

## ***DECISION Image Capture OCX***

### **I. Local Operation:**

#### **1. ImageCapturePassword**

Syntax:

*long ImageCapturePassword(string Password);*

Description:

- The first and foremost function to be called before any other functions. Specifying the password enables you to use the OCX. The Password is "DECISION COMPUTER".

Prerequisites:

- None

Return Value:

0-OK

9-Wrong Image Capture Password

Usage:

long ret=id.ImageCapturePassword("DECISION COMPUTER");

#### **2. Display**

Syntax:

*long Display(long Condition, long UpperLeftXCor, long UpperLeftYCor, long Width, long Height);*

Description:

-The function to be called next after ImageCapturePassword. It is used to set the OCX to display or not the captured image. The Condition parameter is used for specifying whether to display or not, the real-time captured image(1-display, 0-do not display). The UpperLeftXCor and UpperLeftYCor parameters are used for specifying the starting point of the image. The Width and Height parameters are used to indicate the width and height of the image.

Prerequisites:

- ImageCapturePassword

Return Value:

0-OK

9-Wrong Image Capture Password

Usage:

long ret=id.Display(1,0,0,300,300);     //set the image size to 300x300 pixels

#### **3. OpenImageCaptureDevice**

Syntax:

*long OpenImageCaptureDevice(long DeviceNo);*

Description:

-This is used to specify which image capture device to use. The first image capture device is 0.

Prerequisites:

- ImageCapturePassword, Display

Return Value:

0-OK

3-No such Device (the device number you specified, does not exist)

4-Frame size not defined (user must specify the frame size, using the Display method)

9-Wrong Image Capture Password

Usage:

long ret=id.OpenImageCaptureDevice(0);     //opens the first device

#### 4. StartImageCaptureDevice

Syntax:

*long StartImageCaptureDevice();*

Description:

-This is used to signal the OCX to start to capture images from the image capture device. This method may or may not display the captured images to screen, the option to display is in the parameter "Condition" of the "Display" method. While the starting location and size of the image is defined using the UpperLeftXCor, UpperLeftYCor, Width and Height parameters of the "Display" method.

Prerequisites:

- ImageCapturePassword, Display, OpenImageCaptureDevice

Return Value:

0-OK

5-Device not opened

9-Wrong Image Capture Password

Usage:

long ret=id.StartImageCaptureDevice();     //start capturing

#### 5. CaptureImage

Syntax:

*long CaptureImage(string Filename, string Type);*

Description:

-This method is used to save a captured image frame to a file. The "Filename" parameter specifies the path and filename. While the "Type" parameter specifies how the image is going to be saved ("BMP"-as a bitmap file, "JPG"-as a JPEG file). You must first issue the StartImageCaptureDevice method before this one.

Prerequisites:

- ImageCapturePassword, Display, OpenImageCaptureDevice, StartImageCaptureDevice

Return Value:

0-OK

9-Wrong Image Capture Password

Usage:

long ret=id.CaptureImage("c:\\mypic.jpg","JPG");     //save captured image as a jpeg file

## 6. AutoCapture

### Syntax:

*long AutoCapture(short Activate, short Threshold, long UpperLeftXCor, long UpperLeftYCor, long Width, long Height);*

### Description:

-This method is used to set the OCX to motion detection. The "Activate" parameterThe OCX will automatically capture an Image upon detection of motion on the defined hotspot (specified by the parameters "UpperLeftXCor, UpperLeftYCor, Width and Height). The captured image will be saved in the file as "cap\_X-Y\_Z.bmp" in the root directory. The "X" represents the Hour, "Y" represents the minute and Z represents the second the image was captured. The "Threshold" parameter defines the sensitivity during capture (0..5, 0-being the most sensitive to change while 5-being the least sensitive to motion), the motion detection sequence is done using the root mean square equation. To activate the OCX to auto capture, you must first issue the StartImageCaptureDevice method.

