Document Version 1.0

IVI Driver Getting Started Guide

For EEC8500 series

Overview

This application note will describe the installing instructions and several programming examples for IVI Instrument Driver of EEC8500 series. To understand more about the IVI drivers, please refer to the website of IVI Foundation. For more detail of the EEC8500 IVI driver, please check the help document, EEC8500.chm, located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500*.

1. IVI Driver Setup

Instructions on downloading and Installing IVI Instrument Drivers from website. Download and install Shared Components from IVI Foundation Website.

2. Getting Started with C#

A tutorial using IVI driver establishes communication with the instrument by C# programming.

3. Getting Started with C++

A tutorial using IVI driver establishes communication with the instrument by C++ programming.

4. Getting Started with Python

A tutorial using IVI driver establishes communication with the instrument by Python programming.

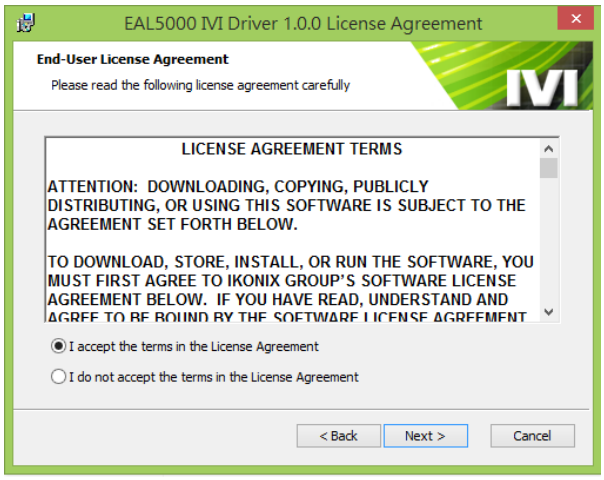
5. Getting Started with LabVIEW

A tutorial using IVI driver establishes communication with the instrument by LabVIEW programming.

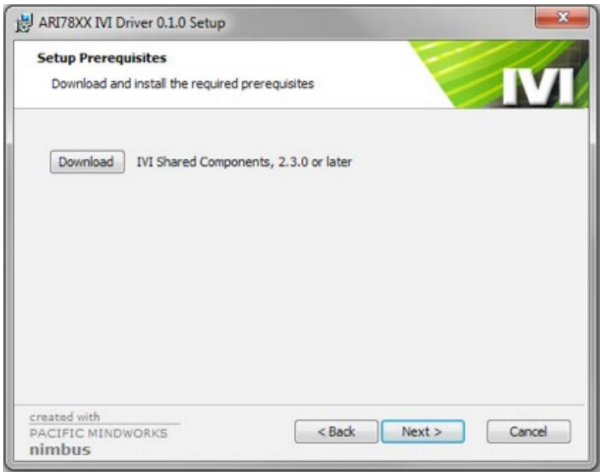
28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

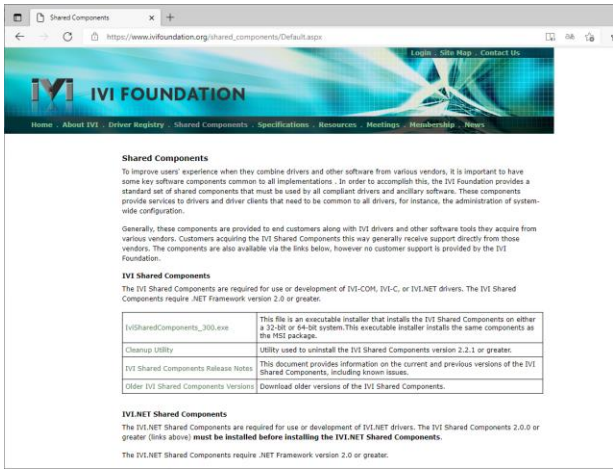
1. IVI Driver Setup

After downloading the IVI Driver, run the self-extracting setup file and you will see the installation wizard to start setup. Please follow the below instruction to complete the installation.



The setup will detect if IVI Shared Components are installed. If prompted with the following screen, click on Download, The IVI Foundation Website will open.

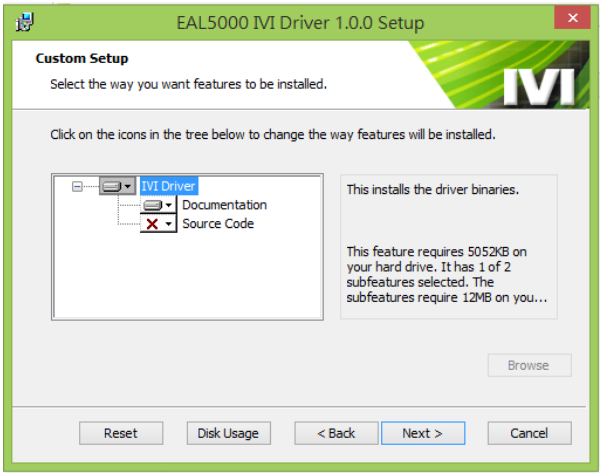


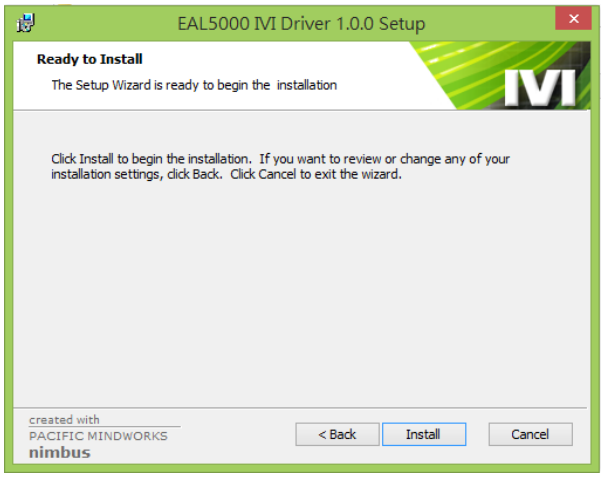


Please download the latest IVI Shared Components either 32-bit or 64-bit version. After downloading, install the shared components and continue the installation.

