# 1. PCIe/PXIe-5114-H7 Specifications

**Multi-functional Data Acquisition Boards**

Please download JYTEK <JYPEDIA>, you can quickly inquire the product prices, the key features and available accessories.

## Overview

JYTEK JY-5114 board includes the PCIe-5114-H7 and PXIe-5114-H7 models, which belong to the JY-5114 series. The PCIe-5114-H7 is a board based on the PCI Express interface，providing up to 96 channels of analog inputs of up to 1M samples per second, 2 channels of analog outputs, 24 channels of digital IO or 2 32-bit counters/timers; the PXIe-5114-H7 is a board based on the PXI Express interface，providing up to 96 channels of analog inputs of up to 1M samples per second, 2 channels of analog outputs, 24 lines of Digital IO or 2 32-bit counters/timers.

## 1.1. Main Features

**AI Features:**

- High accuracy: 470 ppm
- High Density : Supports up to 96 channels
- Anti-aliasing filter : Eliminate out-of-band interference
- 16 bits ADC
- Sampling rates: 1MS/s
- Analog/Digital/Software Trigger
- Onboard memory 256MB for analog inputs

**AO Features:**

- 2 simultaneous 16-bit analog output channels
- Updating rate :2 MS/s
- Onboard memory 64MB for analog outputs

**CIO Features:**

- 2 channels 32-bit timer/counter

**DIO Features**

- 24 lines of Digital IO
- DIO supports hardware timing up to 10MHz

## 1.2. Analog Input

| Model | PCIe/PXIe-5114-H7 |
|---|---|
| Number of channels | 96 SE/ 48 Diff |
| ADC resolution (Bits) | 16 |
| Single channel maximum sample rate | 1 MS/s |
| Multichannel maximum sample rate with same range(aggregate) | 1 MS/s/n (n=1,2,3…96) |
| Clock | 100 MHz |
| Input range(V) | ±10 / ±5 / ±2.5 |
| Anti-aliasing filter | 15 kHz / 39 kHz / 375 kHz (Software selectable) |
| Input mode | RSE / Differential |
| Input impedance | >1MΩ\|\|330pF |
| Input coupling | DC |
| CMRR | 85 dB |
| Crosstalk | -80 dB |
| DNL | No Missing Code |
| INL | 70 ppm of Range Typical |
| Trigger sources | Any PFI, PXI_TRIG, AI <0..95> |
| Trigger modes | Start trigger, Reference trigger, Re-trigger for start trigger and reference trigger modes |
| Analog trigger voltage range | ±10 V Software Programmable |
| Overvoltage Protection | ±15 V |
| Onboard memory | 256MB for analog inputs |
| Data transfers | Polling, scatter-gather DMA |

Table 1 Analog Input Specifications

## 1.3. Analog Output

| Model | | PCIe/PXIe-5114-H7 |
|---|---|---|
| Number of channels | | 2 (SE) |
| DAC resolution | | 16 bits |
| Maximum update rate (simultaneous) | 1 channel | 2 MS/s |
| | 2 channels | 1 MS/s |
| Clock | | 100 MHz |
| Clock accuracy | | Jitter <20 ps |
| Output range(V) | | ±10 |
| Output mode | | RSE |
| Output impedance | | 0.2 Ω |
| Output coupling | | DC |
| Output current drive | | ±10 mA |
| Onboard memory | | 64MB for analog inputs |
| Trigger source | | Any PFI, PXI_TRIG |
| Trigger type | | Start Trigger |

Table 2 Analog Output Specifications

## 1.4. Counter Input/Output

| Model | PCIe/PXIe-5114-H7 |
|---|---|
| Number of channels | 2 channels<br>(sharing the IO pin with PFI) |
| Resolution | 32 bits |
| Counter Input | Edge Counting, Period Measure, SemiPeriod Measure, Freguency Measure, Pulse Measure, Two Edge Separation Measure, Quadrature Encoder,Two Pulse Encoder |
| Counter Output | Single, finite and continuous pulse |
| Onboard memory | 16MB for Counter Input/Output |
| Clock | 100 MHz/5 MHz/100kHz |
| Input | Gate，Source，Aux |
| Output | OUT |

Table 3 Counter Input Operations Specifications

## 1.5. PFI Specifications

| Model | PCIe/PXIe-5114-H7 |
|---|---|
| Number of channels | 24 |
| External digital trigger interface | Trigger voltage : 3.3 V TTL |
| | Trigger edge: Rising/Falling |
| Initial state | Input |
| Functionality | Static digital input, static digital output, timing input, timing output |
| Timing output sources | Many AI, AO, counter, DI, DO timing signals |
| Debounce filter settings | enable, disable;selectable per input |

Table 4 PFI Specifications

## 1.6. Digital IO Specifications

| Model | PCIe/PXIe-5114-H7 |
|---|---|
| Number of channels | 24 Lines |
| Ground reference | D GND |
| Directional control | Independent control of each port |
| Max update rate | 10 MHz |
| Onboard memory | 32MB for Digital Input/Output |
| Initial state | Input |
| Digital Input | Logic Low: VIL Min : 0 / Max : 0.8 V |
| | Logic High: VIH Min : 2 V / Max : 5.3 V |
| Digital Output | Logic Low : 0 V, IOL Max: 12 mA |
| | Logic High : 2.6 V～3.3 V, IOH: -12 mA～0 mA |
| Overvoltage Protection | -3～20V |

Table 5 Digital IO Specifications

## 1.7. Basic DC AI Accuracy

| JY-5114 AI Basic Accuracy = ±(% Reading+% Range) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Nominal Range (V) | 24 Hour Tcal ±1°C | | | 90 Days Tcal±5°C | | | Temperature Coefficients(/°C) | | | 24 Hr Full Scale Accuracy | 90 Days Full Scale Accuracy | Full Scale Accuracy(%) |
| 2.5 | 0.010 | + | 0.036 | 0.024 | + | 0.037 | 0.0004 | + | 0.0002 | 1.15 mV | 1.53 mV | 0.061 |
| 5 | 0.009 | + | 0.026 | 0.024 | + | 0.027 | 0.0004 | + | 0.0002 | 1.75 mV | 2.55 mV | 0.051 |
| 10 | 0.009 | + | 0.023 | 0.023 | + | 0.024 | 0.0004 | + | 0.0002 | 3.20 mV | 4.70 mV | 0.047 |

Table 6 Basic DC AI Accuracy

## 1.8. AI Bandwidth

| Analog Input Bandwidth | | |
|---|---|---|
| Nominal Range Full Scale (V) | Filter | -3dB Bandwidth (kHz) |
| All range | Narrow | 15 |
| 10V | Medium | 39 |
| 5V/2.5V | Medium | 33 |
| All range | Wide | 375 |

Table 7 AI Bandwidth Specifications

## 1.9. System Noise

| Nominal Range (V) | Filter | System Noise (uVrms) |
|---|---|---|
| 10 | wide | 415 |
| 5 | wide | 298 |
| 2.5 | wide | 228 |
| 10 | medium | 194 |
| 5 | medium | 109 |
| 2.5 | medium | 73 |
| 10 | narrow | 172 |
| 5 | narrow | 92 |
| 2.5 | narrow | 57 |
| *Filter mode :wide/medium/narrow - refer to Table 7 | | |

Table 8 System Noise Specifications Physical and Environment

## 1.10. Basic AO Accuracy

| JY-5114 AO Basic Accuracy = ±(% Reading+% Range) | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Nominal Range (V) | 24 Hour Tcal ±1°C | | | 90 Days Tcal±5°C | | | Temperature Coefficients(/°C) | | 24 Hr Full Scale Accuracy | 90 Days Full Scale Accuracy | Full Scale Accuracy(%) |
| 10 | 0.010 | + | 0.008 | 0.029 | + | 0.013 | 0.0018 | + 0.0013 | 1.80 mV | 4.20 mV | 0.04 |

Table 9 Basic AO Accuracy

## 1.11. Physical and Environment

Operating Environment

| | |
|---|---|
| Ambient temperature range | 0 °C to 50 °C |
| Relative humidity range | 20% to 80%, noncondensing |

Storage Environment

| | |
|---|---|
| Ambient temperature range | -20 ˚C to 80 ˚C |
| Relative humidity range | 10% to 90%, noncondensing |

Table 10 Physical and Environment

## 1.12. Power Consumption (typical)

| Voltage | Current | Power | Total Power |
|---|---|---|---|
| 12V | 1.2A | 14.4W | 14.6W |
| 3.3V | 0.06A | 0.2W | |

