



Impact ActiveX Visual Basic 6 Getting Started

Publication # 843-0126 Rev 11.8

Disclaimer

Datalogic Automation, Inc. makes no representations or warranties for merchantability or fitness for any particular purpose, regarding Datalogic's software or hardware. Datalogic Automation, Inc. shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this publication or its contents.

Datalogic Automation, Inc. reserves the right to revise this publication from time to time and to make changes in the content hereof without obligation to notify any person of such revision or changes.

Under the copyright laws, neither this publication nor the software may be copied, photocopied, reproduced, or reduced to any electronic medium or machine-readable form, in whole or in part, without the prior written consent of Datalogic Automation, Inc.

Datalogic and the Datalogic logo are registered trademarks of Datalogic S.p.A. in many countries, including the U.S.A. and the E.U.

All brand and product names mentioned herein are for identification purposes only and may be trademarks or registered trademarks of their respective owners.

Impact, Vision Program Manager, VPM, Control Panel Manager, and CPM are trademarks of Datalogic Automation, Inc.

Patent. See www.patents.datalogic.com for patent list.

This product is covered by one or more of the following patents: Utility patents: US6,512,218 B1; US6,616,039 B1;

US6,808,114 B1; US6,997,385 B2; US7,053,954 B1; US7,387,246 B2; US8,058,600 B2; EP996,284 B1; EP999,514 B1; EP1,014,292 B1; EP1,128,315 B1; EP1,396,811 B1; EP1,413,971 B1; JP4,435,343 B2; JP4,571,258 B2.

Additional patents pending.

Copyright © 2015 Datalogic Automation, Inc. All Rights Reserved.

Technical Support

If you have technical questions about the operation of any Datalogic product, contact your distributor or Datalogic. Please have the following information available before you call:

- The model and serial number of the Datalogic vision device, located on the bottom of the unit. The version number of the Impact software you are running, found in the title bar of the software.
- The type and version number of the operating system software you are running on the client computer.

Chapter 1: Using VB6 with Impact ActiveX

Visual Basic 6 Programming Environment	1-1
Impact dll	1-2
Project Files	1-2
Sample Executable Files	1-2
Sample Vision Program File	1-2
Device Connection Interface	1-3
Sample Program Code	1-4
Device Control Interface	1-5
Sample Program Code	1-7

Using VB6 with Impact ActiveX

This manual documents the Visual Basic sample programs provided by Datalogic that uses the Impact ActiveX interface. The interface allows Visual Basic programmers to access data and events on Datalogic Cameras and the machine vision systems.

Using ActiveX, a Visual Basic programmer can monitor and operate a Datalogic Impact device, including the ability to read and write tool and task inputs and outputs, load and unload programs, trigger the device to snap an image, and respond to device events.

This manual shows several examples of a Visual Basic implementation of the Impact ActiveX interface.

For a more general overview of Impact ActiveX installation, configuration, and use, refer to the Impact ActiveX Reference Guide (Publication # 843-0119).

NOTE: This manual does not provide detailed operation about Datalogic Impact Cameras. It assumes you have some knowledge of how vision programs and Impact tools and tasks are structured, how to access tool and task data, and that you understand Impact terminology.

For more details about using Impact Vision Program Manager (VPM) and Datalogic cameras, refer to the Datalogic Hardware Guides and the Impact Software Reference Guide.

NOTE: Impact ActiveX cannot be used on the Datalogic model A30, T4x-Series, or P-Series Smart Cameras.

Visual Basic 6 Programming Environment

To initially configure the Datalogic Impact camera and create Impact vision programs and control panels, you need to install Impact software, including Vision Program Manager (VPM) and Control Panel Manager (CPM) on a host computer. If you do not intend to create or modify vision programs, you do not need to install the Impact software (VPM and CPM) on the host computer. Refer to the Impact Software Reference Guide for the client computer's requirements.

The Impact Active X Application Programming Interface (API) lets you write a program in Visual Basic 6 to access vision device data by using the necessary function calls. That data can then be sent directly from an Datalogic camera to other devices, such as a Quality Control monitoring system.

Several sample programs are also provided to help you use the APIs. This includes the Visual Basic code, the executable program, and a sample Impact vision program. To view and modify the program code, you need to install Visual Basic 6 or greater. Refer to the Visual Basic software installation and configuration guide for details on installing Visual Basic software. To access these sample programs, you need to install the Impact ActiveX components on the host computer.