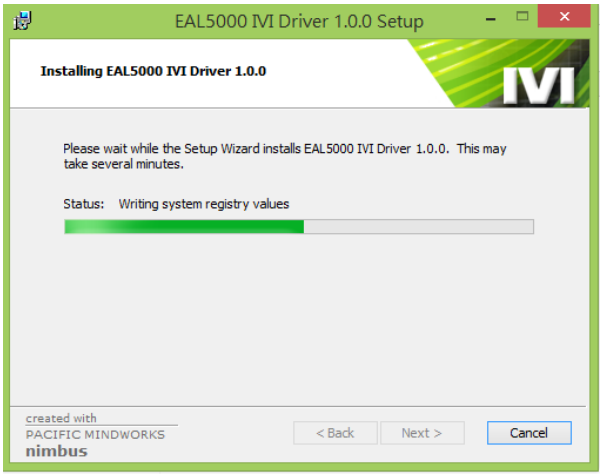
28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

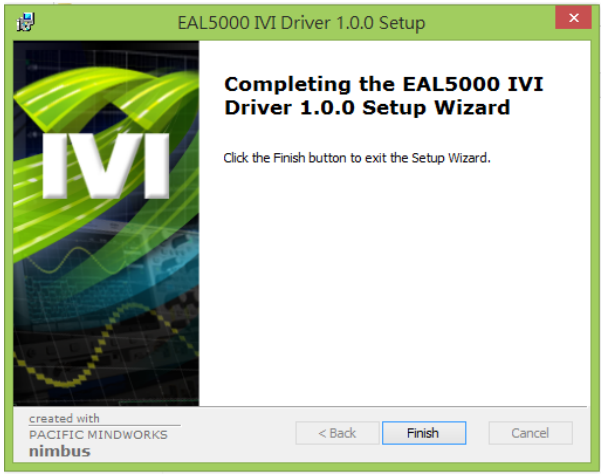
After the IVI Shared Components are installed, please follow the steps to complete installation.





There are options for installing the source code of the IVI Driver, if it is necessary.





The IVI driver would be installed under the path of "<Program Files>\IVI Foundation\IVI". For the files of the *.dll file would be located in the "Bin" folder. And the necessary help documents will be in the folder of "..\Drivers\EEC8500".

28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

2. Getting Started with C#

Introduction

This chapter describes the procedures of using the IVI-COM driver of Ikonix Group by C# programming language. In this exercise, the programmer could import the driver and complete a short program controlling the device step-by step.

The C# could use IVI-C driver, either. However, we suggest that an IVI-COM interop would be easier for you to develop the program.

Requirements

- EEC8500 IVI Driver
- IVI Shared Components, *https://www.ivifoundation.org/shared_components/Default.aspx*
- Microsoft Visual Studio 2015
- An EEC8500 series power supply, including 8505, 8512, 8520, 8530, 8540, 8560

Download the Drivers

Please go to the website of the IKONIX to download the latest version of IVI drivers or contact the vendors. Follow the steps and instructions in Chapter 1 to complete the installation.

References

On the website of IVI Foundation, there are documentations you might be interested while implementing controlling the devices. You could find the resources of developing with an IVI driver, *https://www.ivifoundation.org/resources/default.aspx*. The IVI Shared Components could be download from *https://www.ivifoundation.org/shared_components/Default.aspx*. There are several documents on the website for understanding the IVI.

In the installed directory, there are several documents for your reference understanding the EEC8500 IVI Driver. A help file, ***EEC8500.chm***, would be located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500*. In this help file, you could find all of the provided functions and their hierarchy.

There are four types of sample code for your reference which are located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500\Examples*, including C#, C++, Python and LabVIEW as well.

Development

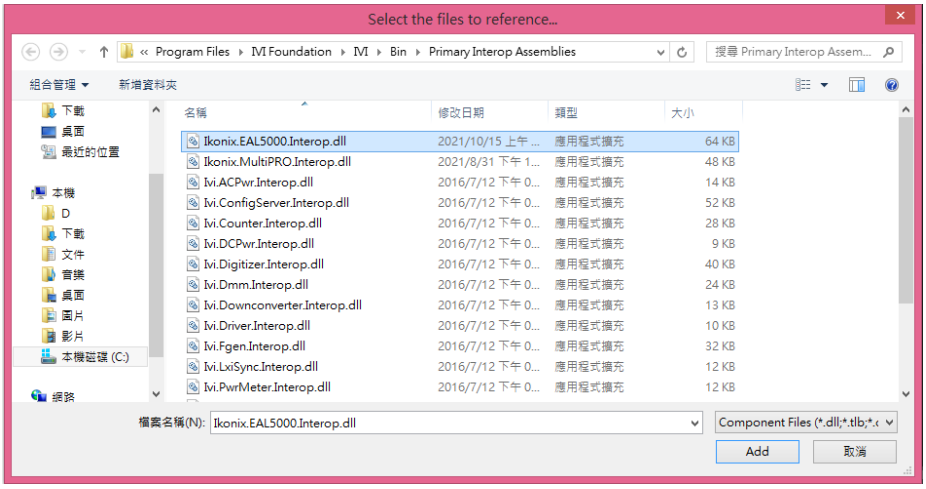
1 Create a C# project

1.1 Open Visual Studio IDE and create a new C# console project.

2 Import Libraries

2.1 Right-click on the reference and select ***Add Reference*** in the solution explorer

2.2 Click on the Browse button and go to the path of "*<Program Files>\IVI Foundation\IVI\Bin\Primary Interop Assemblies*" and choose ***Ikonix.EEC8500.Interop.dll***, ***Ivi.Driver.Interop.dll*** and ***Ivi.ACPwr.Interop.dll.***



2.3 Declare to use the name spaces for the interop assemblies that are specified to reference in the previous section.

```
using Ikonix.EEC8500.Interop;
```

3 Start programming

3.1 Create an object of the driver and use the initialize method to build up the connection.

```
var driver = new EEC8500();
driver.Initialize("ASRL5::INSTR", true, false,"QueryInstrStatus=true,
DriverSetup=BaudRate=115200");
```

For more detail for the parameters of the ***Initialize()*** method, please refer to the help document, ***EEC8500.chm***, which is located at "*<Program Files>\IVI Foundation\IVI\Drivers\EEC8500*".