Table 11 JY-5114 Power Consumption

## 1.13. Front Panel and Pin Definition

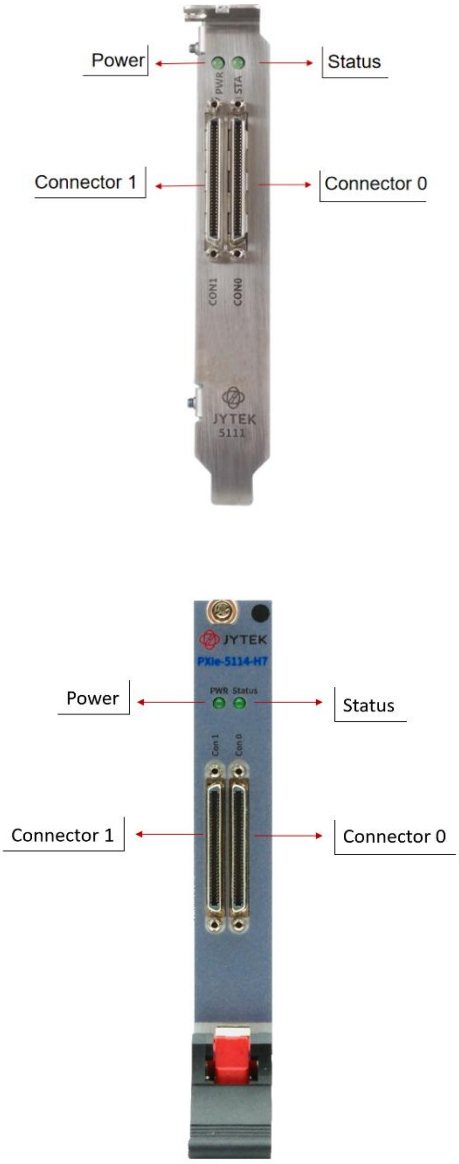


Figure 1 PCIe/PXIe 5114-H7 Front Panel

| Connter 1 | | | |
|---|---|---|---|
| PIN Number | Definition | PIN Number | Definition |
| 1 | AI 32 (AI 16+) | 35 | AI 40 (AI 16-) |
| 2 | AI 33 (AI 17+) | 36 | AI 41 (AI 17-) |
| 3 | AI 34 (AI 18+) | 37 | AI 42 (AI 18-) |
| 4 | AI 35 (AI 19+) | 38 | AI 43 (AI 19-) |
| 5 | AI 36 (AI 20+) | 39 | AI 44 (AI 20-) |
| 6 | AI 37 (AI 21+) | 40 | AI 45 (AI 21-) |
| 7 | AI 38 (AI 22+) | 41 | AI 46 (AI 22-) |
| 8 | AI 39 (AI 23+) | 42 | AI 47 (AI 23-) |
| 9 | AI GND | 43 | AI GND |
| 10 | AI 48 (AI 24+) | 44 | AI 56 (AI 24-) |
| 11 | AI 49 (AI 25+) | 45 | AI 57 (AI 25-) |
| 12 | AI 50 (AI 26+) | 46 | AI 58 (AI 26-) |
| 13 | AI 51 (AI 27+) | 47 | AI 59 (AI 27-) |
| 14 | AI 52 (AI 28+) | 48 | AI 60 (AI 28-) |
| 15 | AI 53 (AI 29+) | 49 | AI 61 (AI 29-) |
| 16 | AI 54 (AI 30+) | 50 | AI 62 (AI 30-) |
| 17 | AI 55 (AI 31+) | 51 | AI 63 (AI 31-) |
| 18 | AI 64 (AI 32+) | 52 | AI 72 (AI 32-) |
| 19 | AI 65 (AI 33+) | 53 | AI 73 (AI 33-) |
| 20 | AI 66 (AI 34+) | 54 | AI 74 (AI 34-) |
| 21 | AI 67 (AI 35+) | 55 | AI 75 (AI 35-) |
| 22 | AI 68 (AI 36+) | 56 | AI 76 (AI 36-) |
| 23 | AI 69 (AI 37+) | 57 | AI 77 (AI 37-) |
| 24 | AI 70 (AI 38+) | 58 | AI 78 (AI 38-) |
| 25 | AI 71 (AI 39+) | 59 | AI 79 (AI 39-) |
| 26 | AI GND | 60 | AI GND |
| 27 | AI 80 (AI 40+) | 61 | AI 88 (AI 40-) |
| 28 | AI 81 (AI 41+) | 62 | AI 89 (AI 41-) |
| 29 | AI 82 (AI 42+) | 63 | AI 90 (AI 42-) |
| 30 | AI 83 (AI 43+) | 64 | AI 91 (AI 43-) |
| 31 | AI 84 (AI 44+) | 65 | AI 92 (AI 44-) |
| 32 | AI 85 (AI 45+) | 66 | AI 93 (AI 45-) |
| 33 | AI 86 (AI 46+) | 67 | AI 94 (AI 46-) |
| 34 | AI 87 (AI 47+) | 68 | AI 95 (AI 47-) |

| Connter 0 | | | |
|---|---|---|---|
| PIN Number | Definition | PIN Number | Definition |
| 1 | PFI 0/DIO 0 | 35 | PFI 1/DIO 1 |
| 2 | PFI 2/DIO 2 | 36 | PFI 3/DIO 3 |
| 3 | PFI 4/DIO 4 | 37 | PFI 5/DIO 5 |
| 4 | PFI 6/DIO 6 | 38 | PFI 7/DIO 7 |
| 5 | PFI 8/DIO 8 | 39 | PFI 9/DIO 9 |
| 6 | PFI 10/DIO 10 | 40 | PFI 11/DIO 11 |
| 7 | PFI 12/DIO 12 | 41 | PFI 13/DIO 13 |
| 8 | PFI 14/DIO 14 | 42 | PFI 15/DIO 15 |
| 9 | DGND | 43 | DGND |
| 10 | PFI 16/DIO 16 | 44 | PFI 17/DIO 17 |
| 11 | PFI 18/DIO 18 | 45 | PFI 19/DIO 19 |
| 12 | PFI 20/DIO 20 | 46 | PFI 21/DIO 21 |
| 13 | PFI 22/DIO 22 | 47 | PFI 23/DIO 23 |
| 14 | DGND | 48 | DGND |
| 15 | AO GND | 49 | AO 0 |
| 16 | AO GND | 50 | AO 1 |
| 17 | AI GND | 51 | AI GND |
| 18 | AI 0 (AI 0+) | 52 | AI 8 (AI 0-) |
| 19 | AI 1 (AI 1+) | 53 | AI 9 (AI 1-) |
| 20 | AI 2 (AI 2+) | 54 | AI 10 (AI 2-) |
| 21 | AI 3 (AI 3+) | 55 | AI 11 (AI 3-) |
| 22 | AI 4 (AI 4+) | 56 | AI 12 (AI 4-) |
| 23 | AI 5 (AI 5+) | 57 | AI 13 (AI 5-) |
| 24 | AI 6 (AI 6+) | 58 | AI 14 (AI 6-) |
| 25 | AI 7 (AI 7+) | 59 | AI 15 (AI 7-) |
| 26 | AI GND | 60 | AI GND |
| 27 | AI 16 (AI 8+) | 61 | AI 24 (AI 8-) |
| 28 | AI 17 (AI 9+) | 62 | AI 25 (AI 9-) |
| 29 | AI 18 (AI 10+) | 63 | AI 26 (AI 10-) |
| 30 | AI 19 (AI 11+) | 64 | AI 27 (AI 11-) |
| 31 | AI 20 (AI 12+) | 65 | AI 28 (AI 12-) |
| 32 | AI 21 (AI 13+) | 66 | AI 29 (AI 13-) |
| 33 | AI 22 (AI 14+) | 67 | AI 30 (AI 14-) |
| 34 | AI 23 (AI 15+) | 68 | AI 31 (AI 15-) |

Table 12 JY-5114 Pin Connector

**Note:**

- At JY-5114 AI channel synchronization, each bank can only select one;
  In RSE mode, the following is the distribution of the banks:

| RSE mode | AI channel |
|---|---|
| Bank1 | AI 0 - AI 7 |
| Bank2 | AI 8 - AI 15 |
| Bank3 | AI 16 - AI 23 |
| Bank4 | AI 24 - AI 31 |
| Bank5 | AI 32 - AI 39 |
| Bank6 | AI 40 - AI 47 |
| Bank7 | AI 48 - AI 55 |
| Bank8 | AI 56 - AI 63 |
| Bank9 | AI 64 - AI 71 |
| Bank10 | AI 72 - AI 79 |
| Bank11 | AI 80 - AI 87 |
| Bank12 | AI 88 - AI 95 |

Table 13 JY-5114 Banks-RSE

In DIFF mode, the following is the distribution of the banks:

| DIFF mode | AI channel |
|---|---|
| Bank1 | AI 0 - AI 7 |
| Bank2 | AI 8 - AI 15 |
| Bank3 | AI 16 - AI 23 |
| Bank4 | AI 24 - AI 31 |
| Bank5 | AI 32 - AI 39 |
| Bank6 | AI 40 - AI 47 |

Table 14 JY-5114 Banks-DIFF

Please pay attention to the wiring method.

| **Connector 0** | | **Connector 0** | |
|---|---|---|---|
| **Pin** | **Identification** | **Pin** | **Identification** |
| **1** | **Ctr0_Source/A** | **3** | **Ctr1_Source/A** |
| **35** | **Ctr0_Gate/Z** | **37** | **Ctr1_Gate/Z** |
| **2** | **Ctr0_Aux/B** | **4** | **Ctr1_Aux/B** |
| **36** | **Ctr0_Out** | **38** | **Ctr1_Out** |

Table 15 JY-5114 Pin Definition-Counter

| | |
|---|---|
| AI_GND | Analog Input Reference Ground |
| AI<0..95> | Analog Input channel |
| AO_GND | Analog Output Reference Ground |
| AO<0..1> | Analog Output Channel |
| D_GND | Digital Signal Reference Ground |
| DIO<0..23> | Digital I/O Channel |
| PFI<0..23> | Programmable Function Interface |

Table 16 Cable Specification

## 1.14. Accessories

- DIN-68 (PN: JY1717615-01)

  SCSI 68-pin Terminal board

- ACL-1016868-1 (PN: JY1016868-01)

  1M 68pin VHDCI68M-SCSI68M cable

- ACL-1016868-2 (PN: JY1016868-02)

  2M 68pin VHDCI68M-SCSI68M cable

## 2. Special Operation Note

In the operation of the JY-5114, it is crucial to note that the device **does not support the mixed use of single-ended and differential measurements.** All input channels must be configured consistently as either single-ended or differential for the entirety of the operation.

- **Single-ended mode:** In this mode, each signal is measured with respect to a common ground.
- **Differential mode:** In this mode, the measurement is taken between two input signals, ensuring noise rejection and higher signal integrity.

If your application requires switching between these two modes, it is recommended to reconfigure the channels uniformly before starting a new measurement session.

# Table of Contents

# 3. Software

## 3.1. Supported Operating System

Windows 7/10, x64/x86, Linux. See additional information.

## 3.2. Programming Languages

Microsoft C#. See additional software information for other languages.

# 4. Order Information

- PCIe-5114-H7 (PN: JY7772358-01)

  96-ch AI (16-Bit, 1 MS/s), 2-ch AO (16-Bit, 2 MS/s), 24 DIO PCIe Multifunction I/O Card

- PXIe-5114-H7 (PN: JY2633977-01)

  96-ch AI (16-Bit, 1 MS/s), 2-ch AO (16-Bit, 2 MS/s), 24 DIO PXIe Multifunction I/O Card

# 5. JYPEDIA

JYPEDIA is an excel file. It contains JYTEK product information, pricing, inventory information, drivers, software, technical support, knowledge base etc. You can register and download a JYPEDIA excel file from our web www.jytek.com. JYTEK highly recommends you use this file to obtain information from JYTEK.

