

# **Yuan's ActiveX Control Function Guide**

## **◆ Overview**

For the ActiveX Control developer, In this document, we explain and guide developer how to use QCAP ActiveX Control that implement in developer computer.

## **◆ Test software**

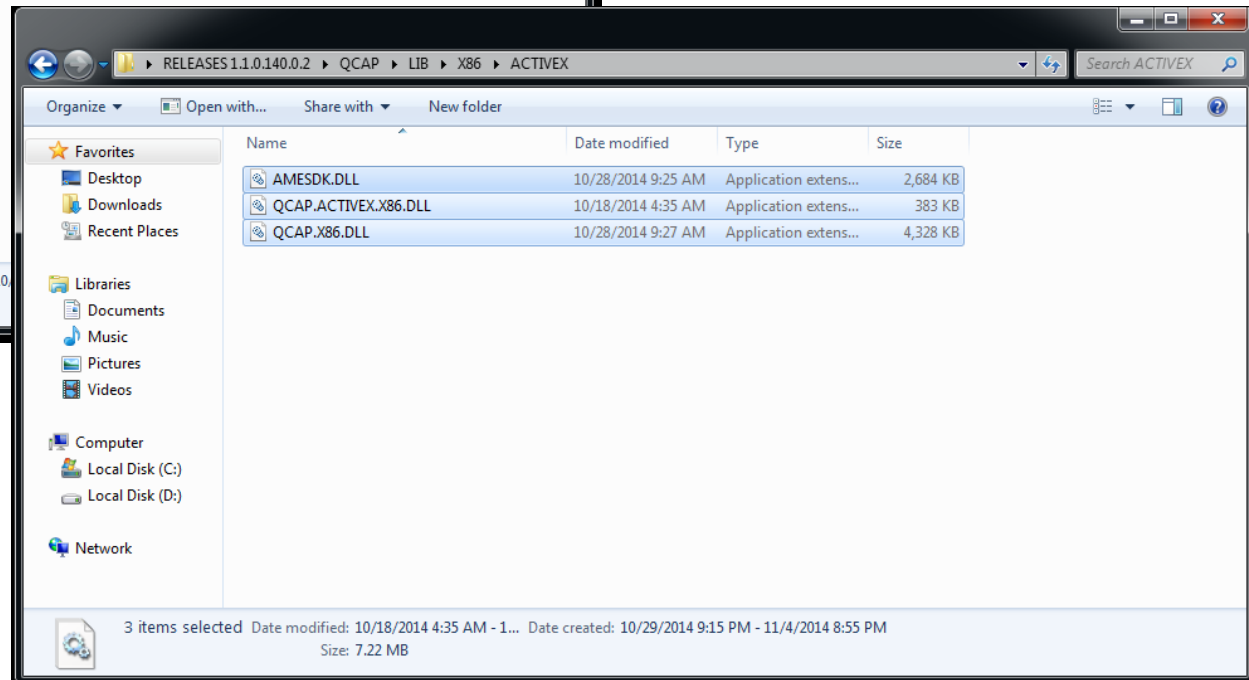
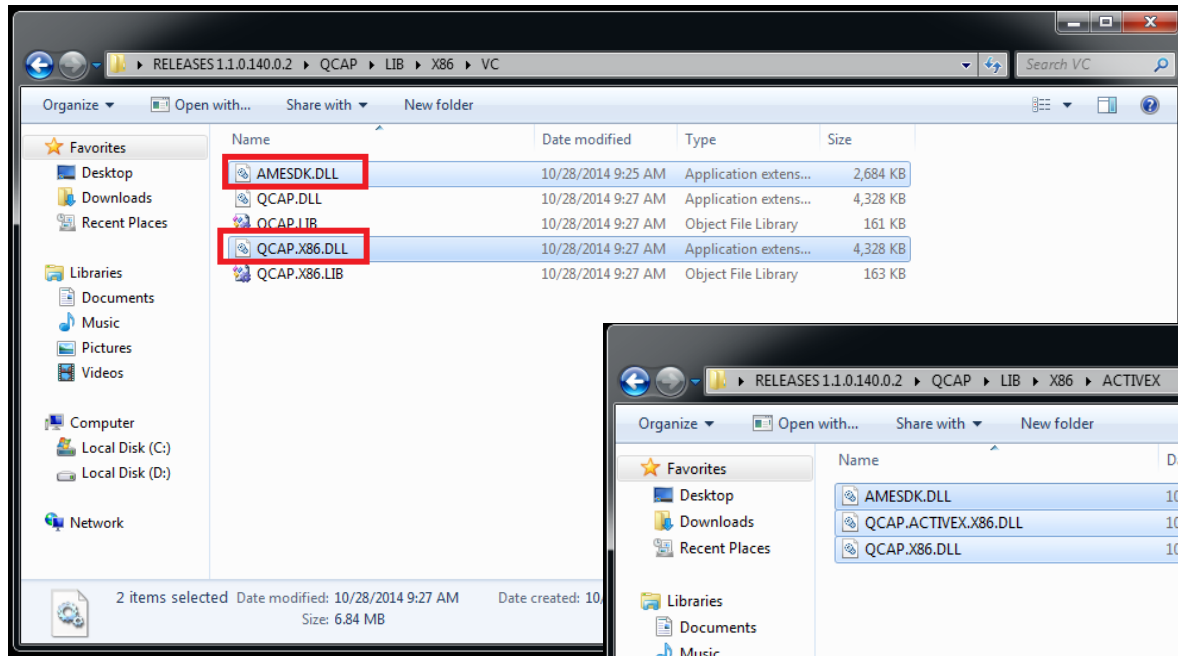
**AMESDK.DLL**