The first parameter ***ResourceName*** is a string type and indicates the interfaces type and address of the connection. The resource name,`"ASRL5::INSTR"`, represents a serial port with address 5. For example, a GPIB connection could be `"GPIB0::8::INSTR"`. For TCP/IP connection, it will be in the format of `"TCPIP0::192.168.0.1::10001::SOCKET"`. The 10001 is the TCP/IP connection port of EEC8500.

If you use serial port to connect to device, please follow the above example, `"DriverSetup=BaudRate=115200"`, to setup the baud rate. There are other parameters for the option of the ***Initialize()*** method, please refer to the ***EEC8500.chm*** for more detail. For

example, "QueryInstrStatus=true" makes the session automatically query the error status for each command was sent.

3.2 Create file and setup test

```
// Create/Edit File in Manual mode
driver.Execution.SetOperationMode(EEC8500OperationModeEnum.EEC8500OperationModeManual);
Console.WriteLine(driver.Execution.GetOperationMode().ToString());
try
{
    driver.ManualMode.File.Add("IVITEST");
}
catch (System.Runtime.InteropServices.COMException e)
{
    // It will return an error if the file was already existed.
    Console.WriteLine("Cannot Create a  Manual File with Name, 'IVITEST'. ");
}
finally
{
    driver.ManualMode.File.Edit("IVITEST");
}

// Confugure the parameters
//
driver.ManualMode.Couple = EEC8500CoupleEnum.EEC8500CoupleACDC;
driver.ManualMode.Waveform = EEC8500WaveformEnum.EEC8500WaveformSINE;
driver.ManualMode.Range = EEC8500RangeEnum.EEC8500RangeAuto;
driver.ManualMode.VoltageAC = 100;
driver.ManualMode.VoltageDC = 10;
driver.ManualMode.Frequency = 62;
driver.ManualMode.RampUp = 0.5;
driver.ManualMode.CurrentHighLimit = 10;
driver.ManualMode.CurrentLimitDelay = 10;
driver.ManualMode.PowerHighLimit = 12;
driver.ManualMode.StartAngle = 180;
```

For the EEC8500, all of the test parameters would be within a file. Therefore, you need to create a file first and then setup the parameters. Also, there are many difference between Standard and Advanced levels of EEC8500. In this example, the configurations of ***CoupleACDC***, ***Waveform***, ***CurrentLowLimit*** and ***PowerLowLimit*** are not available in the Standard level of EEC8500.

3.3 Load file and start a test

```
// Output and Measure
//
driver.ManualMode.File.Load("IVITEST");
driver.OutputPhases.Item["PhaseA"].Enabled = true;
```

Before running output, you have to select a file to load. And then invoke ***`driver.OutputPhases.Item["PhaseA"].Enabled`*** method to start a test.

3.4 Measure during test

```
for (int i = 0; i < 20; i++)
{
    Console.WriteLine("Output State: {0}", driver.Measure.State());
    Console.WriteLine("Measure All: {0}", driver.Measure.All());
}
driver.OutputPhases.Item["PhaseA"].Enabled = false;
```

This while loop would run with the condition of state is testing. Using the methods of Measure subsystem could let you read the immediate readings.

3.5 Close the session

```
driver.OutputPhases.Item["PhaseA"].Enabled = false;
driver.Close();

Console.WriteLine("Done - Press Enter to Exit");
Console.ReadLine();
```

Close() would close the I/O session to the instrument.

4 Completed example

The completed sample code could be find at the path of "*<Program Files>\IVI Foundation\IVI\Drivers\EEC8500\Examples*".

28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

3. Getting Started with C++

Introduction

This chapter describes the procedures of using the IVI-COM driver of Ikonix Group by C++ programming language. In this exercise, the programmer could import the driver and complete a short program controlling the device step-by step.

Requirements

- EEC8500 IVI Driver
- IVI Shared Components, *https://www.ivifoundation.org/shared_components/Default.aspx*
- Microsoft Visual Studio 2015 or other IDEs
- An EEC8500 series power supply, including 8505, 8512, 8520, 8530, 8540, 8560

Download the Drivers

Please go to the website of the IKONIX to download the latest version of IVI drivers or contact the vendors. Follow the steps and instructions in Chapter 1 to complete the installation.

References

On the website of IVI Foundation, there are documentations you might be interested while implementing controlling the devices. You could find the resources of developing with an IVI driver, *https://www.ivifoundation.org/resources/default.aspx*. The IVI Shared Components could be download from *https://www.ivifoundation.org/shared_components/Default.aspx*. There are several documents on the website for understanding the IVI.

In the installed directory, there are several documents for your reference understanding the EEC8500 IVI Driver. A help file, ***EEC8500.chm***, would be located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500*. In this help file, you could find all of the provided functions and their hierarchy.

There are three types of sample code for your reference which are located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500\Examples*, including C#, C++ and Python as well.