### Prerequisites:

- ImageCapturePassword, Display, OpenImageCaptureDevice, StartImageCaptureDevice

### Return Value:

0-OK

9-Wrong Image Capture Password

### Usage:

long ret=id.AutoCapture(1,0,0,0,200,200); //activate auto capture and set to high sensitivity

## 7. StopImageCaptureDevice

### Syntax:

*long StopImageCaptureDevice();*

### Description:

-This is used to signal the OCX to stop capturing images from the image capture device.

### Prerequisites:

- ImageCapturePassword, Display, OpenImageCaptureDevice, StartImageCaptureDevice

### Return Value:

0-OK

9-Wrong Image Capture Password

### Usage:

long ret=id.StopImageCaptureDevice(); //stop capturing

## 8. CloseImageCaptureDevice

### Syntax:

*long CloseImageCaptureDevice();*

### Description:

-This is used to close the opened image capture device. Note, before closing the device, the user must issue StopImageCaptureDevice first to stop the device from running. After issuing StopImageCaptureDevice, the user must delay for 3 seconds before closing the device, because it takes time for an image capture device to stop.

### Prerequisites:

- ImageCapturePassword, Display, OpenImageCaptureDevice, StartImageCaptureDevice, StopImageCaptureDevice

### Return Value:

0-OK

9-Wrong Image Capture Password

### Usage:

long ret=id.StopImageCaptureDevice(); //stop capturing

## 9. ShowImageFiletoScreen

### Syntax:

*long ShowImageFiletoScreen(string Filename, string Type, long UpperLeftXCor, long UpperLeftYCor, long WidthofRect, long HeightofRect);*

### Description:

-This method should be used whenever the user wants to display an image file to screen. The “Filename” parameter defines the file to display to the screen. The “Type” parameter defines the type of the image file (BMP-for bitmapped files, JPG-for jpeg files), while the “UpperLeftXcor” and “UpperLeftYCor” defines the starting point; and the WidthofRect and HeightofRect defines the size of the image frame

### Prerequisites:

- ImageCapturePassword

### - Return Value:

0-OK

9-Wrong Image Capture Password

### Usage:

```
long ret=id.ShowImageFiletoScreen("c:\\test1.bmp","BMP",50,50,250,300); //display the image in a
//250x300 frame starting at 50,50
```

## II. Network/Remote Operation:

### \*\*\*NOTE\*\*\*

-For remote operation, the server must be set-up first before the client. And during shutdown, the client should be the first to be shutdown before the server.

### a. Server Side:

## 1. ActivateRemoteControl

### Syntax:

*long ActivateRemoteControl(LPCTSTR Password, long PortNo);*

### Description:

-This method should be executed in the server side, it triggers/starts the OCX to monitor for any client requests. The “Password” parameter defines the password that the client must issue for any request. The “PortNo” parameter defines the port number to be used for Client-Server connection, this value must also be specified on client requests.

### Prerequisites:

- ImageCapturePassword

### - Return Value:

0-OK

8-Listening error

9-Wrong Image Capture Password

### Usage:

```
long ret=id.ActivateRemoteControl("DECISION P",5152); //activate server
```

## 2. DeactivateRemoteControl

### Syntax:

*long DeactivateRemoteControl();*

### Description:

-This method should be executed in the server side, it halts / stops the OCX to monitor for any client requests. This is used to terminate the connection established by the ActivateRemoteControl method..

### Prerequisites:

- ImageCapturePassword, ActivateRemoteControl

### - Return Value:

0-OK

9-Wrong Image Capture Password

### Usage:

long ret=id.DeactivateRemoteControl(); //stop monitoring

## 3. ProcessingDelay

### Syntax:

*long ProcessingDelay(long DELAY);*

### Description:

-According to test conducted, some Internet connections are slow while others are fast. This greatly affects the response of client-server communication, situation may arise that the server needs to be slowed down for proper CLIENT-SERVER communication. The “DELAY” parameter defines the delay that must be produced for proper communication (the unit is in milliseconds). The prescribed DELAY is 1000. Note that the delay may vary, depending on the network.