# 6. Additional Hardware Information

## 6.1. Basic DC AI Accuracy

| JY-5114 AI Basic Accuracy = ±(% Reading+% Range) | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Nominal Range (V) | 24 Hour Tcal ±1°C | | | 90 Days Tcal±5°C | | | Temperature Coefficients(/°C) | | | 24 Hr Full Scale Accuracy | 90 Days Full Scale Accuracy | Full Scale Accuracy(%) |
| 2.5 | 0.010 | + | 0.036 | 0.024 | + | 0.037 | 0.0004 | + | 0.0002 | 1.15 mV | 1.53 mV | 0.061 |
| 5 | 0.009 | + | 0.026 | 0.024 | + | 0.027 | 0.0004 | + | 0.0002 | 1.75 mV | 2.55 mV | 0.051 |
| 10 | 0.009 | + | 0.023 | 0.023 | + | 0.024 | 0.0004 | + | 0.0002 | 3.20 mV | 4.70 mV | 0.047 |

Table 17 JY-5114 AI Accuracy

## 6.2. Basic AO Accuracy

| JY-5114 AO Basic Accuracy = ±(% Reading+% Range) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Nominal Range (V) | 24 Hour Tcal ±1°C | | | 90 Days Tcal±5°C | | | Temperature Coefficients(/°C) | | | 24 Hr Full Scale Accuracy | 90 Days Full Scale Accuracy | Full Scale Accuracy(%) |
| 10 | 0.010 | + | 0.008 | 0.029 | + | 0.013 | 0.0018 | + | 0.0013 | 1.80 mV | 4.20 mV | 0.04 |

Table 18 JY-5114 AO Accuracy

# 7. Additional Software Information

## 7.1. System Requirements

The PCIe/PXIe-5114-H7 can be used in a Windows or a Linux operating system.

Microsoft Windows: Windows 7 32/64 bit, Windows 10 32/64 bit.

Linux Kernel Versions: There are many Linux versions. It is not possible JYTEK can support and test our devices under all different Linux versions. JYTEK supports the following Linux versions only.

| Linux Version |
|---|
| **Ubuntu LTS** |
| 16.04： 4.4.0-21-generic(desktop/server) |
| 16.04.6：4.15.0-45-generic(desktop) 4.4.0-142-generic(server) |
| 18.04： 4.15.0-20-generic(desktop) 4.15.0-91-generic(server) |
| 18.04.4：5.3.0-28-generic (desktop) 4.15.0-91-generic(server) |
| **Localized Chinese Version** |
| 中标麒麟桌面操作系统软件（兆芯版）V7.0（Build61）: 3.10.0-862.9.1.nd7.zx.18.x86_64 |
| 中标麒麟高级服务器操作系统软件V7.0U6: 3.10.0-957.el7.x86_64 |

Table 19 Supported Linux Versions

## 7.2. System Software

When using the PCIe/PXIe-5114-H7 in the Window environment, you need to install the following software from Microsoft website:

Microsoft Visual Studio Version 2015 or above,

.NET Framework version is 4.0 or above.

.NET Framework is coming with Windows 10. For Windows 7, please check .NET Framework version and upgrade to 4.0 or later version.

Given the resources limitation, JYTEK only tested PCIe/PXIe-5114-H7 be with .NET Framework 4.0 with Microsoft Visual Studio 2015. JYTEK relies on Microsoft to maintain the compatibility for the newer versions.

## 7.3. C# Programming Language

All JYTEK default programming language is Microsoft C#. This is Microsoft recommended programming language in Microsoft Visual Studio and is particularly suitable for the test and measurement applications. C# is also a cross platform programming language.

## 7.4. PCIe/PXIe-5114-H7 Hardware Driver

After installing the required application development environment as described above, you need to install the PCIe/PXIe-5114-H7 hardware driver.

JYTEK hardware driver has two parts: the shared common driver kernel software (FirmDrive) and the specific hardware driver.

Common Driver Kernel Software (FirmDrive): FirmDrive is the JYTEK's kernel software for all hardware products of JYTEK instruments. You need to install the FirmDrive software before using any other JYTEK hardware products. FirmDrive only needs to be installed once. After that, you can install the specific hardware driver.

Specific Hardware Driver: Each JYTEK hardware has a C# specific hardware driver. This driver provides rich and easy-to-use C# interfaces for users to operate various PCIe/PXIe-5114-H7 function. JYTEK has standardized the ways which JYTEK and other vendor's DAQ boards are used by providing a consistent user interface, using the methods, properties and enumerations in the object-oriented programming environment. Once you get yourself familiar with how one JYTEK DAQ card works, you should be able to know how to use all other DAQ hardware by using the same methods.

**Note that this driver does not support cross-process, and if you are using more than one function, it is best to operate in one process.**

## 7.5. Install the SeeSharpTools from JYTEK

To efficiently and effectively use PCIe/PXIe-5114-H7 board, you need to install a set of free C# utilities, SeeSharpTools from JYTEK. The SeeSharpTools offers rich user interface functions you will find convenient in developing your applications. They are also needed to run the examples come with PCIe/PXIe-5114-H7 hardware. Please register and download the latest SeeSharpTools from our website, www.jytek.com.

## 7.6. Running C# Programs in Linux

Most C# written programs in Windows can be run by MonoDevelop development system in a Linux environment. You would develop your C# applications in Windows using Microsoft Visual Studio. Once it is done, run this application in the MonoDevelop environment. This is JYTEK recommended way to run your C# programs in a Linux environment.

If you want to use your own Linux development system other than MonoDevelop, you can do it by using our Linux driver. However, JYTEK does not have the capability to support the Linux applications. JYTEK completely relies upon Microsoft to maintain the cross-platform compatibility between Windows and Linux using MonoDevelop.

# 8. Operating JY5114

This chapter provides the operation guides for PCIe/PXIe-5114-H7, including AI, AO, DI, DO, Timer and programmable I/O interface, etc.

JYTEK provides extensive examples, on-line help and documentation to assist you to operate the PCIe/PXIe-5114-H7 board. JYTEK strongly recommends you go through these examples before writing your own application. In many cases, an example can also be a good starting point for a user application.

## 8.1. Quick Start

After you have installed the driver software and the SeeSharpTools, you are ready to use Microsoft Visual Studio C# to operate the PCIe/PXIe-5114-H7 products.

If you are already familiar with Microsoft Visual Studio C#, the quickest way to use PCIe/PXIe-5114-H7 board is to go through our extensive examples. We provide source code of our examples. In many cases, you can modify the source code and start to write your applications.

We also provide **Learn by Example** in the following sections. These examples will help you navigate and learn how to use this PCIe/PXIe-5114-H7.

## 8.2. Data Acquisition Methods

PCIe/PXIe-5114-H7 uses a scanning method to acquire analog data, meaning there is only one ADC chip on the device and all input channels share this ADC. In the scan acquisition mode, you need to configure AI channels and set up some parameters through PCIe/PXIe-5114-H7 driver software. The most important parameters are *Data Acquisition mode*, *Sample Rate*, *SamplesToAcquire*, *Channel Count*, *ChannelRange* and *Analog Input Terminal Type*.

*AI Acquisition mode* **(***AIMode***)**: PCIe/PXIe-5114-H7 provides 4 acquisition modes, **Continuous**, **Finite**, **Single Point**, **Record,** which will be described in details in Section 8.2.1-8.2.4.

*SampleRate:* How fast data are acquired per second per channel. For example, if the sample rate is 1000Hz, you acquire two channels of data, you will have 2000 points/second.

*SamplesToAcquire*: This parameter behaves differently in the different AI acquisition modes. In the continuous acquisition mode, *SamplesToAcuire* is the buffer size used in the AI acquisition task, please see Section8.2.1; in the finite acquisition mode, it is the total number of samples to capture, please see Section8.2.2.

*Channel Count:* how many channels you want to collect data. You can set up the channels in different orders, for instance 2,3,1,0. The acquired data will be arranged in the way you specify as shown in Figure 2. In this particular case, *Channel Count* is 4.

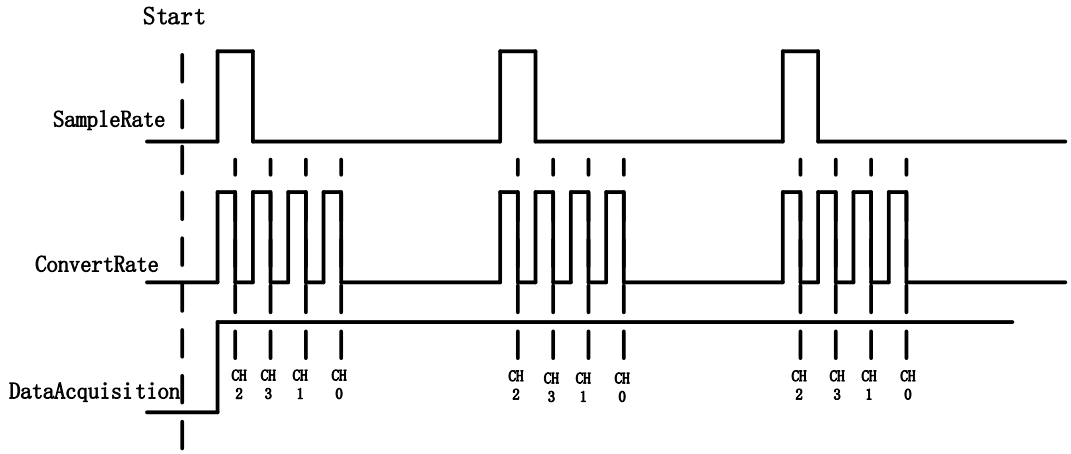


Figure 2 Sample Rate and Internal AD Conversion

*ConvertRate* denotes the working rate of ADC. In default: *ConvertRate* = *SampleRate* * *ChannelCount* . User can redefine the *ConvertRate* in our software. If user want to redefine *ConvertRate, The following conditions must be met:*

*Multichannel maximum sample rate (aggregate) >=ConvertRate >= SampleRate * ChannelCount.*

User can get Multichannel maximum sample rate (aggregate) from section Appendix.

**Learn by Example 8.2**

- Connect the two signal source's positive outputs to PCIe-5114-H7 AI Ch0 (AI0+, Pin#18) and AI Ch1 (AI1+, Pin#19), two negative terminals to the ground (AI_GND, Pin#17) as shown in Figure 3 and 错误!未找到引用源。 (AI0+, AI_GND) and (AI1+, AI_GND) consist of two channels of RSE inputs and they share the same GND.