Development

1 Create a C++ project

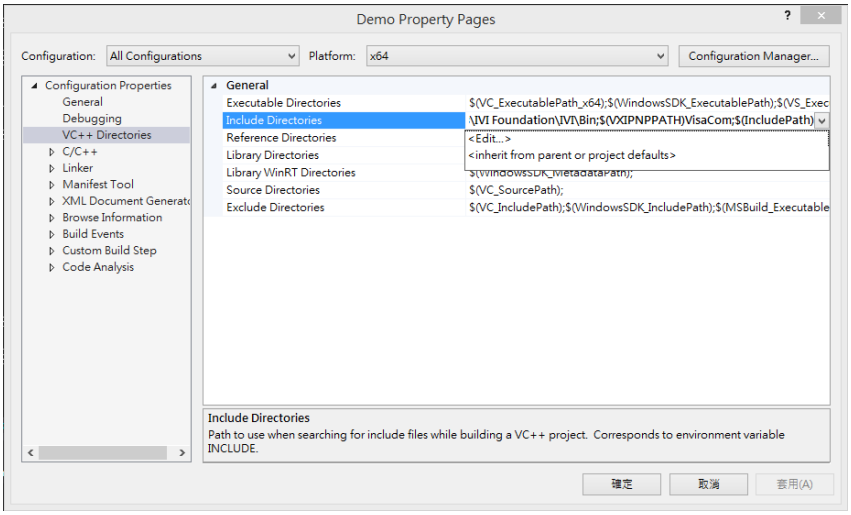
1.1 Open Visual Studio IDE and create a new C++ console project.

2 Include Directories

2.1 Right-click on the project and select ***properties***.

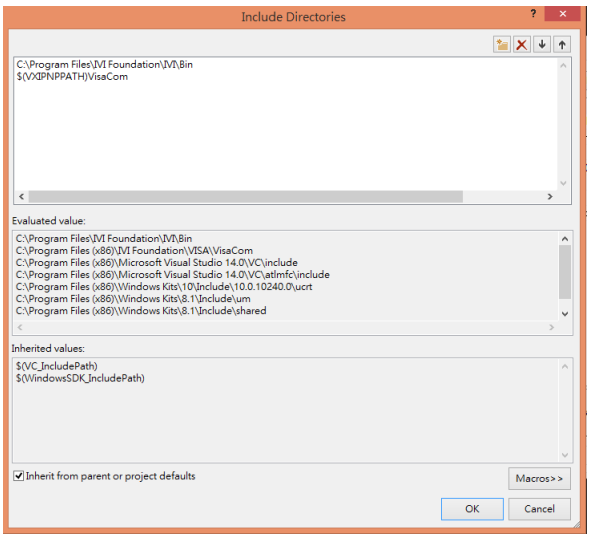
2.2 Expand the ***Configuration Properties*** and select ***VC++ Directories*** on the left menu.

2.3 Click on the drop-down column of the ***Include Directories*** and select ***<Edit..>*** to open the edit window.



2.4 Select the ***New Line*** button to add an include directories. There will be two necessary paths need to be added.

- ***<Program Files>\IVI Foundation\IVI\Bin***
- ***$(VXIPNPPATH)VisaCom***



2.5 Click ***OK*** to complete including the directories.

2.6 Use the `#import` operator to import the necessary DLLs

```
#include "stdafx.h"
```

28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

```
#include <iostream>
#import <IviDriverTypeLib.dll> no_namespace
#import <GlobMgr.dll> no_namespace
#import <EEC8500_64.dll> no_namespace
```

3 Start programming

3.1 Create an instance of the driver by pointer and use the initialize method to build up the connection.

```
HRESULT hr = ::CoInitialize(NULL);
IEEC8500Ptr driver(__uuidof(EEC8500));

// IIviDriverIdentity properties - Initialize required
//
driver -> Initialize("ASRL6::INSTR", true, false, "QueryInstrStatus=true,
DriverSetup=BaudRate=115200");
```

For more detail for the parameters of the ***Initialize***() method, please refer to the help document, ***EEC8500.chm*** located at "*<Program Files>\IVI Foundation\IVI\Drivers\EEC8500*". The first parameter ***ResourceName*** is a string type and indicates the interfaces type and address of the connection. The resource name,`"ASRL6::INSTR"`, represents a serial port with address 5. For example, a GPIB connection could be `"GPIB0::8::INSTR".` For TCP/IP connection, it will be in the format of `"TCPIP0::192.168.0.1::10001::SOCKET"`. The 10001 is the TCP/IP connection port of EEC8500.

If you use serial port to connect to device, please follow the above example, `"DriverSetup=BaudRate=115200"`, to setup the baud rate. There are other parameters for the option of the ***Initialize()*** method, please refer to the ***EEC8500.chm*** for more detail. For example, `"QueryInstrStatus=true"` makes the session automatically query the error status for each command was sent.

3.2 Create file and setup test

```
// Create/Edit File in Manual mode
//
driver -> Execution ->
SetOperationMode(EEC8500OperationModeEnum::EEC8500OperationModeManual);
try
{
        driver -> ManualMode -> File -> Add("IVITEST");
}
catch (...)
{
        driver -> ManualMode -> File -> Edit("IVITEST");
}

// Confugure the parameters
//
driver -> ManualMode -> Couple = EEC8500CoupleEnum::EEC8500CoupleACDC;
driver -> ManualMode -> Waveform = EEC8500WaveformEnum::EEC8500WaveformSINE;
driver -> ManualMode -> Range = EEC8500RangeEnum::EEC8500RangeAuto;
driver -> ManualMode -> VoltageAC = 100;
driver -> ManualMode -> VoltageDC = 10;
driver -> ManualMode -> Frequency = 62;
driver -> ManualMode -> RampUp = 0.5;
driver -> ManualMode -> CurrentHighLimit = 10;
driver -> ManualMode -> CurrentLimitDelay = 1;
driver -> ManualMode -> PowerHighLimit = 500;
driver -> ManualMode -> StartAngle = 180;
```

For the EEC8500, all of the test parameters would be within a file. Therefore, you need to create a file first and then select a file to be edited. Also, there are many difference between Standard and Advanced levels of EEC8500. In this example, the configurations of ***CoupleACDC***, ***Waveform*** are not available in the Standard level of EEC8500.