To access the Impact ActiveX API, you must install Impact Software Release 8.3.0 or greater on the Datalogic vision device.

You must also include the Impact dynamic link library (Impact.dll) in your program or project. This dll is installed by the Impact ActiveX install. See "Impact dll" on page 1-2.

You may copy, modify, and use this sample program code, subject to the terms of Datalogic's Software End User License Agreement.

Impact dll

The Impact ActiveX dynamic link library (Impact.dll) is installed and registered when Impact ActiveX is installed. When you create a new project, you must include the dll so you can use the method calls in your program. These instructions are for Visual Basic 6.

To include the dll in your program

1. Open your project.
2. From the menu, select Project.
3. Select Reference.
4. Scroll down the list to Impact 1.0 Type Library. Check the check box next to the entry.
5. Click OK

Project Files

All the files you need to create a working program are included in the sample project files. The sample project files are installed in a directory under the "install folder":

C:\install folder\IMPACT\Vision Device APIs\ActiveX\Sample Applications\VB6ActiveX\Source

APPDEMO.FRM - The form file for the sample device control interface program.

ENTRYFORM.frm - The form file for the sample device selection interface program.

VB6Demo.vbp: The Visual Basic code for the sample program.

Sample Executable Files

A sample Visual Basic executable program named VB6ActiveXSample.exe is included. This sample file is installed in a directory under the "install folder":

C:\install folder\IMPACT\Vision Device APIs\ActiveX\Sample Applications\VB6ActiveX

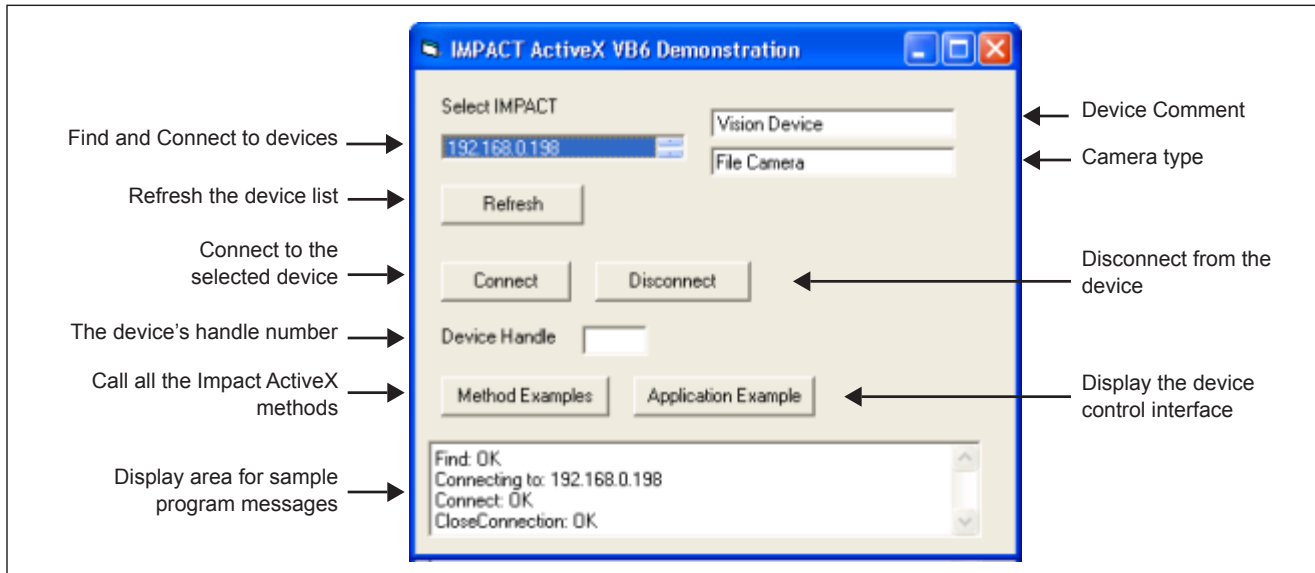
Sample Vision Program File

A sample Impact vision program name VB6ActiveXSample.vp is included for use with the sample executable program. This file must be loaded on the vision device or Emulator for the sample VB6 program to work. It is loaded automatically when you run the sample program. The sample executable will display information when you connect to the Emulator or vision device. This sample vision program file is installed in a directory under the "install folder":

C:\install folder\IMPACT\Vision Device APIs\ActiveX\Sample Applications\VB6ActiveX

Device Connection Interface

The sample programs provided let you monitor and control a Datalogic Impact camera. When you start the sample program named VB6ActiveXSample.exe, the device connection interface is displayed. This section briefly explains the interface.