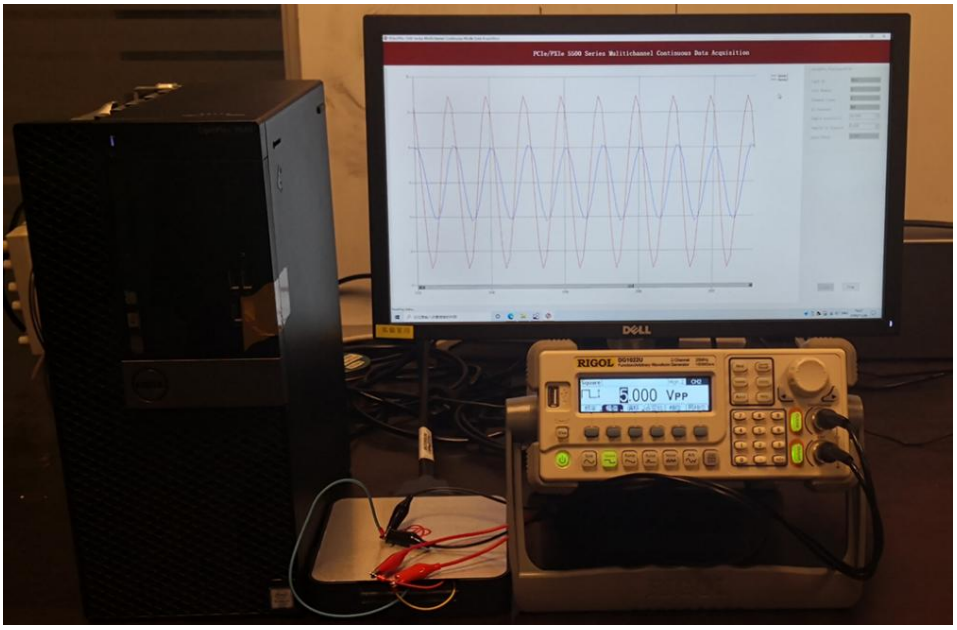


Figure 3 PCIe-5114-H7 experiment

- Set a sinewave signal (f=4Hz, Vpp=5V) and a squarewave signal (f=4Hz, Vpp=5V).
- Open **Analog Input-->Winform AI Continuous MultiChannel**, set the following numbers as shown. This sample program will continuously acquire data from multiple channels.

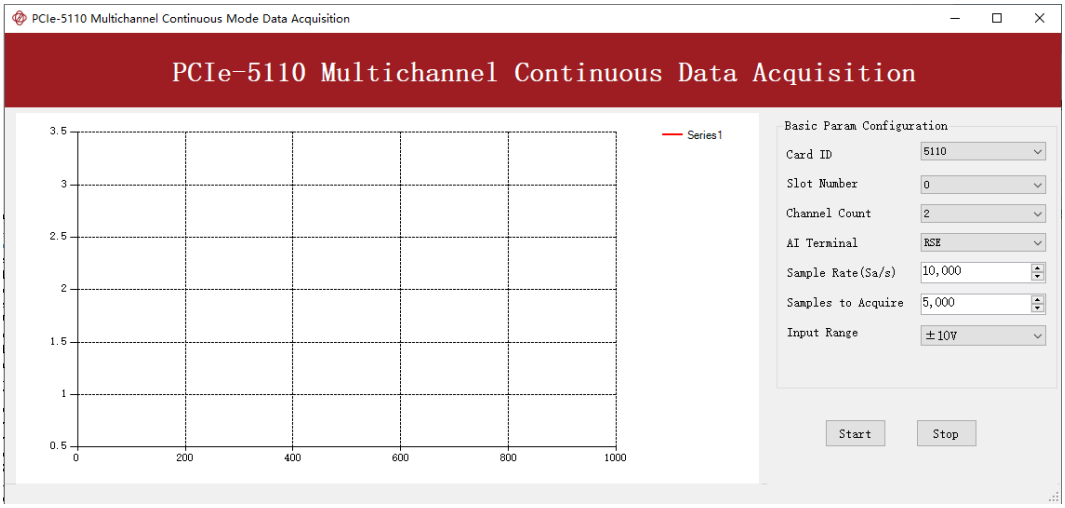


Figure 4 Continuous MultiChannel Paraments

- SampleRate is set by **Sample Rate**
- **Samples to Acquire** is the samples to be acquired for each channel in one block. The continuous mode will acquire blocks after blocks until **Stop** button is pressed.

- When start is clicked, it generates a software trigger, which starts the acquisition. The result is shown below.

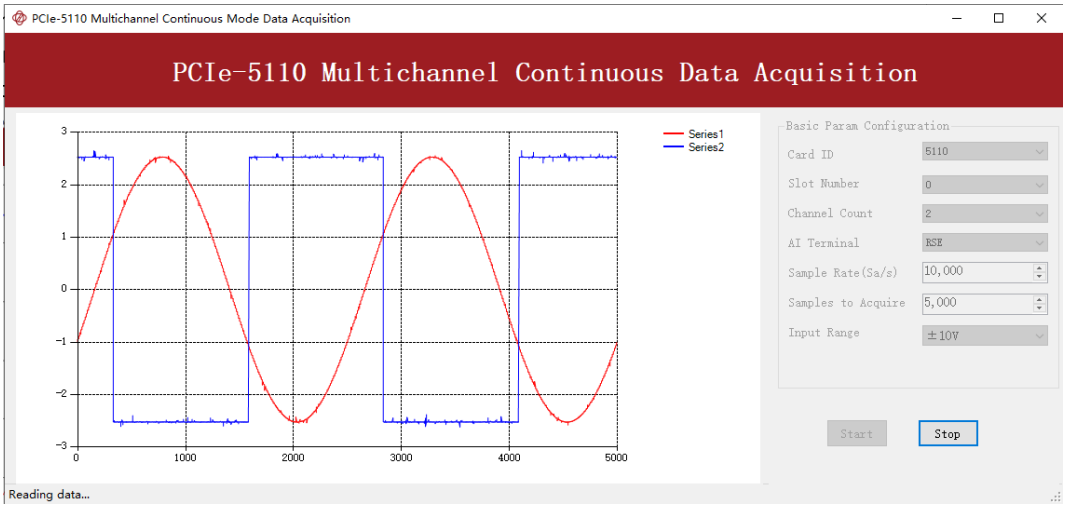


Figure 5 MultiChannel Continuous Acquisition

### 8.2.1. Continuous Acquisition

An AI acquisition task will acquire the data continuously until the task is stopped. The PCIe/PXIe-5114-H7 device will continue acquiring data and save the data in a

circular buffer. You specify how many samples to read back by the user buffer's length, if your program does not read the data fast enough, the circular buffer may overflow. In this case, the driver software will throw out an error message.

*Tip*: User buffer's length 1/10$^{th}$ to 1/4$^{th}$ *SampleRate* is a good start.

#### 8.2.2. Finite Acquisition

In the Finite Acquisition mode, an AI acquisition task will capture specific total number of samples by the parameter, SamplesToAcquire.

You can use the sample program **Analog Input --> Winform AI Finite** to learn more about Finite Acquisition.

#### 8.2.3. Single Point Acquisition

In the Single Acquisition mode, it is to capture a single sample for each acquisition.

You can use sample program: **Analog Input --> Console AI Single Point** to learn more about the single point Acquisition.

#### 8.2.4. Record Acquisition

AI Task will continuously capture the data and then save them to a storage disk. During the capturing process, user can preview the captured data randomly when the capturing process is available. The mode is particularly useful for high-speed acquisition and recording applications.

### 8.3. Analog Input Terminal Type

The PCIe/PXIe-5114-H7 provide 2 analog input terminal types:

- Differential (DIFF)
- Referenced Single-Ended (RSE)

The DIFF connection is recommended for ground-referenced signal sources and it is usually better in rejecting the common-mode noise. However, to acquire one input signal, two AI channels are required to form the differential pair. The RSE is recommended when the input signal sources are floating signals. In RSE mode, these floating signal sources all share the same ground reference (AI_GND). Because of it, the RSE mode can acquire twice as many channels than the DIFF mode. Appendix has more details on these 2 modes.

### 8.3.1. DIFF Mode

The DIFF mode connects signal's positive side to AI's positive input, signal's grounded negative side to AI's negative input as shown in Figure 6. The common noise appears on both positive and negative terminals of the differential amplifier; thus it will be cancelled out. Therefore, the DIFF mode has better signal-to-noise ratio (SNR). Please see Appendix for more explanations.

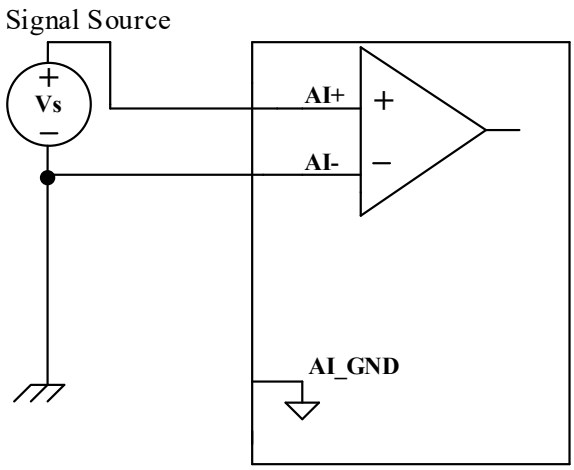


Figure 6 Differential Mode for Grounding Signals

The following figures show the actual collected voltage range diamond plots in DIFF mode for three ranges: ±10V(Figure 6), ±5V(Figure 6), and ±2.5V(Figure 6).The inputs at both ends Vin+ and Vin- must be within the range.