3.3 Load file and start a test

```
// Output and Measure
//
driver -> ManualMode -> File -> Load("IVITEST");
driver -> OutputPhases -> Item["PhaseA"] -> Enabled = true;
```

Before running output, you have to select a file to load. And then set the property of ***driver->OutputPhases->Item["PhaseA"]->Enabled*** to start a test by a Boolean value.

3.4 Measure during test

```
std::cout << "\nLoad File, IVITEST, and start Test\n" << std::endl;
for (int i = 0; i < 20; i++)
{
        std::cout << driver -> Measure -> State() << std::endl;
        std::cout << driver -> Measure -> All() << std::endl;
}
driver -> OutputPhases -> Item["PhaseA"] -> Enabled = false;
```

This while loop would run with polling the states and meters. Using the methods of Measure subsystem could let you read the immediate readings.

3.5 Close the session

```
//Close connection
driver -> Close();
std::cout << "Done - Press Enter to Exit" << std::endl;
std::cin.get();
```

Close() would close the I/O session to the instrument.

4 Completed example

The completed sample code could be found at the path of "*<Program Files>\IVI Foundation\IVI\Drivers\EEC8500\Examples*".

28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

4. Getting Started with Python

Introduction

This chapter describes the procedures of using the IVI-COM driver of Ikonix Group by Python programming language. In this exercise, the programmer could import the driver and complete a short program controlling the device step-by step.

Requirements

- EEC8500 IVI Driver
- IVI Shared Components, *https://www.ivifoundation.org/shared_components/Default.aspx*
- Python IDE
- Cometypes Library (pip install cometypes)
- An EEC8500 series power supply, including 8505, 8512, 8520, 8530, 8540, 8560

Download the Drivers

Please go to the website of the IKONIX to download the latest version of IVI drivers or contact the vendors. Follow the steps and instructions in Chapter 1 to complete the installation.

References

On the website of IVI Foundation, there are documentations you might be interested in while implementing controlling the devices. You could find the resources of developing with an IVI driver, *https://www.ivifoundation.org/resources/default.aspx*. The IVI Shared Components could be download from *https://www.ivifoundation.org/shared_components/Default.aspx*. There are several documents on the website for understanding the IVI.

In the installed directory, there are several documents for your reference understanding the EEC8500 IVI Driver. A help file, ***EEC8500.chm***, would be located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500*. In this help file, you could find all of the provided functions and their hierarchy.

There are three types of sample code for your reference which are located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500\Examples*, including C#, C++ and Python as well.

Development

1 Install the ***Comtypes*** library

```
pip install cometypes
```

In order to call an external com DLL in Python, you will need ***comtypes*** library installed.

2 Create a Python file

2.1 Open any IDE of Python and create a new Python file.

3 Import Libraries

3.1 Import the cometypes library and EEC8500_64.dll

```
import time
import comtypes
import comtypes.client as cc

cc.GetModule('EEC8500_64.dll')
from comtypes.gen import EEC8500Lib
```

4 Start programming

4.1 Create an object of the driver and use the initialize method to build up the connection.

```
driver = cc.CreateObject('EEC8500.EEC8500', interface=EEC8500Lib.IEEC8500)

#   Initialize Driver and make connection
driver.Initialize('ASRL6::INSTR', True, False, 'QueryInstrStatus=true,
DriverSetup=BaudRate=115200')
```

For more detail for the parameters of the ***Initialize***() method, please refer to the help document, ***EEC8500.chm*** located at "*<Program Files>\IVI Foundation\IVI\Drivers\EEC8500*". The first parameter ResourceName is a string type and indicates the interfaces type and address of the connection. The resource name,`"ASRL5::INSTR"`, represents a serial port with address 5. For example, a GPIB connection could be `"GPIB0::8::INSTR".` For TCP/IP connection, it will be in the format of `"TCPIP0::192.168.0.1::10001::SOCKET"`. The 10001 is the TCP/IP connection port of EEC8500.

If you use serial port to connect to device, please follow the above example, `"DriverSetup=BaudRate=115200"`, to setup the baud rate. There are other parameters for the option of the ***Initialize()*** method, please refer to the ***EEC8500.chm*** for more detail. For example, `"QueryInstrStatus=true"` makes the session automatically query the error status for each command was sent.

4.2 Create file and setup test

```
#   Create File
try:
    driver.ManualMode.File.Add('IVITEST')
#   Edit File
except:
    print('File cannot be added, perhaps it is already existed.')
    driver.ManualMode.File.Edit('IVITEST')

#   Configure Parameters
```

28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

```
driver.ManualMode.Couple = EEC8500Lib.EEC8500CoupleACDC;
driver.ManualMode.Waveform = EEC8500Lib.EEC8500WaveformSINE;
driver.ManualMode.Range = EEC8500Lib.EEC8500RangeAuto;
driver.ManualMode.VoltageAC = 100;
driver.ManualMode.VoltageDC = 10;
driver.ManualMode.Frequency = 62;
driver.ManualMode.RampUp = 0.5;
driver.ManualMode.CurrentHighLimit = 10;
driver.ManualMode.CurrentLimitDelay = 10;
driver.ManualMode.CurrentLowLimit = 10;
driver.ManualMode.PowerHighLimit = 12;
driver.ManualMode.PowerLowLimit = 13;
driver.ManualMode.StartAngle = 180;
```

For the EEC8500, all of the test parameters would be within a file. Therefore, you need to create a file first and then select a file to be edited. Also, there are many difference between Standard and Advanced levels of EEC8500. In this example, the configurations of ***CoupleACDC***, ***Waveform***, ***CurrentLowLimit*** and ***PowerLowLimit*** are not available in the Standard level of EEC8500.

4.3 Load file and start a test

```
#   Load file and start test
driver.ManualMode.File.Load("IVITEST");
print("Load File then Start Output")
driver.OutputPhases.Item["PhaseA"].Enabled = True;
```

Before running output, you have to select a file to load. And then set ***driver.OutputPhases.Item["PhaseA"].Enabled*** property to start a test with a Boolean value.