NOTE: All fields on this interface are display only. Even if you modify a field, the program will overwrite it.

Select Impact

This field lists all the Datalogic vision devices on the network. Use the up and down scroll arrows to the right of the field to display the entire list of devices. Click the Refresh button to refresh this device list.

Device Comment

This field displays the text from the selected device's comment information.

Camera Type

This is the selected device's type or the camera type connected to the device if it is a C-Series device.

Refresh

When you click this button, the sample program searches the network for vision devices, then displays the IP Address of all the devices that respond.

Connect

Click this button to connect to the device currently selected in the device list. The result of the connect operation is displayed in the program message area. If security is enabled on the device, an error message "Connect: User not authorized to perform this function" will be displayed.

Disconnect

Click this button to disconnect from a currently connected device. The result of the disconnect operation is displayed in the program message area.

Device Handle

This field displays the Device Handle number of the currently connected device. This number is used in the program code to refer to the device when making method calls.

Method Examples

When you click this button, the sample program makes a call using each of the Impact ActiveX methods. The result of the calls is shown in the program message area.

Application Example

When you click this button, the sample program displays the device control interface. See the next section for more details.

Message Display Area

Program messages are displayed in this area.

Sample Program Code

Refresh

Here is the section of the Main Visual Basic program code for the Device Connection Interface that refreshes the Device List when you click the Refresh button. See the rest of the program for variable and procedure declarations. It uses the Impact method "Find" among others. Refer to the Impact ActiveX Reference Guide for more details.

```

'
' Show information for selected IP address
'
Private Sub ShowDetails()
    impDetails.Index = IPAddressList.ListIndex
    DescriptionText.Text = impDetails.deviceDescription
    ModelText.Text = impDetails.cameraModel
    ipAddr = IPAddressList.Text
End Sub
'
' Get list of IMPACTs on network and display as list
'
Private Sub RefreshList()
    Dim lRes As ImpactLib.ImpactErrorCode
    Dim iCnt As Integer

    lRes = impDisc.Find(impDetails)
    Call DisplayResult("Find", lRes)

    Call IPAddressList.Clear
    If impDetails.Size > 0 Then
        For iCnt = 0 To impDetails.Size - 1
            impDetails.Index = iCnt
            Call IPAddressList.AddItem(impDetails.IPAddress, iCnt)
        Next iCnt
        IPAddressList.ListIndex = 0
        Call ShowDetails
    End If
End Sub