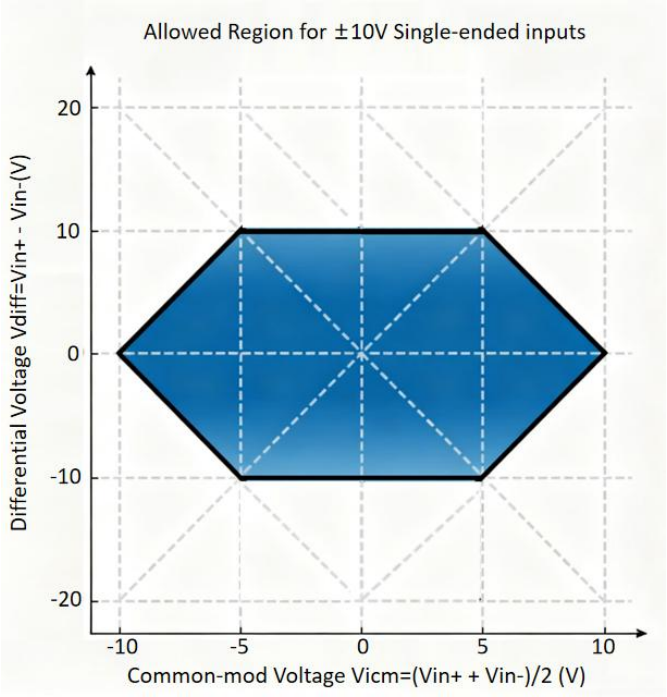


Figure 7 10V Range Diamond Plot

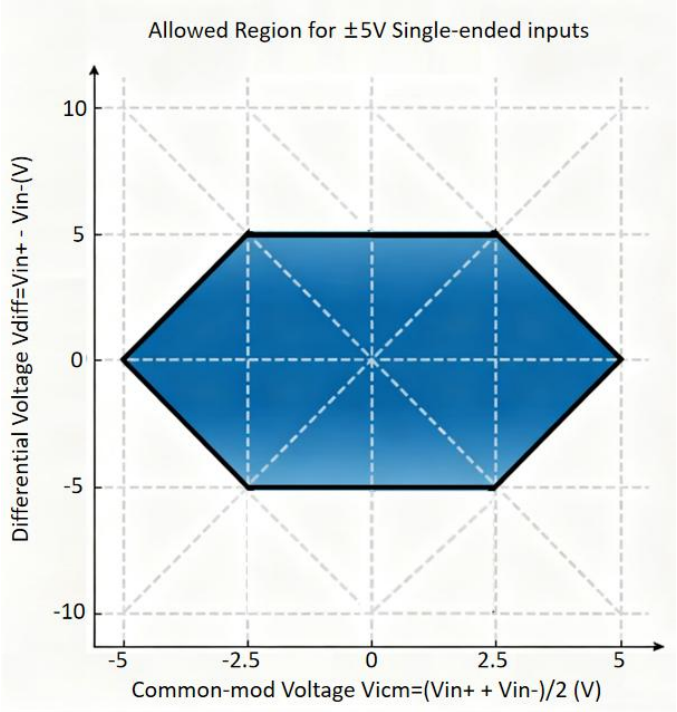


Figure 8 5V Range Diamond Plot

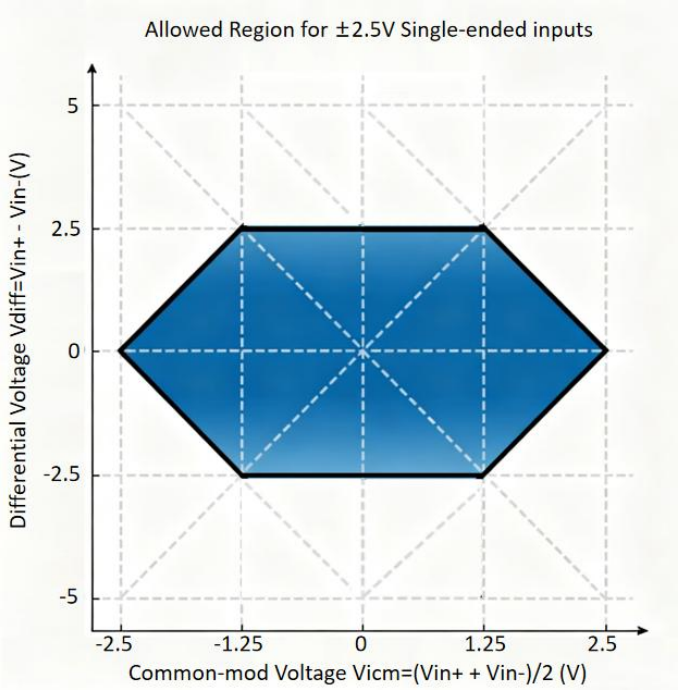


Figure 9 2.5V Range Diamond Plot

**Learn by Example 8.3.1**

- Open the program **Analog Input-->Winform AI Continuous MultiChannel**
- Connect the two signal source's positive outputs to PCIe-5114-H7 AI Ch0 (AI0+, Pin#18) and AI Ch1 (AI1+, Pin#19), two negative terminals to AI Ch0 negative (AI0-, Pin#52) and AI Ch1 negative (AI1-, Pin#53) as shown in Figure 3 and 错误!未找到引用源。. (AI0+, AI0-) and (AI1+, AI1-) consist of two pairs of DIFF inputs;
- Choose Differential in **AI Terminal**;
- Set other numbers as shown and click **start**.

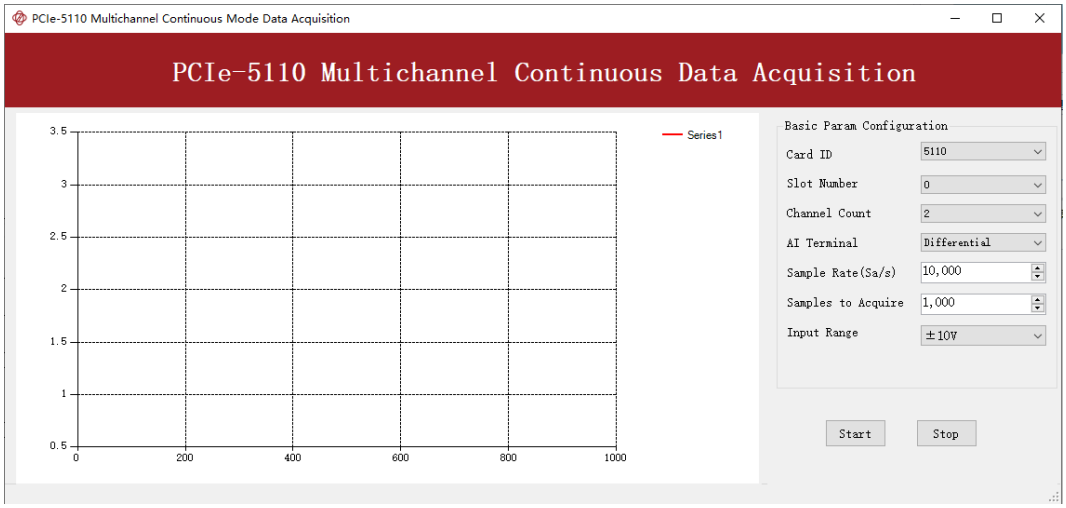


Figure 10 Choose Differential in AI Terminal

### 8.3.2. RSE Mode

In the RSE mode, all input signals' negative sides are connected to the AI ground of Instrumentation Amplifier, as shown in Figure 11. This mode works for measurements from floating sources. The RSE mode is suitable when these two conditions exist:

- The input signals are floating, meaning they are not connected to the ground
- When the common mode noise is low, meaning a clean environment.

The RSE mode offers twice as many measurement channels as the DIFF mode. Please see Appendix for more explanations.

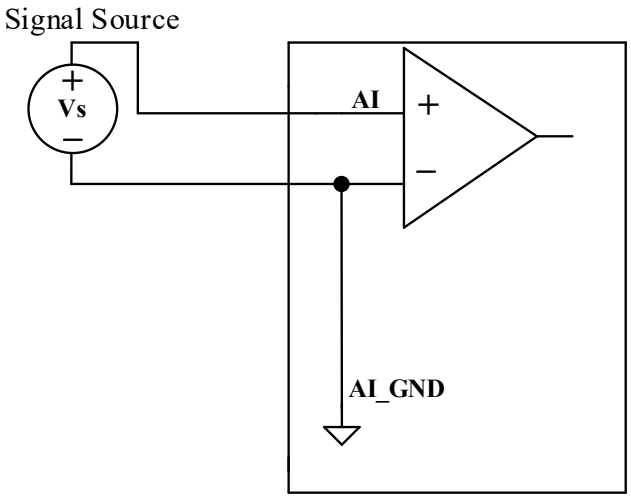


Figure 11 RSE Mode for Floating Signals

## 8.4. Trigger Source

There are 4 trigger types: Immediate trigger, Software trigger, Analog trigger, and Digital trigger. The trigger type is a property and set by driver software.

### 8.4.1. Immediate trigger

This trigger mode does not require configuration and is triggered immediately when an operation starts. The operation can be AI, AO, DI, DO, CI, CO etc.

**Learn by Example 8.4.1**

- Use the same program and connection as in
- **Learn by Example8.2.**

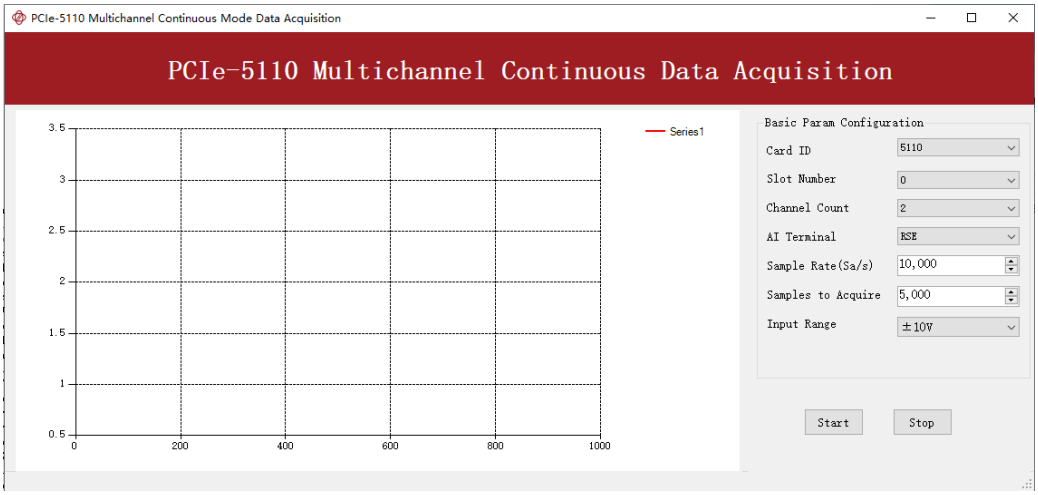


Figure 12 Immediate trigger Paraments

➢ With Immediate trigger you can click **Start** to generate the task instead of sending a trigger signal.

### 8.4.2. Software Trigger

A software trigger must be configured by the driver software. The trigger starts when a trigger software routine is called.

**Learn by Example 8.4.2**

- Connect the signal source's positive terminal to PCIe-5114-H7 AI Ch0 (AI0+, Pin#18), the negative terminal to the ground (AI_GND, Pin#17) as shown in Figure 3 and 错误!未找到引用源。. (AI0+, AI_GND) consists of a RSE input.
- Set a sinewave signal (f=4Hz, Vpp=5V).
- Open **Analog Input-->Winform AI Continuous Soft Trigger**, set the following numbers as shown.
- Click **Start** to run the task.