4.4 Measure during test

```
for i in range(0,20):
    print("Output State: ", driver.Measure.State());
    print("Measure All: ", driver.Measure.All());
driver.OutputPhases.Item["PhaseA"].Enabled = False;
```

This for loop would run with polling the state and meters. Using the methods of Measure subsystem could let you read the immediate readings.

4.5 Close the session

```
# Close connection
driver.Close();
print("Done - Press Enter to Exit");
```

Close() would close the I/O session to the instrument.

5 Completed example

The completed sample code could be find at the path of "*<Program Files>\IVI Foundation\IVI\Drivers\EEC8500\Examples*".

5. Getting Started with LabVIEW

Introduction

This chapter describes the procedures of using the IVI-COM driver of Ikonix Group by LabVIEW programming language. In this exercise, the programmer could learn how to import the driver and complete a short program controlling the device step-by step.

Even though the programmers could control the device by IVI Driver. For the LabVIEW programmer, we suggest that using LabVIEW plu&play driver would be easier for your programming and debugging. The LabVIEW driver from Ikonix Group are all made up with commands directly, so you could clearly check how the commands were sent to instruments.

Requirements

- EEC8500 IVI Driver
- IVI Shared Components, *https://www.ivifoundation.org/shared_components/Default.aspx*
- National Instruments LabVIEW (This example was written in LabVIEW 2014)
- An EEC8500 series power supply, including 8505, 8512, 8520, 8530, 8540, 8560

Download the Drivers

Please go to the website of the IKONIX to download the latest version of IVI drivers or contact the vendors. Follow the steps and instructions in Chapter 1 to complete the installation.

References

On the website of IVI Foundation, there are documentations you might be interested while implementing controlling the devices. You could find the resources of developing with an IVI driver, *https://www.ivifoundation.org/resources/default.aspx*. The IVI Shared Components could be download from *https://www.ivifoundation.org/shared_components/Default.aspx*. There are several documents on the website for understanding the IVI.

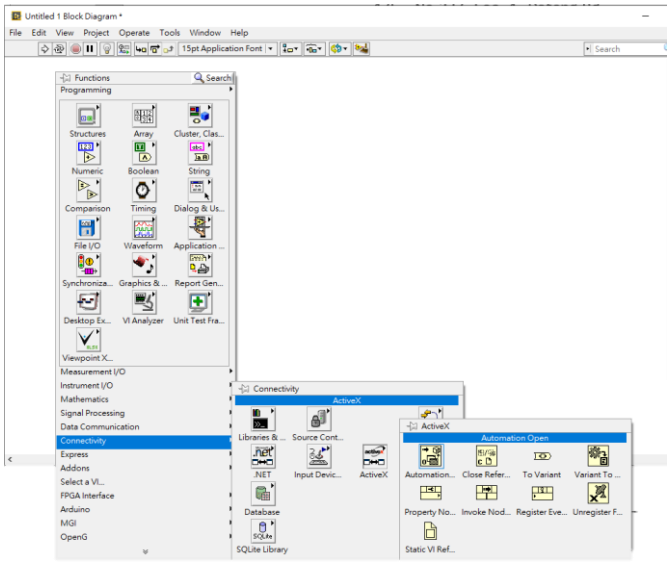
In the installed directory, there are several documents for your reference understanding the EEC8500 IVI Driver. A help file, ***EEC8500.chm***, would be located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500*. In this help file, you could find all of the provided functions and their hierarchy.

There are three types of sample code for your reference which are located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500\Examples*, including C#, C++ and Python as well.

Development

1 Open a new vi.

2 Import the DLL component.

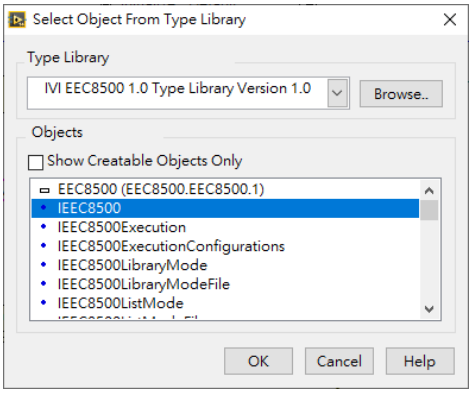


Open the ***Function Palette*** by right-clicking on the block diagram. Then select ***Connectivity*** -> ***ActiveX***. Select or drop the ***Automation Open*** function on the block diagram.

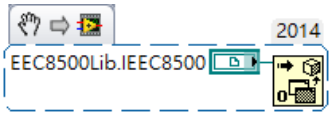
3 Right-clicking on the ***Automation Open*** and select ***Select ActiveX Class*** -> ***Browse*** will open a window for choosing the DLL.

4 Select the Browse button and select the file ***EEC8500.dll*** located at *<Program Files> (x86)\IVI Foundation\IVI\Bin.* The ***IVI EEC8500 Type Library*** would be added into the ***Type Libraries*** drop down menu.

5 Select ***IEEC8500*** and then click ***OK*** to complete creating an object of EEC8500 driver instance.

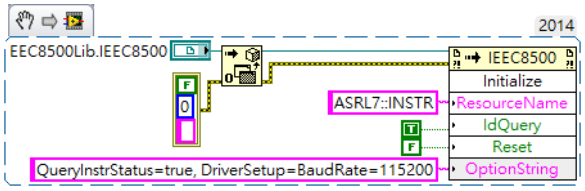


The Labview will automatically generate an ***Automation refnum*** of ***EEC8500Lib.IEEC8500*** control and connect to the Automation Open function.



28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

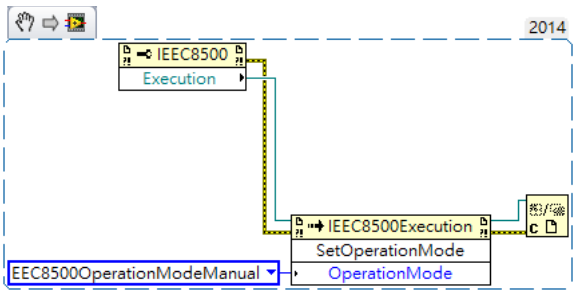
6 Create an ***Invoke Node*** function and connect the reference to the output of ***Automation Refnum*** and then click on the ***Method*** and select ***Initialize*** to initialize the connection with device.