### Prerequisites:

-ImageCapturePassword,

### - Return Value:

0-OK

9-Wrong Image Capture Password

### Usage:

long ret=id.ProcessingDelay(1000); //produce a processing delay of 1000 milliseconds

## **b. Client Side:**

### **4. DisplayRemote**

#### **Syntax:**

*long **DisplayRemote**(string IPaddress, long PortNo, string Password, long Condition, long UpperLeftXCor, long UpperLeftYCor, long Width, long Height);*

#### **Description:**

-Use this method to set the OCX to display or not, the captured image. The “IPaddress” parameter defines the ip address of the server. The “PortNo” defines the port number of the server as specified in the ActivateRemoteControl method, and the “Password” defines the password used for the server as specified in the ActivateRemoteControl method. The Condition parameter is used for specifying whether to display or not the real-time captured image(1-display, 0-do not display). The UpperLeftXCor and UpperLeftYCor parameters are used for specifying the starting point of the image. The Width and Height parameters are used to indicate the width and height of the image. All of the effects of this method is done in the server side.

#### **Prerequisites:**

- ImageCapturePassword

#### **- Return Value:**

0-OK

2-Server not Ready

6-Wrong Password

7-Unknown Service Request

9-Wrong Image Capture Password

#### **Usage:**

```
long ret=id.DisplayRemote(“202.52.117.90”,5152,“DECISION P”,1,10,10,200,300); //set the server
//image capture device to display the captured image to screen
```

### **5. OpenRemoteImageCaptureDevice**

#### **Syntax:**

*long **OpenRemoteImageCaptureDevice**(LPCTSTR IPaddress, long PortNo, LPCTSTR Password, long DeviceNo);*

#### **Description:**

--This is used to specify which image capture device to use. The first image capture device is 0. The “IPaddress” parameter defines the ip address of the server. The “PortNo” defines the port number of the server as specified in the ActivateRemoteControl method, and the “Password” defines the password used for the server as specified in the ActivateRemoteControl method. The “DeviceNo” parameter defines the device to be opened. All of the effects of this method is done in the server side.

#### **Prerequisites:**

- ImageCapturePassword, DisplayRemote

0-OK

2-Server not Ready

3-No such device

4-Frame size not defined

6-Wrong Password

7-Unknown Service Request

9-Wrong Image Capture Password

#### **Usage:**

```
long ret=id.OpenRemoteImageCaptureDevice(“202.51.117.90”,5152,“DECISION P”,0); //opens the first
//device
```

## 6. StartRemoteImageCaptureDevice

### Syntax:

*long StartRemoteImageCaptureDevice(LPCTSTR IPaddress, long PortNo, LPCTSTR Password);*

### Description:

- This is used to signal the OCX to start to capture images from the image capture device. This method may or may not display the captured images to screen, the option to display is in the parameter "Condition" of the "DisplayRemote" method. While the starting location and size of the image is defined using the UpperLeftXCor, UpperLeftYCor, Width and Height parameters of the "DisplayRemote" method. The "IPaddress" parameter defines the ip address of the server. The "PortNo" defines the port number of the server as specified in the ActivateRemoteControl method, and the "Password" defines the password used for the server as specified in the ActivateRemoteControl method. All of the effects of this method is done in the server side.

