IB-SDK-00003

Ver. 3.0.0 2012-04-04

IBSDK Quick Start Tutorial for C# 2010

Copyright @2012, Integrated Biometrics LLC. All Rights Reserved



1 QuickStart Project – C# 2010 Example

Follow these steps to setup the C# example:

- 1. Open Visual Studio and click New Project...
- 2. If not already, select Visual C# > Windows > Windows Forms Application
- 3. At the bottom of the selection panel, change the Name and Solution Name to

QuickStart



4. Set the file location to whatever you like and click **OK**.



- 5. Next, click **Tools > Choose Toolbox Items...**
- 6. Select the .NET Framework Components tab (if necessary) and locate EnrollTemplateControl and FingerReaderControl. In the items list and make sure the checkboxes are checked. If you can't find these controls in the list of .NET Framework Components, then click the Browse... button and go to C:\Program Files (x86)\Integrated Biometrics\Web Control\64 (or C:\Program Files (x86)\Integrated Biometrics\Web Control\32 if you are developing a 32 bit application) and doubleclick IntegratedBiometrics.Enroll3Finger to register the first component. Then click Browse... again and repeat the process for IntegratedBiometrics.FingerReader. If you cannot find the Web Control folder or the components in it, then please go to the section, "Installing the .NET/ActiveX Programming Controls". Once the boxes are checked, click the OK button.

Silverlight Components	mponents	System.Activities Components						
.NET Framework Con	nponents	nts COM Components		WPF Components				
Name	Namespace		Assembly Name		Directory			
CrystalReportSource	CrystalDecisions.Web		CrystalDecisions.Web		Global Asse			
CrystalReportViewer	CrystalDecisions.Web		CrystalDecisions.Web		Global Asse			
ReportServiceBase	se CrystalDecisions.Web.Services		CrystalDecisions.Web		Global Asse			
ServerFileReportServi	CrystalDecisions.Web.Services		CrystalDecisions.Web		Global Asse			
CrystalReportViewer	CrystalDecisions.Windows.Fo		CrystalDecisions.Win		Global Asse			
EnrollTemplateControl	IntegratedBiomet	trics.Enroll3	IntegratedBiometrics		C:\Program			
FingerReadControl	IntegratedBiometrics.FingerB		IntegratedBiometrics		C:\Program			
ActionsPane	Microsoft Office Tools		Microsoft.Office.Tool		Global Asse			
ActionsPane	Microsoft Office Tools		Microsoft.Office.Tool		Global Asse			
Chart	Microsoft Office Tools Excel		Microsoft Office Tool		Global Asse			
Chart	Microsoft Office Tools Excel		Microsoft Office Tool		Global Asse			
	Microsoficomeet		Microsoficom	·		-		
ilter:						<u>C</u> lear		
AccessDataSource Browse Browse								



- In the Solution Explorer, rename "Form1.cs" to "frmQuickStart.cs" (when asked about renaming all references in the project as well, click Yes).
- 8. Click on the form in the designer view, in the **Properties** window:
- 9. Make sure the Name property is *frmQuickStart*
 - a. Change the **Size** property to 550,350
 - b. Change the **Text** property to "Quick Start 3.0"

10. Click on the **Toolbox** tab at the far left of the design environment. Under **General** (at the very bottom), double-click *EnrollTemplateControl* and *FingerReadControl*. This will add the **Integrated Biometrics** ActiveX Controls to your project's form. Click on the larger control at the top to highlight it. In the **Properties** window:

- a. Change the Name property from enrollTemplateControl1 to ibEnroller
- b. Change the Location property to 18,7
- c. Change the **Size** property to 499,121
- 11. Now click on the smaller of the two controls. In the **Properties** window:
 - a. Change the Name property from *fingerReadControl1* to *ibVerifier*
 - b. Change the Location property to 24,137
 - c. Change the Size property to 105,121