For more detail for the parameters of the ***Initialize***() method, please refer to the help document, ***EEC8500.chm*** located at "*<Program Files>\IVI Foundation\IVI\Drivers\EEC8500*". The first parameter ***ResourceName*** is a string type and indicates the interfaces type and address of the connection. The resource name,`"ASRL7::INSTR"`, represents a serial port with address 4. For example, a GPIB connection could be `"GPIB0::8::INSTR".` For TCP/IP connection, it will be in the format of `"TCPIP0::192.168.0.1::10001::SOCKET"`. The 10001 is the TCP/IP connection port of EEC8500.

If you use serial port to connect to device, please follow the above example, `"DriverSetup=BaudRate=115200"`, to setup the baud rate. There are other parameters for the option of the Initialize() method, please refer to the ***EEC8500.chm*** for more detail. For example, `"QueryInstrStatus=true"` makes the session automatically query the error status for each command was sent.

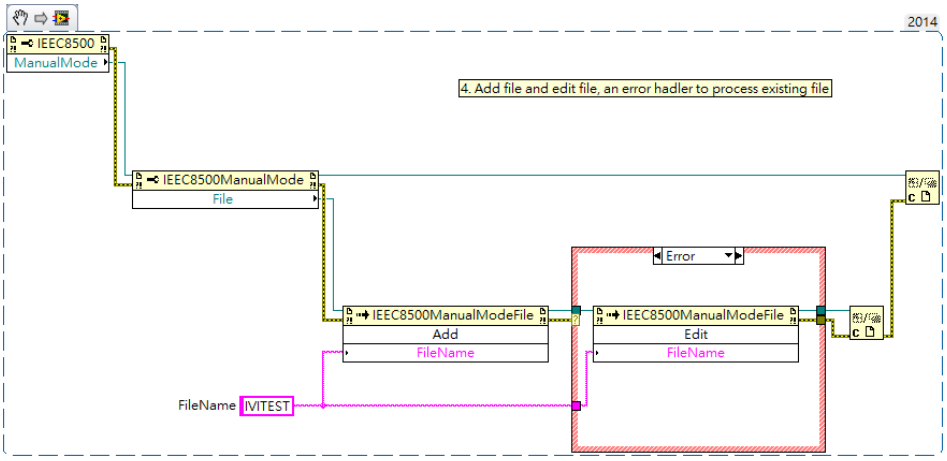
6.1 Switch operation mode



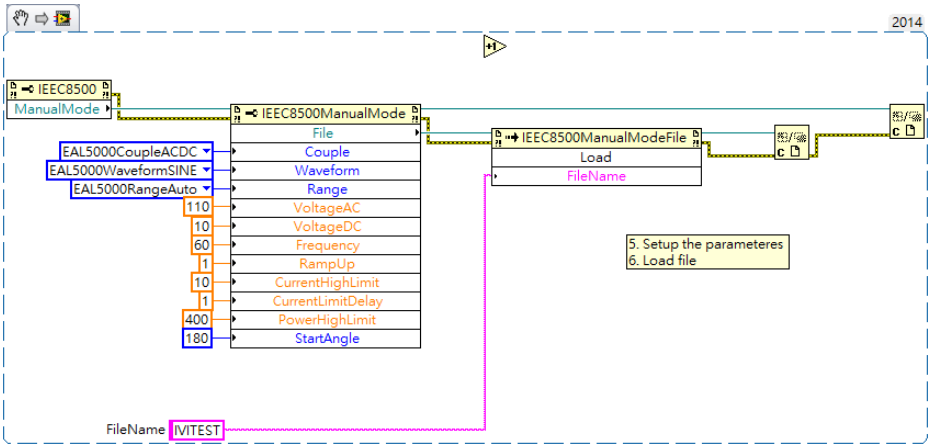
Before we create a Manual Mode file on the EEC8500 power supply, we have to switch the operation mode to Manual Mode. The SetOperationMode() method is under the Execution interface. Please be noted that only the Advanced level of 8500 supports the Step, Pulse and Library modes.

28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

6.2 Create file and setup test



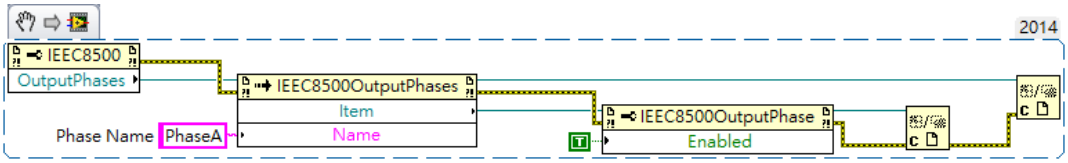
Use the ***Property Node*** to get reference of the sub-system of ***IEEC8500*** class. For example, ***File*** is one of the sub class of ***IEEC8500.ManualMode***, and then we could use the ***Add*** method under ***File*** to create a file in the device.



For the EEC8500, all of the test parameters would be within a file. Therefore, you need to create a file first and then select a file to be edited. Also, there are many differences between Standard and Advanced levels of EEC8500. In this example, the configurations of ***CoupleACDC***, ***Waveform*** are not available in the Standard level of EEC8500.