```

```

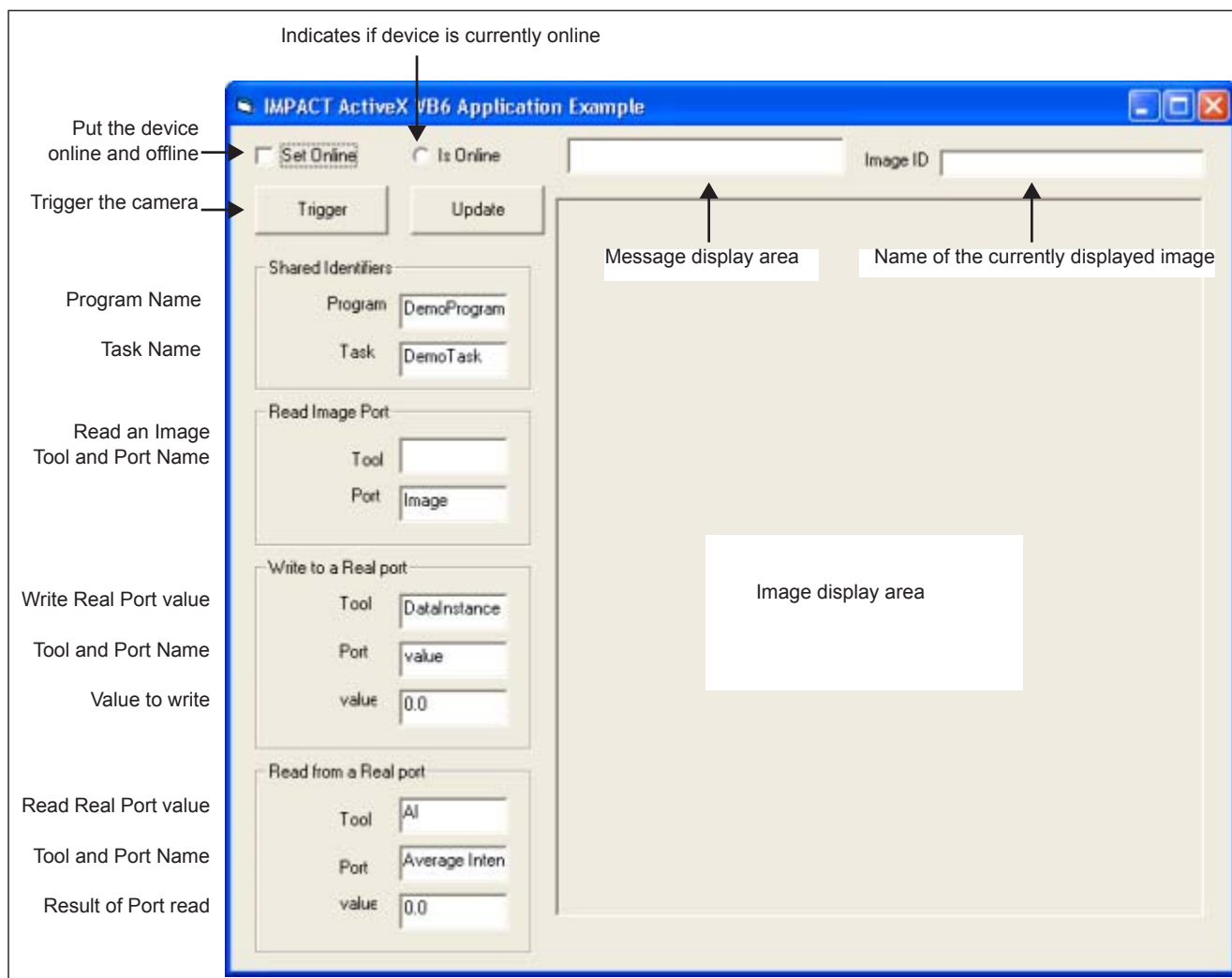
        End If
    End Sub
    '
    ' Refresh the IP Address list
    '
    Private Sub RefreshButton_Click()
        Call RefreshList
    End Sub

```

Device Control Interface

The sample program device control interface lets you monitor and control a Datalogic Impact camera. This interface is displayed when you click the Application Example button in the Device Connection Interface. You must use the Device Connection Interface to connect to a device to display this interface.

This section briefly explains the interface.



NOTE: All name entry fields on this program interface are case sensitive. The name entered in the field must match the name in the inspection program.

Set Online

When this check box is checked, the device is put online and the Is Online radio button is selected. When it is unchecked, the device is put offline. This check box must be checked for the Trigger button to create triggers.

Is Online

When this radio button is selected, it means the device is online.

Message Display Area

Messages about the most recent action are displayed in this area.

Image ID

This field contains the name of the image currently displayed in the Image display area.

Trigger

Click this button to trigger the device, capture an image, and run the sample vision program. The Set Online check box must be checked for the trigger to succeed.

Update

Click this button to refresh the displayed port values, including the image.

Image Display Area

The image indicated in the Read Image Port fields is displayed in this area. This image placement and size are determined by the GetImagePort method call.

Shared Identifiers

- Program - Enter the name of the program that contains the task you want to access
- Task - Enter the name of the task that contains the tool you want to access.

Read Image Port

The image contained in the tool and port named here is displayed in the Image Display area.

If the Tool field is blank and the Port name is "Image," the image will be read from the Image Port of the task named in Shared Identifiers.

- Tool - Enter the name of the tool that contains the image here.
- Port - Enter the name of the tool port that contains the image you want to display.

Write to a Real Port

Use this function to write a real value to a tool's port. After you enter the value, click the Trigger button to send the value to the tool port.