😎 QuickStart3_0 - Microsoft Visual Studio (Admini	istrator)		-		100		×
File Edit View Project Build Debug Tear	m Data Tools VMware Test Windo	w Help					
1 51 - 51 - 12 🖬 🗃 🕹 🗛 🖎 💌 - 0	- 🖾 - 🖾 🕨 Debug 🔹 🕅	enhancingimage	- 🗠 🕾 🗟 🐋 🐼 🖬 🔮	P + _		Colores and	
	1 mm 205 00 00 1 & At At At Im P					Solution Expl	orer window
	100 100 ++ ++ ++ 2 2+ 2+ 2+ 1010		I man Manual Mindaux	<u>_</u>			
	- Wai =		signer view window)			
🗃 Toolbox 🔫 म 🗙	frmQuickStart.cs [Design] ×					 Solution Explorer 	- † ×
🚽 🖉 SerialPort 🔷		/				🕒 🗿 💽 🗉	3 &
👸 🚟 ServiceController	🖳 Quick Start 3.0	/				Solution 'Quick'	Start3_0 (1 project)
🗧 じ Timer						QuickStart	3_0
A Printing						Propertie	es 👌
Rev Pointer						Reference	.es
PageSetupDialog		A .				⊿ ImQuic	kstart.cs JuickStart Designer.cs
PrintDialog						1 frmC	JuickStart.resx
PrintDocument						Program	1.CS
PrintPreviewControl		S					
PrintPreviewDialog							
▲ Dialogs							
Pointer							
ColorDialog						Solutio 🏹 Te	eam E 🥺 Class Vi
FolderBrowserDialog						Properties	+ # ×
FontDialog						ibVerifier Integrated	Biometrics FingerRe +
2 OpenFileDialog	P 9					AL DO AL	
SaveFileDialog						i i i i i i i i i i i i i i i i i i i	2
WPF Interoperability						GenerateMembe	True ^
▶ Reporting						ImageSize	1013/0
Visual Basic PowerPacks	òò					ImageXSize	252
Pointer						ImeMode	NoControl
PrintForm	Taultau		_	Incompation M/in		IndexingLevel	0
LineShape	Error List Toolbox			ropercies win	40W • 4	> Location	10, 180 =
OvalShape	0 Warnings 0 Me	issages				Locked	False
RectangleShape	Description		File Li	ne Column	Project	Margin	3, 3, 3, 3
DataRepeater						MaximumSize	0, 0
▲ General =						MinimumNFIQ	5
Pointer						MinimumSize	0,0 🔻
EnrollTemplateControl						(ApplicationSetting	s)
FingerReadControl						Maps property settin	gs to an application
20 I oolbox Resource View	📑 Error List 📄 Output 📑 Find Resul	s 1 🌇 Find Symbol Results				comgaration me.	
Ready							
C							

- Now add two buttons to the form (Double-click the **Button** object in the **Toolbox** window twice). In the Properties window:
 - a. Change the Name property to btnCompare
 - b. Change the Enabled property to False
 - c. Change the **Location** property to 442,203
 - d. Change the **Text** property to "Match"
- 13. Select the next unmodified button on the form, in the **Properties** window:
 - a. Change the Name property to "btnEnroll"
 - b. Change the **Location** property to 442,174
 - c. Change the **Text** property to "Enroll"
- 14. As above add two more buttons to the form. In the **Properties** window:
 - a. Change the Name property to "btnCompareCancel"
 - b. Change the Location property to 442,267
 - c. Change the **Text** property to "Cancel"
 - d. Change the **Visible** property to False
- 15. Select the next unmodified button on the form, in the **Properties** window:
 - a. Change the Name property to "btnEnrollCancel"
 - b. Change the **Location** property to 442,235
 - c. Change the **Text** property to "Cancel"
 - d. Change the Visible property to False
- 16. Now add two trackbar objects (found in the **All Windows Forms** section at the top of the toolbox) to the form. In the **Properties** window:
 - a. Change the Name property to "tbSecurityLevel"
 - b. Change the Location property to 188,218
 - c. Change the Maximum property to 7



- d. Change the Minimum property to 1
- e. Change the Size property to 159,45
- f. Change the Value property to 4

NOTE: This object will control the overall strictness required for a successful match.

- 17. Select the next unmodified trackbar on the form, in the Properties window:
 - a. Change the Name property to tbNFIQLevel
 - b. Change the Location property to 188,152
 - c. Change the Maximum property to 5
 - d. Change the Minimum property to 1
 - e. Change the Size property to 159,45
 - f. Change the Value property to 3

NOTE: This object will control minimal image quality (based on NFIQ scoring) before a print is accepted (turns green and moves to the next step in the ActiveX control). In this case, a lower number means a more stringent quality requirement.