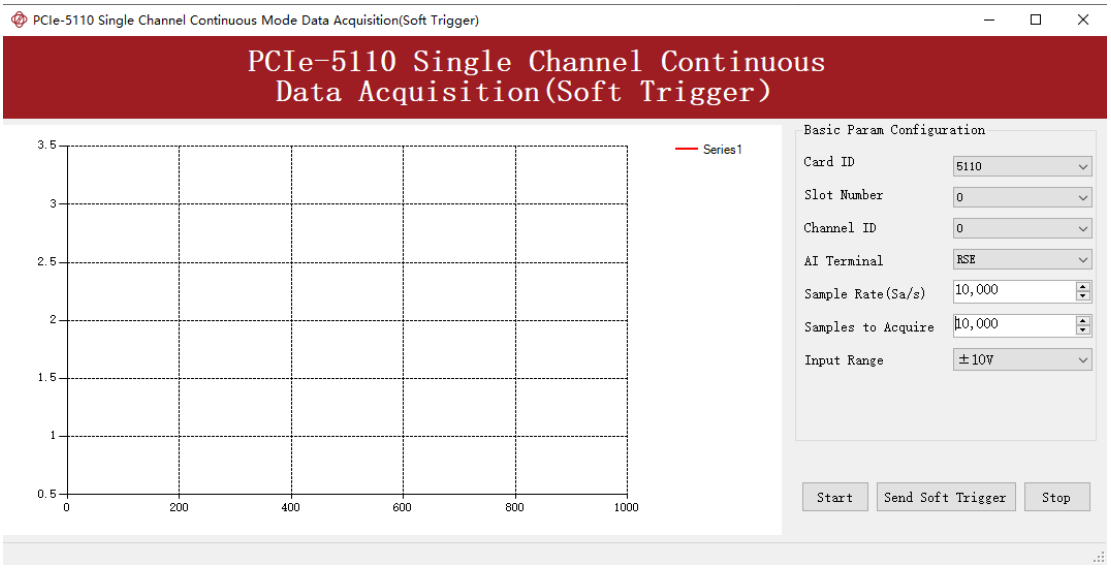


Figure 13 Software trigger Paraments

- Data will not be acquired until there is a positive signal from *Software Trigger* when **Send Soft Trigger** is clicked.

- After sending the trigger signal, the result will be like this:

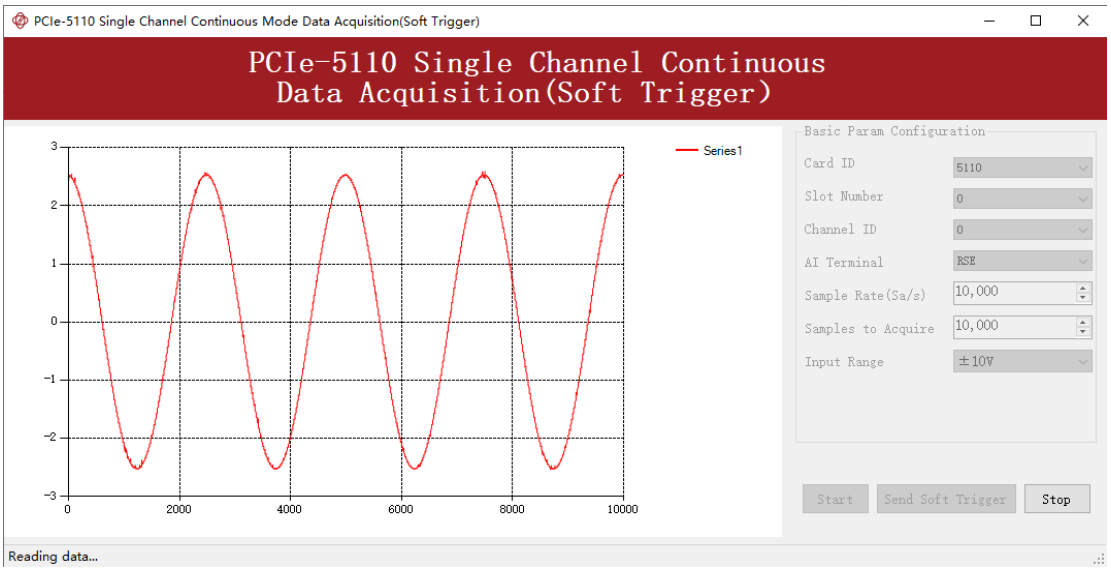


Figure 14 Software trigger Acquisition

### 8.4.3. External Analog Trigger

You can assign one of measurement channels as the analog trigger source.

PCIe/PXIe-5114-H7 provides three analog trigger modes:

- Edge comparator,
- Hysteresis comparator,
- Window comparator.

Analog trigger threshold range can be arbitrarily selected in the effective range of the selected channel. When setting the threshold, please pay attention to the physical unit currently in use.

**Edge comparator**

In the Edge comparator, there are two trigger conditions: *Rising Slope Trigger* and *Falling Slope Trigger*.

*Rising Slope Trigger*: The Edge comparator output is high when the signal goes above the threshold; the output is low when the signal goes below the threshold as shown in Figure 15.

*Falling Slope Trigger*: The Edge comparator output is high when the signal goes below the threshold; the output is low when the signal goes above the threshold as shown in Figure 16.

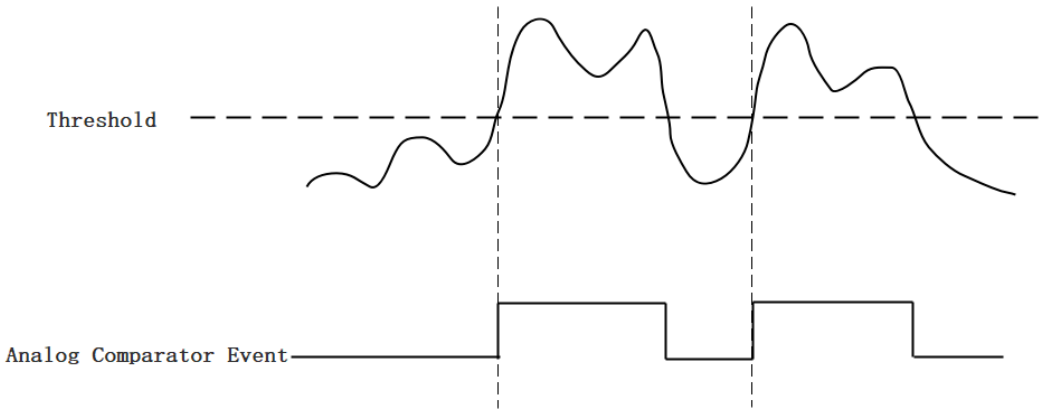


Figure 15 Rising Slope Trigger

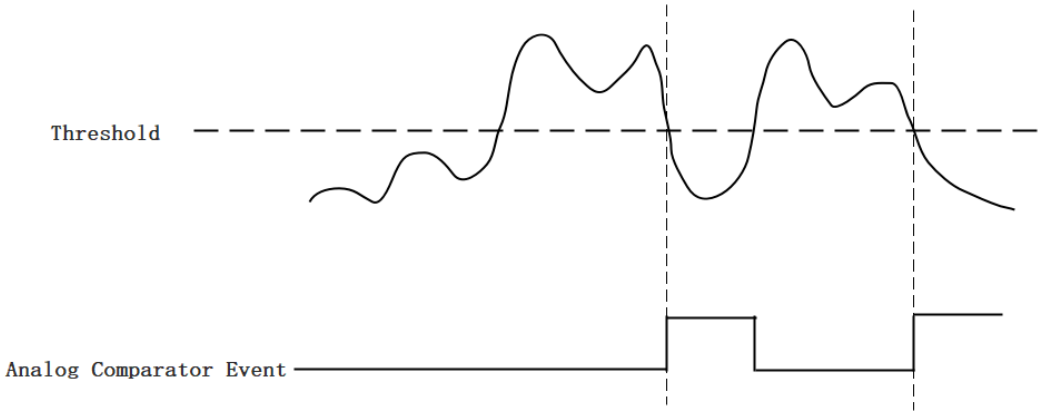


Figure 16 Falling Slope Trigger

**Hysteresis Comparator**

The hysteresis comparator is designed for preventing spurious triggering. You can set hysteresis region by setting high threshold and low threshold. There are two trigger conditions: *Hysteresis with Rising Slope Trigger* and *Hysteresis with Falling Slope Trigger*.

*Hysteresis with Rising Slope Trigger*: The Hysteresis comparator output is high when the signal must first be below the low threshold, then goes above the high threshold. The output will change to low when the signal goes below the low threshold as shown in Figure 17.

*Hysteresis with Falling Slope Trigger*: The Hysteresis comparator output is high when the signal must first be above the high threshold, then goes below the low threshold. The output will change to low when the signal goes above the high threshold as shown in Figure 18.

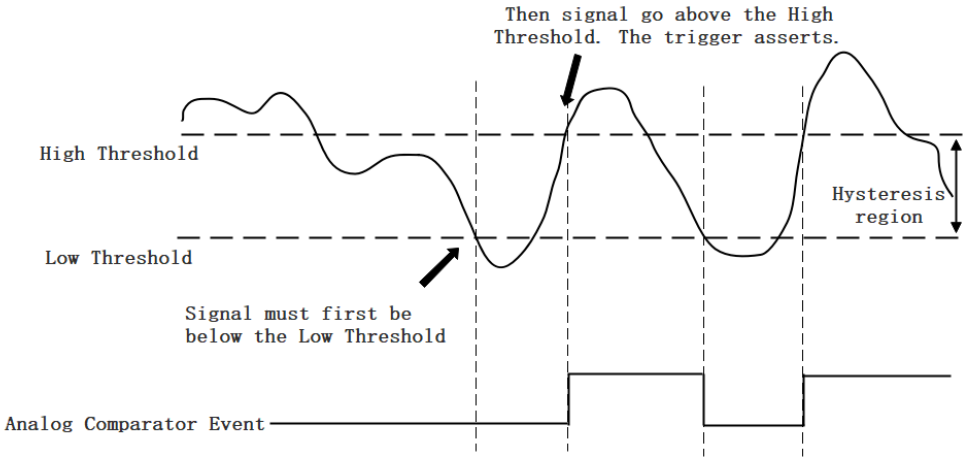


Figure 17 Hysteresis with Rising Slope Trigger

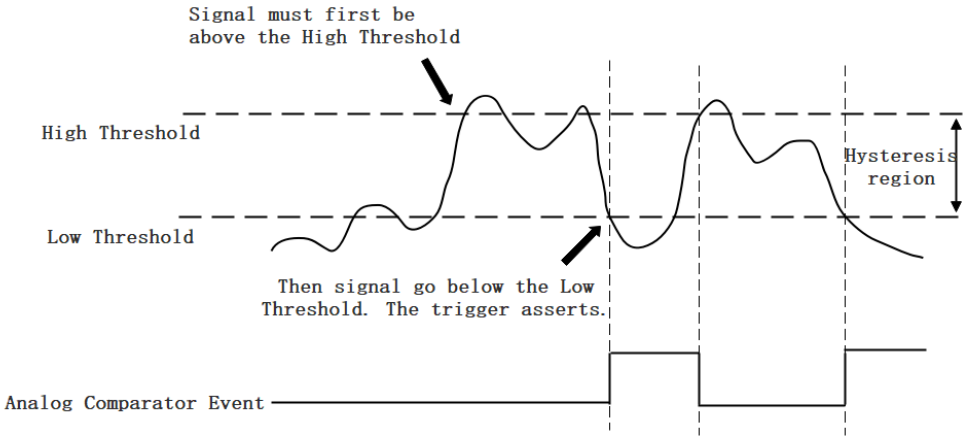


Figure 18 Hysteresis with Falling Slope Trigger

**Window comparator**

The window comparator is designed to acquire signal from interesting window by setting High Threshold and Low Threshold. There are two trigger conditions: *Entering Window Trigger* and *Leaving Window Trigger*.