**QCAP.X86.DLL**

**QCAP.ACTIVEX.X86.DLL**

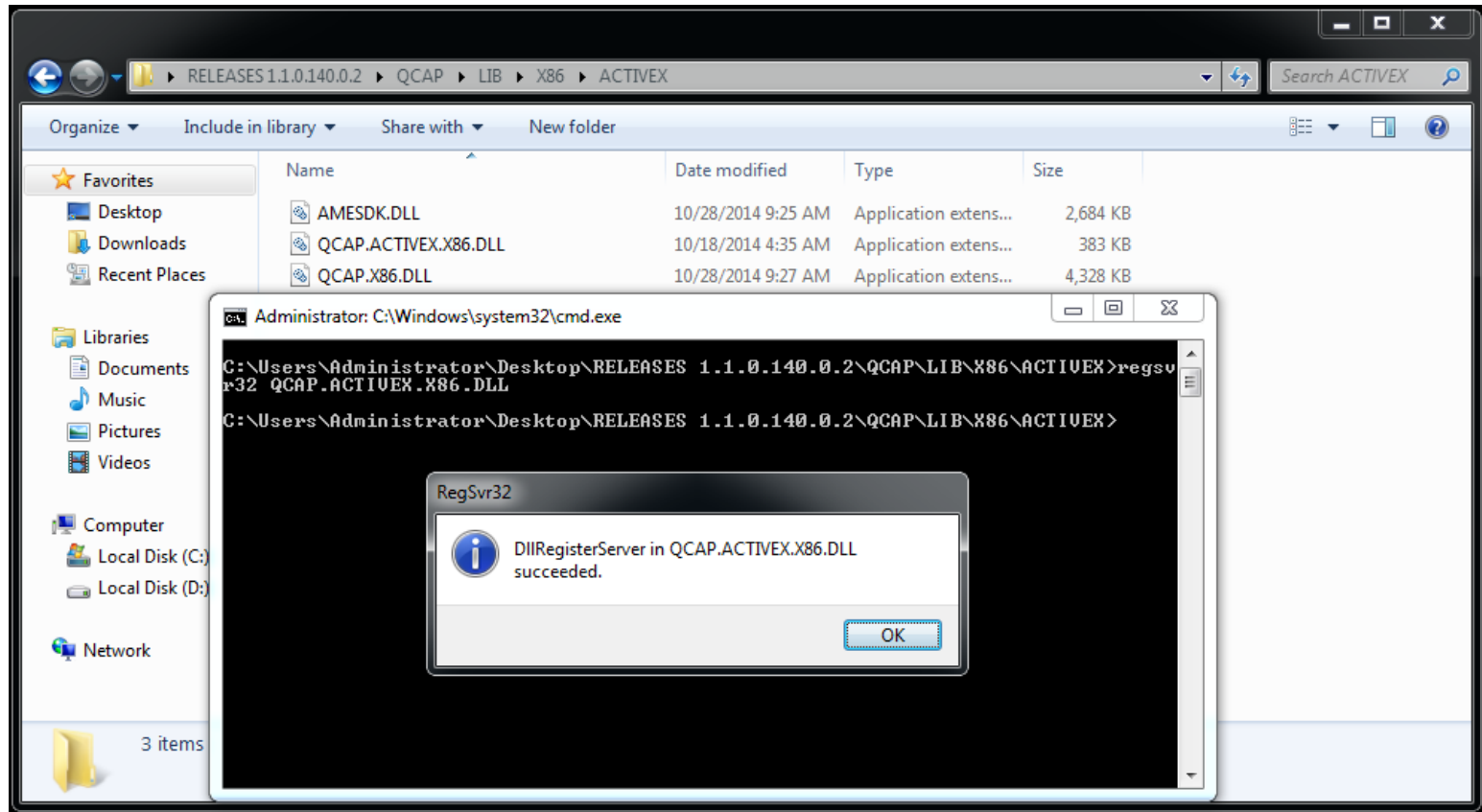
# Build QCAP ACTIVEX CONTROL application