- Now add seven labels to the form (Double-click the Label object in the Toolbox window seven times). In the Properties window:
 - a. Change the Name property to *lblEnrollStatus*
 - b. Change the AutoSize property to False
 - c. Change the BackColor property to ActiveCaption
 - d. Change the Font property to Microsoft Sans Serif, 10 pt
 - e. Change the Location property to 117,268
 - f. Change the Size property to 300,40



- g. Change the **Text** property to Press the *Enroll Button to Start*
- h. Change the TextAlign property to MiddleCenter

NOTE: No other labels require special naming. They literally are static labels for display enhancement. Tab order and other considerations are equally unimportan t.

19. Select the next unmodified label on the form, in the **Properties** window:

- a. Change the **Location** property to 209,139
- b. Change the Text property to NFIQ Enrollment Quality
- 20. Select the next unmodified label on the form, in the Properties window:
 - a. Change the **Location** property to 206,205
 - b. Change the Text property to Matching Security Level
- 21. Select the next unmodified label on the form, in the **Properties** window:
 - a. Change the **Location** property to 195,176
 - b. Change the **Text** property to "1"
- 22. Select the next unmodified label on the form, in the **Properties** window:
 - a. Change the Location property to 328,177
 - b. Change the **Text** property to "5"
- 23. Select the next unmodified label on the form, in the **Properties** window:
 - a. Change the **Location** property to 195,243
 - b. Change the **Text** property to "1"
- 24. Select the next unmodified label on the form, in the **Properties** window:
 - a. Change the **Location** property to 327,243
 - b. Change the **Text** property to "7"



								_
😋 QuickSta	rt3_0 - Microsoft Visual Studio							- • ×
File Edit	View Project Build Debug Te	am Data Format Tools	VMware Test Window Help					
i 🛅 - 🔛	• 📂 🔜 🥔 🔏 📭 🖭 🄊 🗣	(* - 📮 - 🖳 🕨 Debug	 enrollTemplateControl 	- 🗠 🕾 🚮 🕾 🔊	💁 🖸 * 🖕			
「菫」に	종태 교 아파 문화 路道	* ~ 관 및 및 을 찾	밝 랴 단 [단] 👊 🍕 💷 📑 🚽					
	3 强 🖄 🗈 🍙 🖄 🎫 🕲 1 8	110 -						
	- 0 -							- 1 - 1
	x * + .	frmQuickStart.cs [Design]	TrmQuickStart.cs*			-	Solution Explorer	• • ~
ata 💭	GroupBox							
	Heiperovider HScrollPar	🛃 Quick Start 3.0					Solution QuickStart3_0	(1 project)
â 🎽	ImageList						Properties	
A	Label	0.6 0					References	
A	LinkLabel						⊿ Image: a frmQuickStart.cs	s
-	ListBox		N R				frmQuickSta	rt.Designer.cs
222	ListView						Program.cs	TLIESK
	MaskedTextBox							
2	MenuStrip		NFIQ Enrollment Quality					
9	MessageQueue							
	MonthCalendar		1 5	Enroll				
	NotifyIcon		Matching Security Level	Match			Nolutio 🏹 Team E	💐 Class Vi
10	NumericUpDown	=		Cancel			Properties	- ∓ ×
<u>.</u>	OpenFileDialog		1	Cancer			label5 System.Windows.For	ms.Label 🔹
	PageSetupDialog		Press the "Enroll" Button to Star	Cancel			2 21 🗉 🖌 🖂	
	Panel						▷ Location 327, 2	43 ^
	PerformanceCounter				-		Locked False	
	PictureBox	-					 Margin 3, 0, 3, 	0
	PrintDialog						MaximumSize 0, 0	
	PrintDraviewControl	-					MinimumSize 0, 0 Medifiers Drivete	
	PrintPreviewDialog	Error List				- t >	Padding 0.0.0	0
	Process	3 0 Errors 1 0 Warn	ings i 0 Messages				RightToLeft No	
	ProgressBar	Description		File	Line Column	Project	> Size 13, 13	E
-	PropertyGrid						TabIndex 19	
•	RadioButton						Tag	
1	RichTextBox						lext 7	-
3	SaveFileDialog						lag	l with the
* To	olbox 🔄 Resource View	🕆 Error List 🔲 Output	Find Results 1 K Find Symbol Results				object.	r with the
Ready							<u>⊒</u> 13 x 13	