*Entering Window Trigger*: The window comparator output is high when the signal enters the window defined by the *Low Threshold* and *High Threshold*. The output will change to low when the signal leaves the window as shown in Figure 19.

*Leaving Window Trigger*: The window comparator output is high when the signal leaves the window defined by the *Low Threshold* and *High Threshold*. The output will

change to low when the signal enters the window as shown in Figure 20 Leaving Window Trigger.

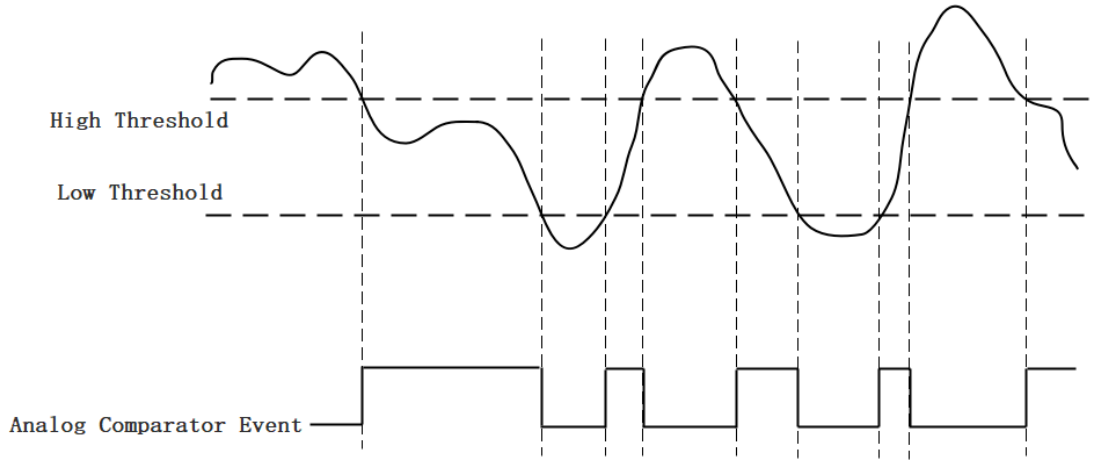


Figure 19 Entering Window Trigger

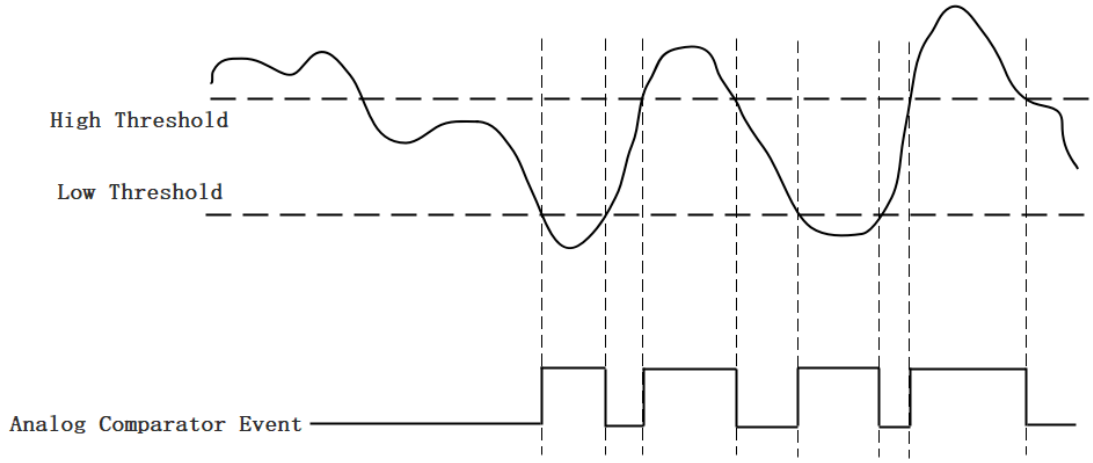


Figure 20 Leaving Window Trigger

**Learn by Example 8.4.3**

- Connect the signal source's positive terminal to PCIe-5114-H7 AI Ch0 (AI0+, Pin#18), the negative terminal to the ground (AI_GND, Pin#17) as shown in Figure 3 and 错误!未找到引用源。. (AI0+, AI_GND) consists of a RSE input.
- Set a sinewave signal (f=4Hz, Vpp=5V).
- Open **Analog Input-->Winform AI Continuous Analog Trigger**, set the following numbers as shown.

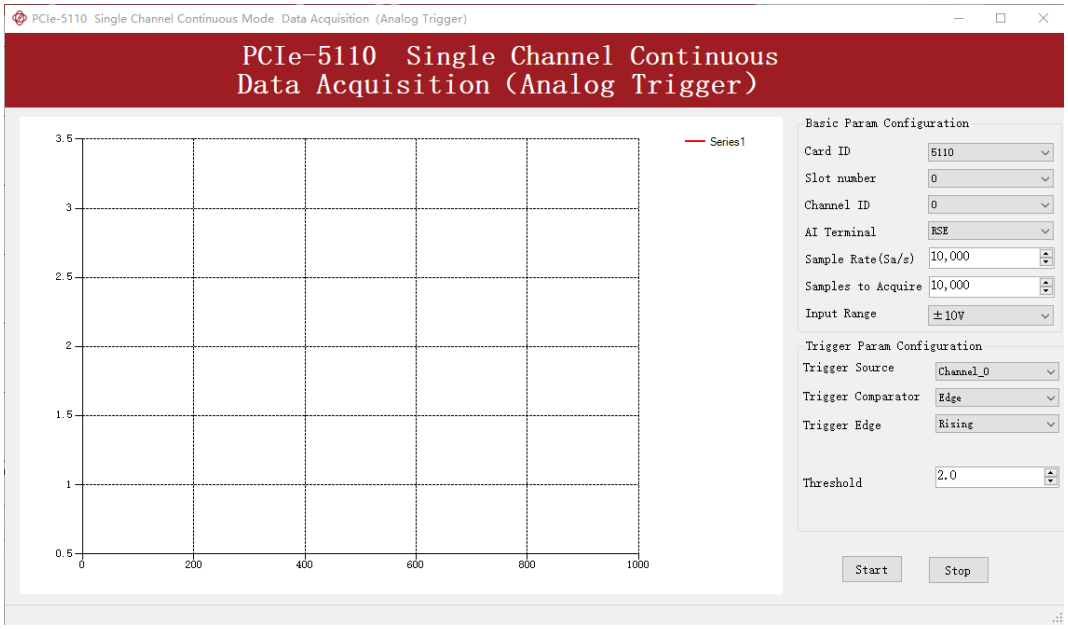


Figure 21 Analog Trigger Paraments

- Modes of the *Analog Trigger* are set by **Trigger Comparator.** Set it to **Edge**.
- The edge of *EdgeComparator* set by **Trigger Edge**. (**Rising** and **Falling**)
- **Trigger source** can be any channel of PCIe-5114-H7 analog input. Set it to **Channel_0**.
- According to the rules of **Rising** mentioned above, the signal acquisition will not start until it raises to 2.0 V, which is set by **Threshold** above.

■ Click Start, a message will appear in the lower left corner:

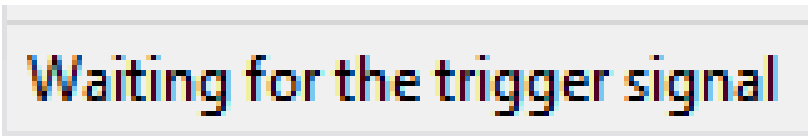


Figure 22 Waiting For Trigger

- This indicates the data acquisition will start only after a triggering event. In this example a trigger signal will occur when the *hysteresis comparator* meets the condition explained in **8.4.3**.

■ The result is shown below:

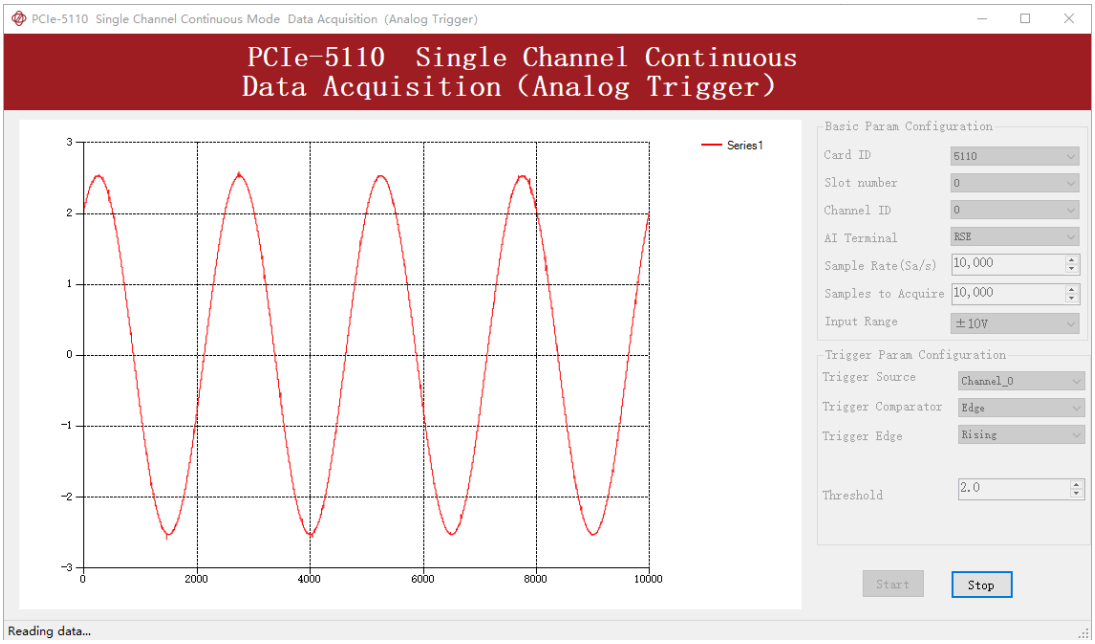


Figure 23 Analog Trigger Acquisition

- The signal starts at 2.0V, which matches the Edge mode set before.