1. Before register QCAP.ACTIVEX.X86.DLL, **AMESDK.DLL** and **QCAP.X86.DLL** must to be put on the same folder with QCAP.ACTIVEX.X86.DLL.



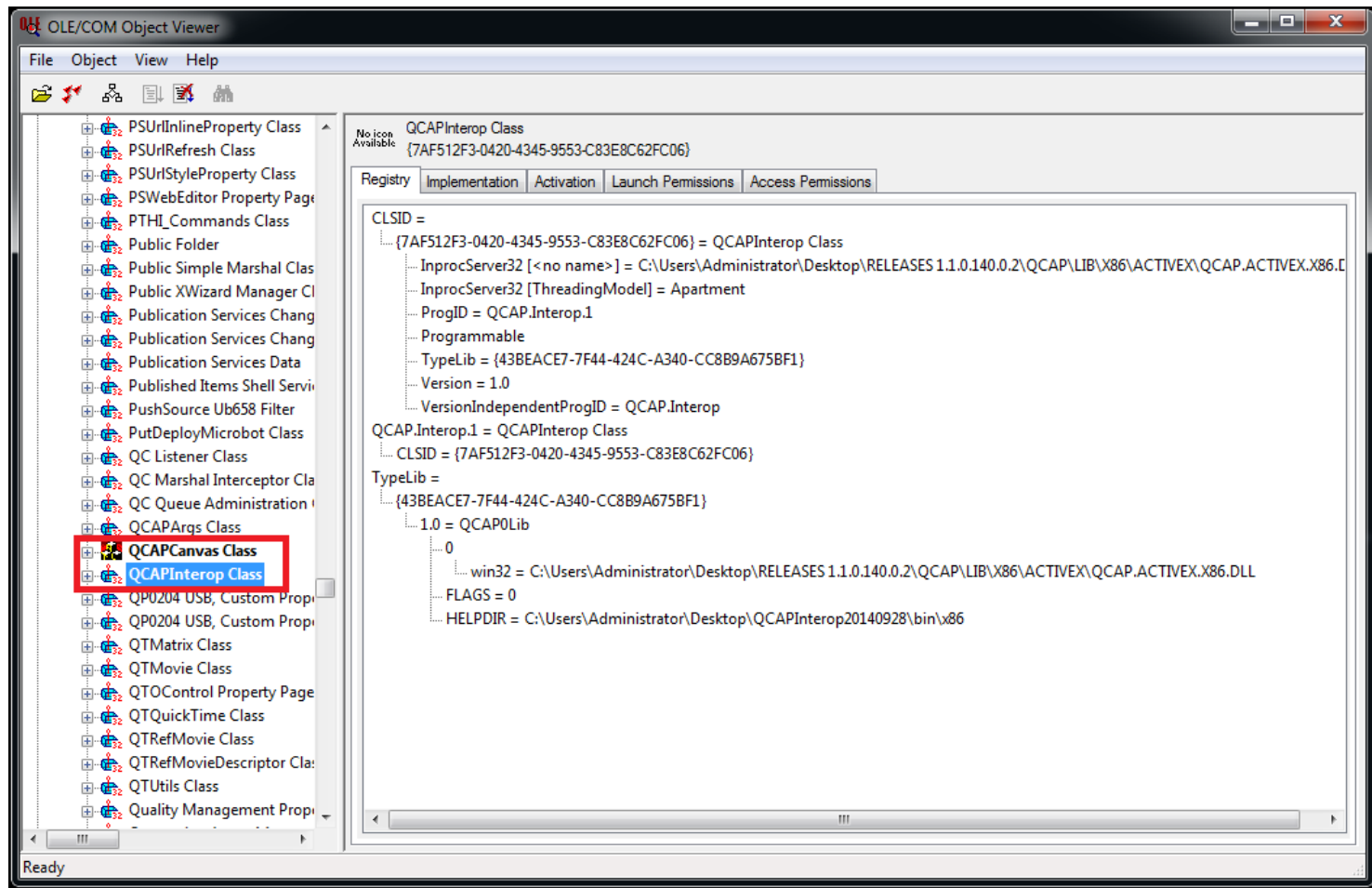
# Build QCAP ACTIVEX CONTROL application