- Tool - Enter the name of the tool that has the port you want to access.
- Port - Enter the name of the tool port that contains the value you want to write. You can only write to a tool input port that has a type of Real.
- Value - Enter the value that you want to write to the tool port. You may enter a whole or decimal number.

Read from a Real Port

Use this function to read a real value from a tool's port. After you enter the tool and port name, click the Trigger or Update button to read the value from the tool port. You can read a tool's input or output ports. The value is displayed in the value field.

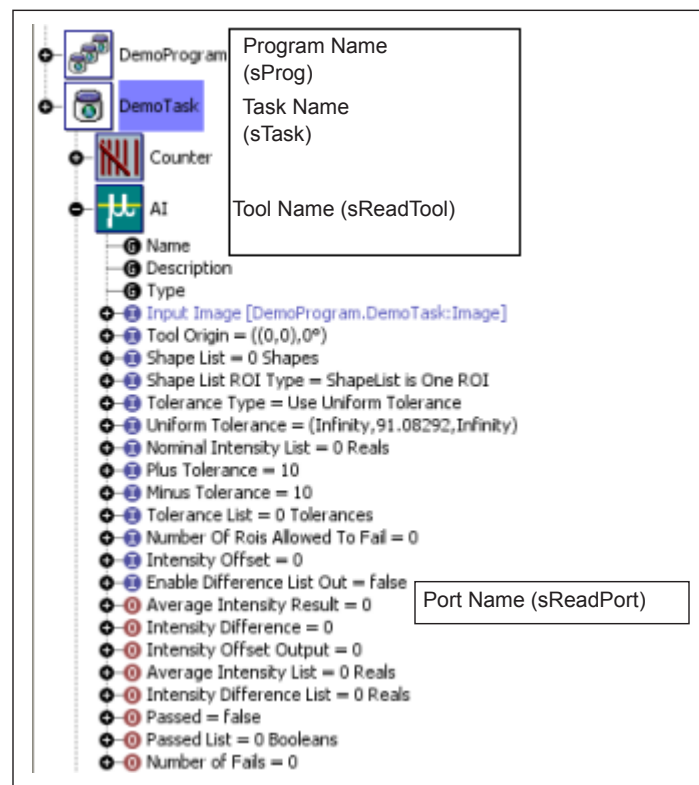
- Tool - Enter the name of the tool that has the port you want to access.
- Port - Enter the name of the tool port that has a type of Real.
- Value - The value read from the tool port is displayed in this field.

Sample Program Code

Vision Program

When you connect to a device with the sample program, it automatically copies a sample vision program to the device and loads it. The program, task, and port names in the VB API call to read and write data must match the names in the vision program.

When you create your own VB program, you need to use the names from the vision program to access the data. For example, here is a portion of the sample vision program referred to in the code.



Read from a Real Port

Here is the section of the Device Control Interface program code that runs the Read from a Real Port function. See the rest of the program for variable and procedure declarations. It uses the Impact API call "GetRealPort" among others. Refer to the Impact ActiveX Reference Guide for more details.

```
' Get the program, task, tool and port names for reading a float
Call GetProgTask
sReadTool = ReadTool.Text
sReadPort = ReadPort.Text

'
' Get the float value and display it
'
lRes = impactOper.GetRealPort(Form1.theForm1.GetDevHnd(), sProg, sTask,
sReadTool, sReadPort, "", fData)
Call ShowStatus("GetRealPort", lRes)

If lRes = ImpactLib.ImpactErrorCode.ImpactOK And bLoaded Then
    ReadValue.Text = Str(fData)
End If
```

A C T I V E X V B 6 R E F E R E N C E

Index

A

API 1-1
Application example 1-4

C

Camera
 type 1-3
Connect 1-3
CPM 1-1

D

Device
 comment 1-3
 handle 1-4
 select 1-3
Disconnect 1-3
Dll
 IMPACT 1-2

E

Environment
 programming 1-1

F

Files

executable 1-2
project 1-2
vision program 1-2

I

Image
 display 1-6
IMPACT
 dll 1-2
Interface
 device connection 1-3

M

Message display area 1-4
Method examples 1-4

R

Refresh 1-3

S

Select IMPACT 1-3
Software
 version required 1-1

V

Version
 software required 1-1
VPM 1-1