25. Double click on the top of the **Quick Start** form in the designer, this will put you in the code editor (and register the "form load" event in the event handler). Add the following statements to the **frmQuickStart_Load** subroutine:

```
enrollTemplate = new byte[NFingers][]; // Placeholder for 10
templates
```

 $\ensuremath{//}$ Move the cancel buttons underneath their appropriate counterparts.

```
btnEnrollCancel.Location = btnEnroll.Location;
btnCompareCancel.Location = btnCompare.Location;
```

26. Also, let's go ahead and define some constants and variables that we will use throughout the project. Add the following lines just above the **frmQuickStart_Load** subroutine:

```
const int NFingers = 10;
private byte[][] enrollTemplate = null;
string[] FingerName =
```



{ "L5", "L4", "L3", "L2", "L1", "R1", "R2", "R3", "R4", "R5" };

Your code should now look something like this:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
namespace QuickStart
{
   public partial class frmQuickStart : Form
   {
       const int NFingers = 10;
       private byte[][] enrollTemplate = null;
       string[] FingerName = { "L5", "L4", "L3", "L2", "L1", "R1", "R2", "R3", "R4",
"R5" };
       public frmQuickStart()
       {
           InitializeComponent();
       }
       private void frmQuickStart_Load(object sender, EventArgs e)
       {
           enrollTemplate = new byte[NFingers][]; // Placeholder for 10 templat
es
           // Move the cancel buttons underneath their appropriate counterparts.
           btnEnrollCancel.Location = btnEnroll.Location;
           btnCompareCancel.Location = btnCompare.Location;
       }
   }
}
```

27. Click on the **frmQuickStart.cs** [Design] tab just under the menu bar at the top of the development environment. This should bring you back to the form designer view. Click on the **ibEnroller** form object to highlight it. In the **Properties** window (which should



now be showing properties for the **ibEnroller**), click on the button that resembles a lightning bolt. This is the event editor.



28. Double-click the EnrollComplete event property. This will automatically name and register this event to your source code and put you back into the editor. Add the following statements to this event (ibEnroller_EnrollComplete):

```
btnEnrollCancel.Visible = false;
btnEnroll.Visible = true;
string statusText = "Enroll complete. Result = " +
ibEnroller.EnrollResult;
if (ibEnroller.EnrollResult > 0)
{
statusText += " (Good)";
// Good result. Now available: ibEnroller.GetEnrollTemplate(),
ibEnroller.GetEnrollImage(1..3);
int enrollingFingerNumber = ibEnroller.EnrollingFingerNumber;
enrollTemplate[enrollingFingerNumber - 1] =
ibEnroller.GetEnrollTemplate();
```



```
btnCompare.Enabled = true; // Allow compare / verify
}
lblEnrollStatus.Text = statusText;
```

29. Again, click on the **frmQuickStart.cs** [Design] tab just under the menu bar to return to the design environment. Click on the **ibEnroller** form object to highlight it. In the **Properties** window, click on the events button (resembling a lightning bolt). Double-click the EnrollError event property. This will automatically name and register this event to your source code and put you back into the editor. Add the following statements to this event (**ibEnroller EnrollError**):

lblEnrollStatus.Text = "Problem while enrolling: " + ibEnroller.LastErrorText();

30. Return to the [Design] tab. Click on the ibEnroller form object to highlight it. In the Properties window, click on the events button. Double-click the PlaceFinger event property. Add the following statements to the event, ibEnroller_PlaceFinger:

lblEnrollStatus.Text = "Place Finger on the sensor";

31. Return to the [Design] tab. Click on the ibEnroller form object to highlight it. In the Properties window, click on the events button. Double-click the RaiseFinger event property. Add the following statements to the event, ibEnroller_RaiseFinger:

lblEnrollStatus.Text = "Raise Finger";

32. Return to the [Design] tab. Click on the ibVerifier form object (the single white box on the lower left) to highlight it. In the Properties window, click on the events button. Double-click the RaiseFinger event property. Add the following statements to the event, ibVerifier_RaiseFinger:



```
btnCompareCancel.Visible = false;
btnCompare.Visible = true;
string statusText = "Image complete";
ibVerifier.SecurityLevel9052vs9052 = tbSecurityLevel.Value;
// Now compare to enrolled image
byte[] compareTemplate = ibVerifier.Template();
string matchFingerName = "";
int compareResult = SearchForMatch(compareTemplate, ref
matchFingerName);
if (compareResult > 0)
{
statusText += " - Match found - " + matchFingerName + ".";
}
else
{
statusText += " - Not matched.";
1
lblEnrollStatus.Text = statusText;
```

33. Return to the [Design] tab. Click on the ibVerifier form object to highlight it. In the Properties window, click on the events button. Double-click the ScanError event property. Add the following statements to the event, ibVerifier_ScanError:

```
btnCompare.Visible = true;
btnCompareCancel.Visible = false;
lblEnrollStatus.Text = "Scan error : " +
ibVerifier.LastErrorMessage();
```

34. Return to the [**Design**] tab. Double-click the **Enroll** button to register the appropriate button-click event, add the following statements to this event:

```
if (ibVerifier.DeviceStatus == 0)
{
lblEnrollStatus.Text = "No USB Scanner found";
```



```
return;
}
btnEnrollCancel.Visible = true;
btnEnroll.Visible = false;
ibEnroller.MinimumNFIQ = tbNFIQLevel.Value;
ibEnroller.StartEnrollSequence();
Click again on the "frmQuickStart.cs [Design]" tab, double click the
"Match" button, and add the following statement:
if (ibVerifier.DeviceStatus == 0)
{
lblEnrollStatus.Text = "No USB Scanner found";
return;
}
btnCompareCancel.Visible = true;
btnCompare.Visible = false;
lblEnrollStatus.Text = "Place your finger on the sensor";
ibVerifier.StartScanning();
```

35. Return to the [**Design**] tab. Double-click the topmost **Cancel** button to register the appropriate button-click event, add the following statements to this event:

```
ibEnroller.StopEnrollSequence();
btnEnroll.Visible = true;
btnEnrollCancel.Visible = false;
lblEnrollStatus.Text = "Enroll sequence aborted";
```

36. Return to the [**Design**] tab. Double-click the lower **Cancel** button to register the appropriate button-click event, add the following statements to this event:



```
ibVerifier.StopScanning();
btnCompare.Visible = true;
btnCompareCancel.Visible = false;
lblEnrollStatus.Text = "Aborted";
```

37. Finally add the following subroutine just below the **ibVerifier_RaiseFinger** event you just inserted. This scans the enrolled fingerprints to find a match in the verifier fingerprint:

```
private int SearchForMatch(byte[] compareTemplate, ref string
matchFingerName)
{
int compareResult = 0;
for (int finger = 0; finger < NFingers; finger++)</pre>
{
if (enrollTemplate[finger] != null)
{
compareResult = ibVerifier.CompareFt9052vs9052(compareTemplate,
enrollTemplate[finger]);
if (compareResult > 0)
{
matchFingerName = FingerName[finger];
break;
                            // No need to search further
}
}
}
return compareResult;
}
```



38. You should now have a working project showing you how to enroll a fingerprint and then verify it. Press F5 to run the software, and see the fruits of your effort. Press the Enroll button to enter your fingerprint 3 times. Once you've successfully "enrolled", the Match button should become active. Press it to score your fingerprint against your enrollment.



Your final code should look something like the listing on the next page.



2 QuickStart Project Listing (C#)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
namespace QuickStart3_0
{
   public partial class frmQuickStart : Form
   {
       const int NFingers = 10;
       private byte[][] enrollTemplate = null;
       string[] FingerName = { "L5", "L4", "L3", "L2", "L1", "R1", "R2", "R3", "R4", "R5" };
       public frmQuickStart()
       {
           InitializeComponent();
       }
       private void frmQuickStart_Load(object sender, EventArgs e)
       {
           enrollTemplate = new byte[NFingers][];
                                                             // Placeholder for 10 templates
           // Move the cancel buttons underneath their appropriate counterparts.
           btnEnrollCancel.Location = btnEnroll.Location;
           btnCompareCancel.Location = btnCompare.Location;
       }
       11
       11
              Enrollment Events
       11
       private void btnEnroll_Click(object sender, EventArgs e)
       {
          if (ibVerifier.DeviceStatus == 0)
           {
              lblEnrollStatus.Text = "No USB Scanner found";
              return;
          }
```



```
btnEnrollCancel.Visible = true;
   btnEnroll.Visible = false;
   ibEnroller.MinimumNFIQ = tbNFIQLevel.Value;
   ibEnroller.StartEnrollSequence();
}
private void ibEnroller_EnrollComplete()
{
   btnEnrollCancel.Visible = false;
   btnEnroll.Visible = true;
   string statusText = "Enroll complete. Result = " + ibEnroller.EnrollResult;
   if (ibEnroller.EnrollResult > 0)
   {
       statusText += " (Good)";
       // Good result. Now available:
       // ibEnroller.GetEnrollTemplate(), ibEnroller.GetEnrollImage(1..3);
       int enrollingFingerNumber = ibEnroller.EnrollingFingerNumber;
       enrollTemplate[enrollingFingerNumber - 1] = ibEnroller.GetEnrollTemplate();
       btnCompare.Enabled = true;
                                     // Allow compare / verify
   }
   lblEnrollStatus.Text = statusText;
}
private void ibEnroller_EnrollError()
{
   lblEnrollStatus.Text = "Problem while enrolling: " + ibEnroller.LastErrorText();
}
private void ibEnroller_PlaceFinger()
{
   lblEnrollStatus.Text = "Place Finger on the sensor";
}
private void ibEnroller_RaiseFinger()
{
   lblEnrollStatus.Text = "Raise Finger";
}
11
```



```
11
       Match Events
11
private void btnCompare_Click(object sender, EventArgs e)
{
   if (ibVerifier.DeviceStatus == 0)
   {
       lblEnrollStatus.Text = "No USB Scanner found";
       return;
   }
   btnCompareCancel.Visible = true;
   btnCompare.Visible = false;
   lblEnrollStatus.Text = "Place your finger on the sensor";
   ibVerifier.StartScanning();
}
private void ibVerifier_ScanError()
{
   btnCompare.Visible = true;
   btnCompareCancel.Visible = false;
   lblEnrollStatus.Text = "Scan error : " + ibVerifier.LastErrorMessage();
}
private void ibVerifier_RaiseFinger()
{
   btnCompareCancel.Visible = false;
   btnCompare.Visible = true;
   string statusText = "Image complete";
   ibVerifier.SecurityLevel9052vs9052 = tbSecurityLevel.Value;
   // Now compare to enrolled image
   byte[] compareTemplate = ibVerifier.Template();
   string matchFingerName = "";
   int compareResult = SearchForMatch(compareTemplate, ref matchFingerName);
   if (compareResult > 0)
   {
       statusText += " - Match found - " + matchFingerName + ".";
   }
   else
   {
       statusText += " - Not matched.";
   }
   lblEnrollStatus.Text = statusText;
```



```
IBI SDK Manual for Windows
```

}

```
}
   private int SearchForMatch(byte[] compareTemplate, ref string matchFingerName)
   {
       int compareResult = 0;
       for (int finger = 0; finger < NFingers; finger++)</pre>
       {
           if (enrollTemplate[finger] != null)
           {
              compareResult = ibVerifier.CompareFt9052vs9052(compareTemplate,
              enrollTemplate[finger]);
              if (compareResult > 0)
              {
                  matchFingerName = FingerName[finger];
                  break;
                                                 // No need to search further
               }
           }
       }
       return compareResult;
   }
   11
   11
           Cancellation Events
   11
   private void btnEnrollCancel_Click(object sender, EventArgs e)
   {
       ibEnroller.StopEnrollSequence();
       btnEnroll.Visible = true;
       btnEnrollCancel.Visible = false;
       lblEnrollStatus.Text = "Enroll sequence aborted";
   }
   private void btnCompareCancel_Click(object sender, EventArgs e)
   {
       ibVerifier.StopScanning();
       btnCompare.Visible = true;
       btnCompareCancel.Visible = false;
       lblEnrollStatus.Text = "Aborted";
   }
}
```





© 2012 Integrated Biometrics LLC

Integrated Biometrics LLC.

121 Broadcast Drive Spartanburg SC 29303 phone: (864) 990-3711 toll free: (888) 840-8034 fax: (864) 631-1719 http://www.integratedbiometrics.com