2. Please register **QCAP.ACTIVEX.X86.DLL** .



# Build QCAP ACTIVEX CONTROL application

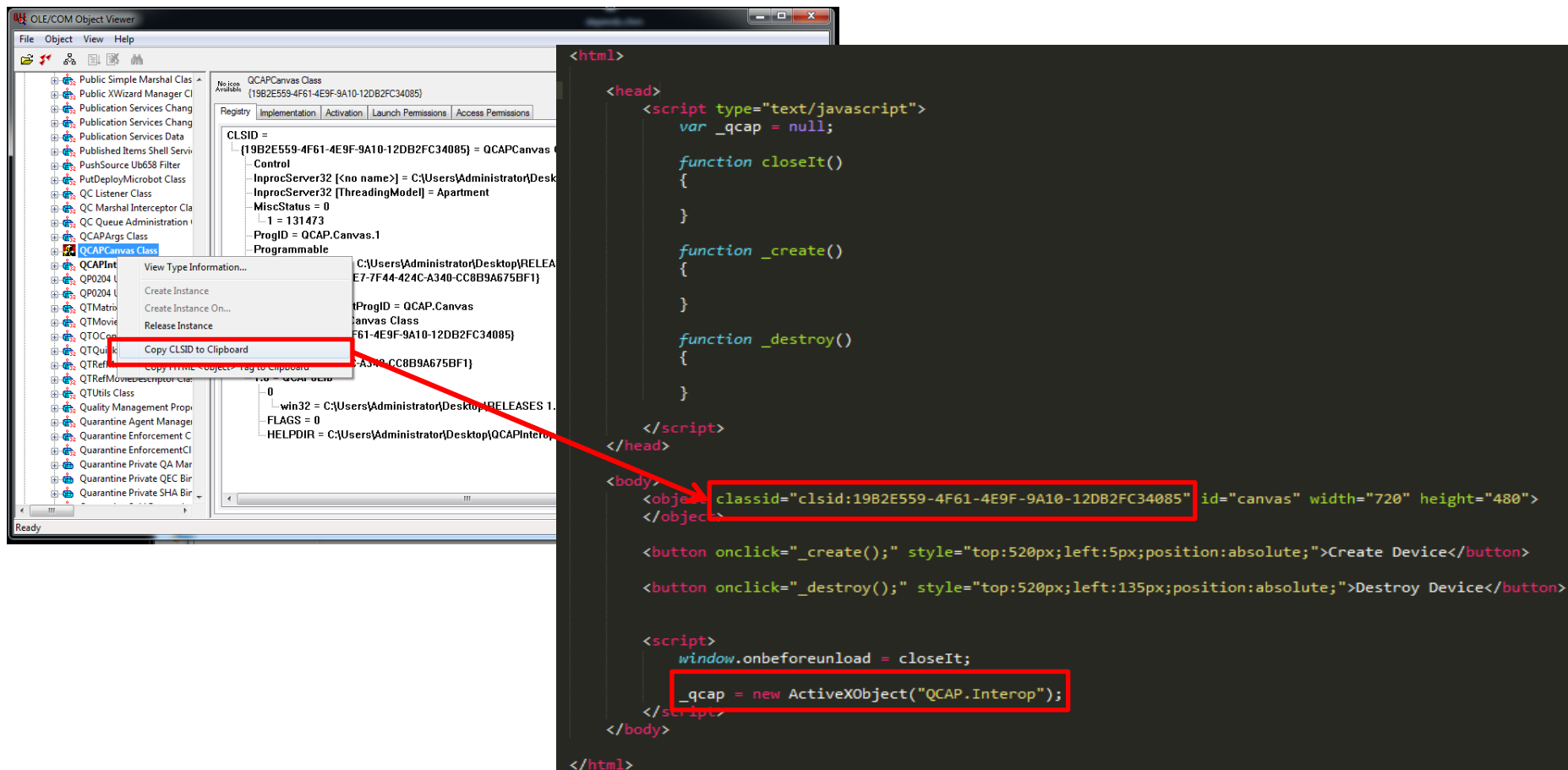
3. After register QCAP.ACTIVEX.X86.DLL, Developer will find **QCAPCanvas Class** and **QCAPInterop Class** in OLE/COM Object Viewer.



# Build QCAP ACTIVEX CONTROL application

3. Using javascript to develop QCAP ActiveX Control.

**Step1.** We have add a ActiveXObject("QCAP.Interop") and 2 buttons and 1 canvas in javascript framework, the canvas from **QCAPCanvas Class**.



The image shows two side-by-side screenshots illustrating the development of a QCAP ActiveX control.

The left screenshot displays the **OLE/COM Object Viewer** window. The **QCAPCanvas Class** is selected in the left pane. The right pane shows the class's properties, including the **CLSID** (19B2E559-4F61-4E9F-9A10-12DB2FC34085) and the **ProgID** (QCAP.Canvas.1). A red box highlights the **Copy CLSID to Clipboard** option in the context menu.