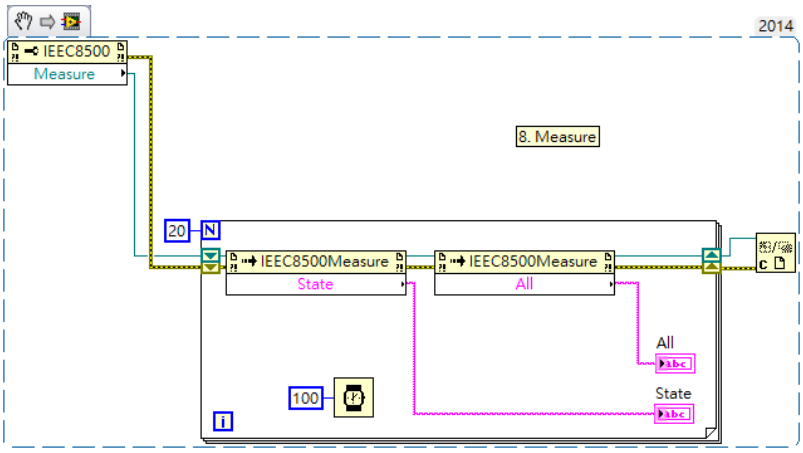
Please be noted that the flow of error data could make sure that the procedure ran sequentially.

6.3 Load file and start a test



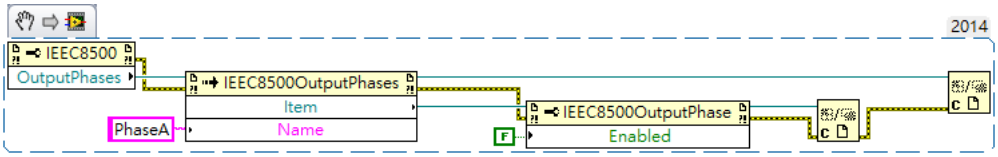
Before running output, you have to select a file to load. And then invoke ***driver.OutputPhases.Item["PhaseA"].Enabled*** property to start a test with a Boolean value. The ***OutputPhases*** class is one property of the ***IEEC8500*** class. The function of ***Item["PhaseA"]*** is the method of ***OutputPhases***. This method inherits the IviACPwr class, it provides the features for multiple phases, however, the EEC8500 provides only single phase capability.

6.4 Measure during test

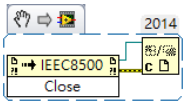


This for loop would run with polling the state and meters. Using the methods of Measure subsystem could let you read the immediate readings.
Measure is one property of ***IEEC8500***, and ***All*** is a method of ***Measure*** class. It will return the measurements in a raw string.

6.5 Stop and close the session



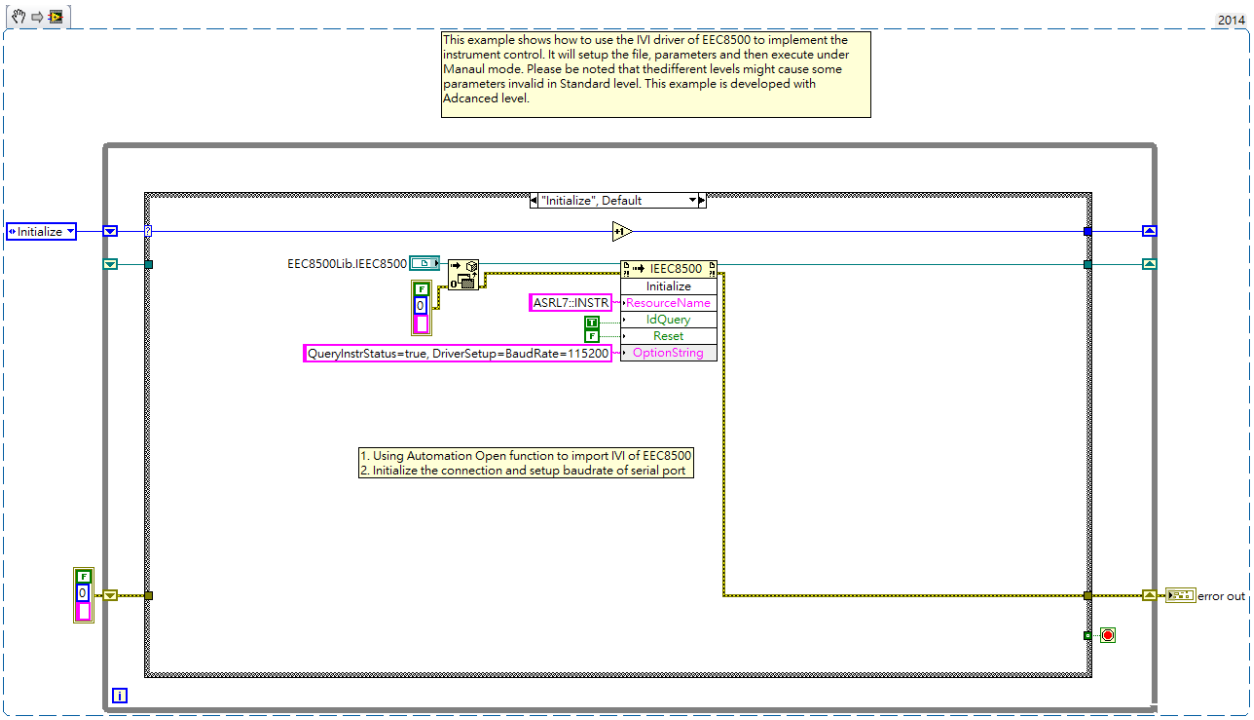
The above procedure which is similar with the output procedure shows how to stop output, just to set the Enable property to False.



Close method in ***IEEC8500*** class would close the I/O session to the instrument. Also, all of the references should be closed using the ***Close Reference*** function.

28105 N. Keith Drive
Lake Forest, IL 60045 USA
Toll Free: 1-800-858-8378 US/Canada
Phone: 1-847-367-4077 | Fax: 1-847-367-4080 | www.eecsources.com

7 Completed example



The completed example for your reference which are located at the path of *<Program Files>\IVI Foundation\IVI\Drivers\EEC8500\Examples*, including C#, C++ and Python as well. However, we suggest that using LabVIEW plug & play driver would be easier for LabVIEW developers. If you need a LabVIEW driver, please download it from the website of IKONIX or contact the vendor.