### Prerequisites:

- ImageCapturePassword, DisplayRemote, OpenRemoteImageCaptureDevice
- 0-OK
- 2-Server not Ready
- 5-Device not opened
- 6-Wrong Password
- 7-Unknown Service Request
- 9-Wrong Image Capture Password

### Usage:

```
long ret=id.StartRemoteImageCaptureDevice("202.51.117.90",5152,"DECISION P"); //tell the server to
                                                                    //start
```

## 7. CaptureRemoteImage

### Syntax:

*long CaptureRemoteImage(string IPaddress, long PortNo, string Password, string Filename, string Type, string SaveLocation);*

### Description:

-This method is used to save a captured image frame to a file. The "Filename" parameter specifies the path and filename. While the "Type" parameter specifies how the image is going to be saved ("BMP"-as a bitmap file, "JPG"-as a JPEG file). The parameter "SaveLocation" defines upon which computer the image will be saved (SERVER-the image will be saved in the server side, CLIENT-the image will be saved in the client side).You must first issue the StartRemoteImageCaptureDevice method before this one. The "IPaddress" parameter defines the ip address of the server. The "PortNo" defines the port number of the server as specified in the ActivateRemoteControl method, and the "Password" defines the password used for the server as specified in the ActivateRemoteControl method.

### Prerequisites:

- ImageCapturePassword, DisplayRemote, OpenRemoteImageCaptureDevice, StartRemoteImageCaptureDevice, StartRemoteImageCaptureDevice
- 0-OK
- 2-Server not Ready
- 6-Wrong Password
- 7-Unknown Service Request
- 9-Wrong Image Capture Password

### Usage:

```
long ret=id.CaptureRemoteImage("202.51.117.90",5152,"DECISION P","c:\\test3.jpg","JPG","CLIENT")
                                                                    //capture and save in client side as jpeg file
```

## 8. RemoteAutoCapture

### Syntax:

*long RemoteAutoCapture(string IPaddress, long PortNo, string Password, short Active, short Threshold, long UpperLeftXCor, long UpperLeftYCor, long Width, long Height);*

### Description:

-This method is used to set the OCX to motion detection. The OCX will automatically capture an Image upon detection of motion on the defined hotspot (specified by the parameters “UpperLeftXCor, UpperLeftYCor, Width and Height). The captured image will be saved on the server in the file as “cap\_X-Y\_Z.bmp” in the root directory; “X” represents the Hour, “Y” represents the minute and Z represents the second the image was captured(eg. cap\_21-16\_49.bmp). The “Threshold” parameter defines the sensitivity during capture (0..5, 0-being the most sensitive to change while 5-being the least sensitive to motion), the motion detection sequence is done using the root mean square equation. To activate the OCX to auto capture, you must first issue the StartRemoteImageCaptureDevice method. The “IPaddress” parameter defines the ip address of the server. The “PortNo” defines the port number of the server as specified in the ActivateRemoteControl method, and the “Password” defines the password used for the server as specified in the ActivateRemoteControl method. All of the effects of this method are done in the server side.

### Prerequisites:

- ImageCapturePassword, DisplayRemote, OpenRemoteImageCaptureDevice,
  - 0-OK
  - 2-Server not Ready
  - 6-Wrong Password
  - 7-Unknown Service Request
  - 9-Wrong Image Capture Password

### Usage:

```
long ret=id.RemoteAutoCapture(“202.51.117.90”,5152,”DECISION P”,0,0,0,200,200)  
//detect motion in this location
```

## 9. StopRemoteImageCaptureDevice

### Syntax:

*long StopRemoteImageCaptureDevice(LPCTSTR IPaddress, long PortNo, LPCTSTR Password);*

### Description:

-This is used to signal the OCX to stop capturing images from the image capture device. The “IPaddress” parameter defines the ip address of the server. The “PortNo” defines the port number of the server as specified in the ActivateRemoteControl method, and the “Password” defines the password used for the server as specified in the ActivateRemoteControl method. All of the effects of this method are done in the server side.