The right screenshot shows an **HTML** document with JavaScript code. The code defines a **\_qcap** variable, **\_create()**, **\_destroy()**, and **closeIt()** functions. A red box highlights the **classid** attribute in the **<object>** tag, which is set to the CLSID from the previous screenshot. Another red box highlights the **new ActiveXObject("QCAP.Interop")** line in the JavaScript code.

```
<html>
<head>
  <script type="text/javascript">
    var _qcap = null;

    function closeIt()
    {
    }

    function _create()
    {
    }

    function _destroy()
    {
    }
  </script>
</head>
<body>
  <object classid="clsid:19B2E559-4F61-4E9F-9A10-12DB2FC34085" id="canvas" width="720" height="480">
</object>

  <button onclick="_create();" style="top:520px;left:5px;position:absolute;">Create Device</button>

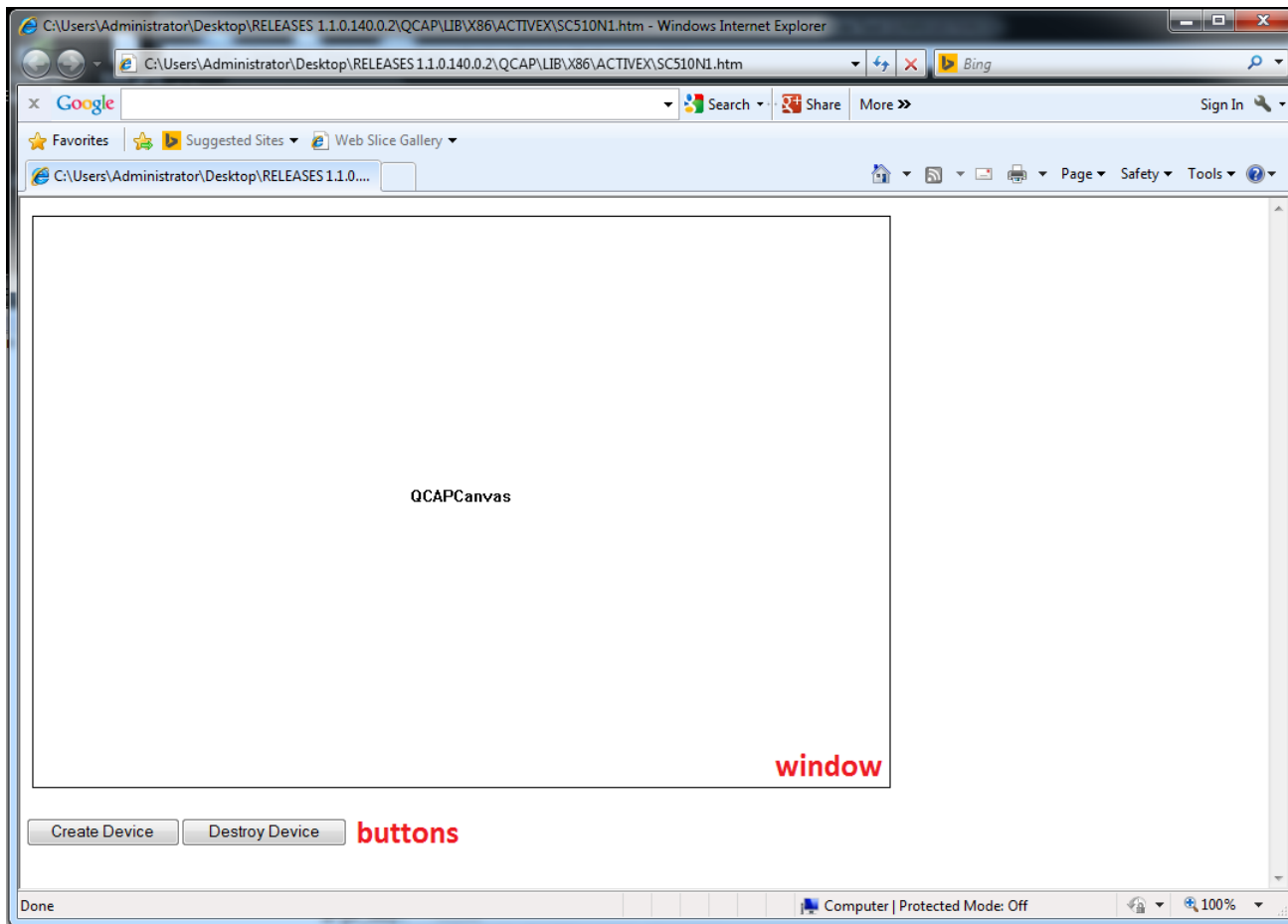
  <button onclick="_destroy();" style="top:520px;left:135px;position:absolute;">Destroy Device</button>

  <script>
    window.onbeforeunload = closeIt;
    _qcap = new ActiveXObject("QCAP.Interop");
  </script>
</body>
</html>
```

# Build QCAP ACTIVEX CONTROL application

3. Using javascript to develop QCAP ActiveX Control.

**Step2.** After Step1. developer will find 1 canvas and 2 buttons in HTM file.



# Build QCAP ACTIVEX CONTROL application

## 3. Using javascript to develop QCAP ActiveX Control.

**Step3.** Function `_create()` will run when button “create device” be click. We need add parameter by using the help object **Append** of `ActiveXObject(“QCAP.Args”)` before using `QCAP_CREATE`. Also developer get parameter by using the help object **GetAt**.

```
QRESULT QCAP_CREATE( CHAR *   pszDevName,  
                     UINT     iDevNum,  
                     HWND     hAttachedWindow,  
                     PVOID *  ppDevice, output  
                     BOOL     bThumbDraw = FALSE  
                     BOOL     bMaintainAspectRatio = FALSE )
```

```
<html>  
  
<head>  
  <script type="text/javascript">  
    var qcap = null;  
    var _dev = null;  
  
    function closeIt()  
    {  
      if(_dev != null)  
      {  
        _destroy();  
      }  
    }  
  
    function _create()  
    {  
      //create  
      var args_create = new ActiveXObject("QCAP.Args");  
      args_create.Append("SA7160 PCI");  
      args_create.Append(0);  
      args_create.Append(canvas.Handle);  
      args_create.Append(null); // OUT  
      args_create.Append(1);  
      var ret = _qcap.QCAP_CREATE(args_create);  
      _dev = args_create.GetAt(3);  
  
      //run  
      var args_run = new ActiveXObject("QCAP.Args");  
      args_run.Append(_dev);  
      var ret = _qcap.QCAP_RUN(args_run);  
    }  
  </script>  
</head>  
</html>
```

# Build QCAP ACTIVEX CONTROL application

3. Using javascript to develop QCAP ActiveX Control.

**Step4.** Function `_destroy()` will run when button “destroy device” be click. In this button, we want to destroy the device, before destroy driver must stop driver first.

```
function _destroy()
{
    if(_dev != null)
    {
        //stop
        var args_stop = new ActiveXObject("QCAP.Args");
        args_stop.Append(_dev);
        var ret = _qcap.QCAP_STOP(args_stop);

        //destroy
        var args_destroy = new ActiveXObject("QCAP.Args");
        args_destroy.Append(_dev);
        var ret = _qcap.QCAP_DESTROY(args_destroy);

        _dev = null;
    }
}
```

# Build QCAP ACTIVEX CONTROL application

3. Using javascript to develop QCAP ActiveX Control.

**Step5.** Make sure device will be destroy when browser close.

```
1 <html>
2
3 <head>
4 <script type="text/javascript">
5     var _qcap = null;
6     var _dev = null;
7
8     function closeIt()
9     {
10         if(_dev != null)
11         {
12             _destroy();
13         }
14     }
15
16     function _create()
17     {
18         //create
19         var args_create = new ActiveXObject("QCAP.Args");
20         args_create.Append("SA7160 PCI");
21         args_create.Append(0);
22         args_create.Append(canvas.Handle);
23         args_create.Append(null); // OUT
24         args_create.Append(1);
25         var ret = _qcap.QCAP_CREATE(args_create);
26         _dev = args_create.GetAt(3);
27
28         //run
29         var args_run = new ActiveXObject("QCAP.Args");
30         args_run.Append(_dev);
31         var ret = _qcap.QCAP_RUN(args_run);
32     }
33
34     function _destroy()
35     {
36         if(_dev != null)
37         {
38             //stop
39             var args_stop = new ActiveXObject("QCAP.Args");
40             args_stop.Append(_dev);
41             var ret = _qcap.QCAP_STOP(args_stop);
42
43             //destroy
44             var args_destroy = new ActiveXObject("QCAP.Args");
45             args_destroy.Append(_dev);
46             var ret = _qcap.QCAP_DESTROY(args_destroy);
```

# Build QCAP ACTIVEX CONTROL application

3. Using javascript to develop QCAP ActiveX Control.

**Step6.** After all steps, developer will see preview from capture card when button “create device” be click.