### Prerequisites:

- ImageCapturePassword, DisplayRemote, OpenRemoteImageCaptureDevice, StartRemoteImageCaptureDevice
  - 0-OK
  - 2-Server not Ready
  - 6-Wrong Password
  - 7-Unknown Service Request
  - 9-Wrong Image Capture Password

### Usage:

```
long ret=id.StopRemoteImageCaptureDevice(“202.51.117.90”,5152,”DECISION P”); //tell the server to  
//stop capturing
```

## 10. CloseRemoteImageCaptureDevice

### Syntax:

*long CloseRemoteImageCaptureDevice(LPCTSTR IPaddress, long PortNo, LPCTSTR Password);*

### Description:

-This is used to close the opened image capture device. Note, before closing the device, the user must issue StopRemoteImageCaptureDevice first to stop the device from running. After issuing StopRemoteImageCaptureDevice, the user must delay for 3 seconds before closing the device, because it takes time for an image capture device to stop. The "IPaddress" parameter defines the ip address of the server. The "PortNo" defines the port number of the server as specified in the ActivateRemoteControl method, and the "Password" defines the password used for the server as specified in the ActivateRemoteControl method. All of the effects of this method are done in the server side.

### Prerequisites:

- ImageCapturePassword, DisplayRemote, OpenRemoteImageCaptureDevice, StartRemoteImageCaptureDevice, StopRemoteImageCaptureDevice

0-OK

2-Server not Ready

6-Wrong Password

7-Unknown Service Request

9-Wrong Image Capture Password

### Usage:

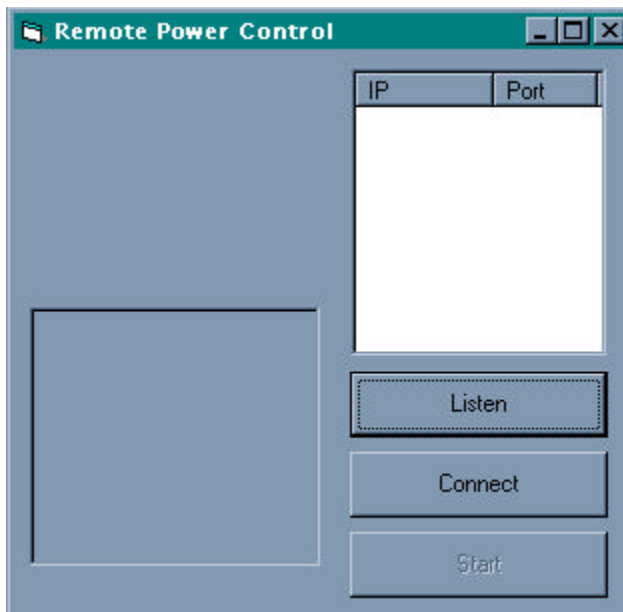
```
long ret=id.CloseRemoteImageCaptureDevice("202.51.117.90",5152,"DECISION P"); //tell the server to
//close the device
```

## **ERROR CODES Table:**

- 0- OK –operation was successful
- 1- Bind Error –someone is using the same portno (try changing the port number)
- 2- Server not Ready –the server side is not yet up, or the server cant be found (make sure the server side is up and running)
- 3- No such Device –invalid device no., there is no such device (try changing the device no/ make sure the image capture device is installed correctly)
- 4- Frame size not defined –the image frame is not defined (issue first the Display/ DisplayRemote method)
- 5- Device not opened –the device is not yet opened for opeartion (issue first the OpenImageCaptureDevice/ OpenRemoteImageCaptureDevice)
- 6- Wrong Password –the password is invalid (make sure the password specified in the ActivateRemoteControl is the password being issued)
- 7- Unknown Service Request –bad packet (try issuing the method again)
- 8- Listening Socket not Created –network problem (try restarting your pc)
- 9- Wrong Image Capture Password –wrong password (the password is DECISION COMPUTER)

## **Remote Video Operation Manual**

### **Server Application:**



1. To start Server Application just click "Listen" to invoke Image Capture Device to start capturing image.

Note: Before running Remote video you must install a CCD Camera in your system together with the necessary driver needed to make your hardware function properly.

### **Client Application:**

1. To start Client Application just click on Connect then a dialog box will prompt you to input the IP address of the Server then click OK then Start to invoke Server to send Captured image to client side.