### 8.4.4. External Digital Trigger

PCIe/PXIe-5114-H7 supports different external digital trigger sources from PXI Trigger bus (PXI_TRIG<0..7>), PXI_STAR and connectors of front panel (PFI). The high pulse width of digital trigger signal must be longer than 20 ns for effective trigger. The module will monitor the signal on digital trigger source and wait for the rising edge or falling edge of digital signal which depending on the set trigger condition, then cause the module to acquire the data as shown in Figure 24


Figure 24 External Digital Trigger

**Learn by Example 8.4.4**

- Connect the signal source two positive terminals to PCIe-5114-H7 AI Ch0, (AI0+, Pin#18) and digital trigger source (PFI 0, Pin#1), two negative terminals to the ground of analog input (AI_GND, Pin#17) and the ground of digital input/output (DGND, Pin#9) as shown in Figure 3 and 错误!未找到引用源。 (AI0+, AI_GND) consists of a RSE input. (PFI0, DGND) provides the trigger signal.
- Set a sinewave signal (f=4Hz, Vpp=5V) and a squarewave signal (f=4Hz, Vpp=5V).
- Open **Analog Input-->Winform AI Continuous Digital Trigger**, set the following numbers as shown.

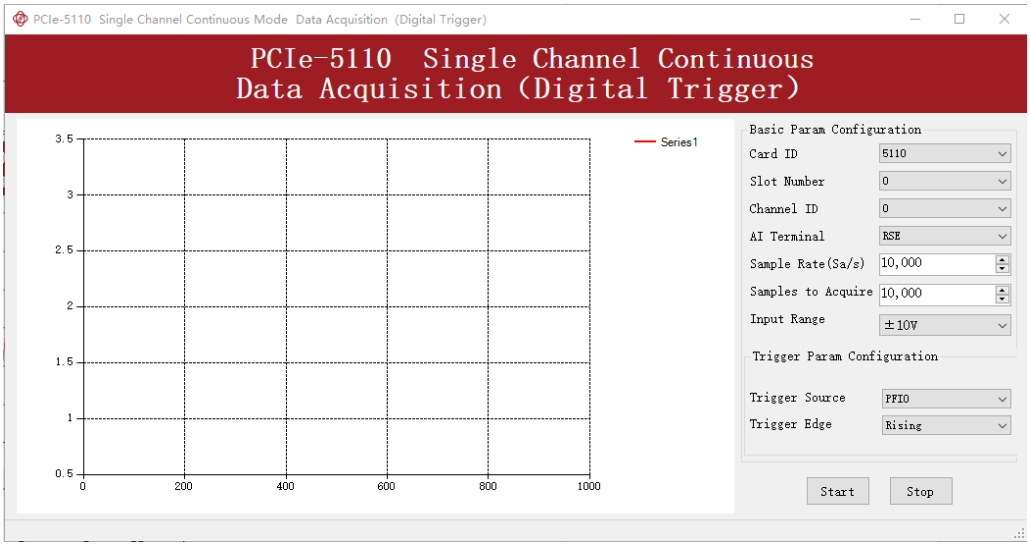


Figure 25 Digital Trigger Paraments

- **Trigger Source** must match the pin on 5110.
- There are two **Trigger Edge**: **Rising** and **Falling**.

- Click **Start** and the result shows below:

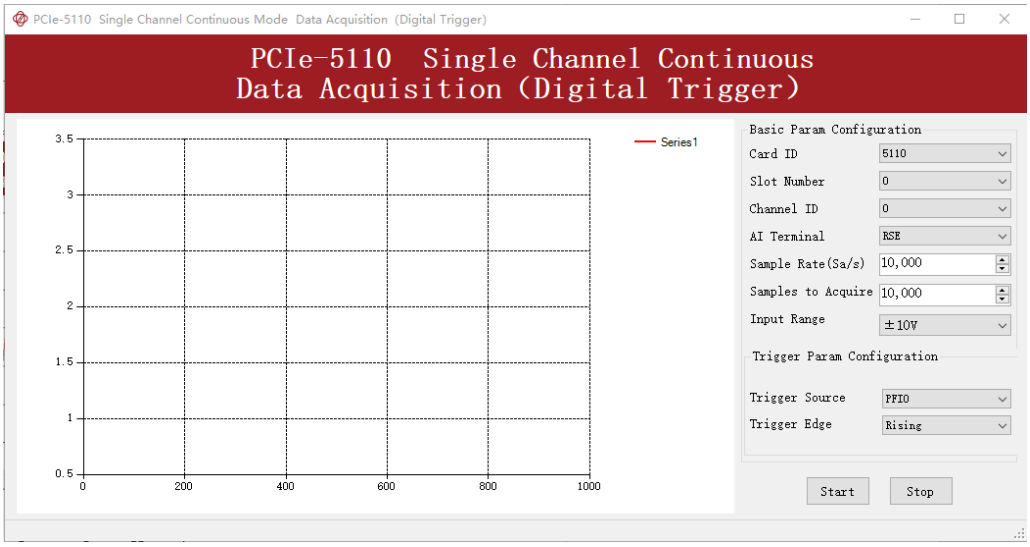


Figure 26 Digital Trigger Acquisition

- Since the squarewave is used for the digital trigger source, when a rising edge of the squarewave occurs, the digital trigger will be activated, and the data acquisition will start.

## 8.5. Trigger Mode

The PCIe/PXIe-5114-H7's analog inputs support several trigger modes: start trigger, reference trigger, and re-trigger.

### 8.5.1. Start Trigger

In this mode, data acquisition begins immediately after the trigger. This trigger mode is suitable for continuous acquisition and finite acquisition. As shown in Figure 27.

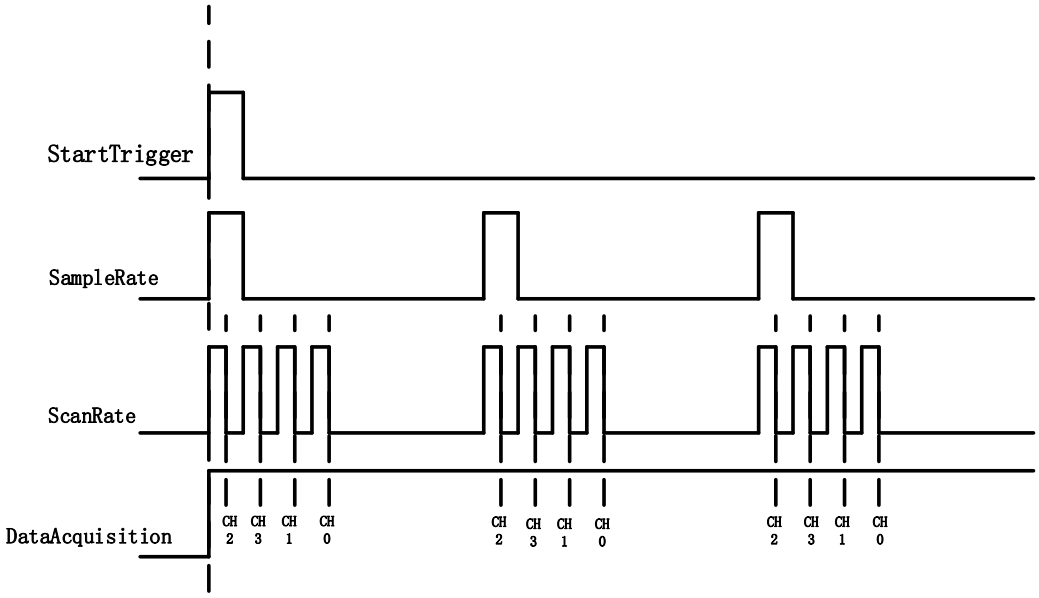


Figure 27 Start Trigger

### 8.5.2. Reference Trigger

This trigger mode is suitable for finite acquisition. In this mode, user can set the number of pre-trigger samples. The default number of pre-trigger points is 0. First you need to start the data acquisition. When the reference trigger condition is met, the routine will return the acquired data points. If when the points less than the pre-trigger samples, the trigger signal be ignored. An example is show below.

Example

- Total samples: 1000;
- Channel Count: 1
- Pre-trigger samples: 10;
- After triggering, it returns total 1000 samples, 10 being pre-triggered, 990 after triggering

The principle is shown in Figure 28.

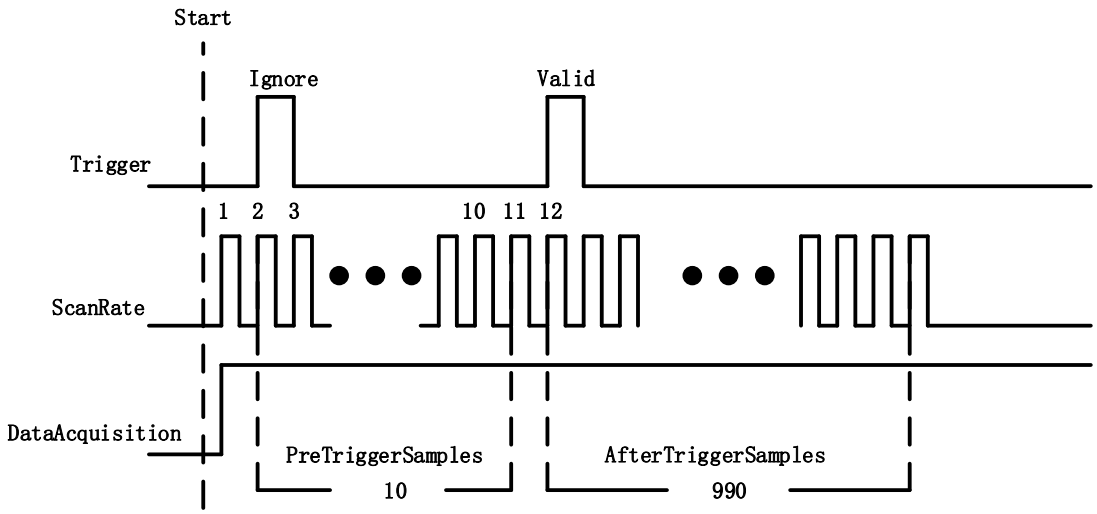


Figure 28 Reference Trigger

### 8.5.3. ReTrigger

PCIe/PXIe-5114-H7 supports retrigger mode. In the retrigger mode, you can set the number of retrigger and the length of each acquisition. Assuming that the number of re triggers is n and the length of each trigger acquisition is m, the length of all acquisition data is n * m * channelcount. Show in Figure 29.

When the number of retrigger is - 1, it is infinite.

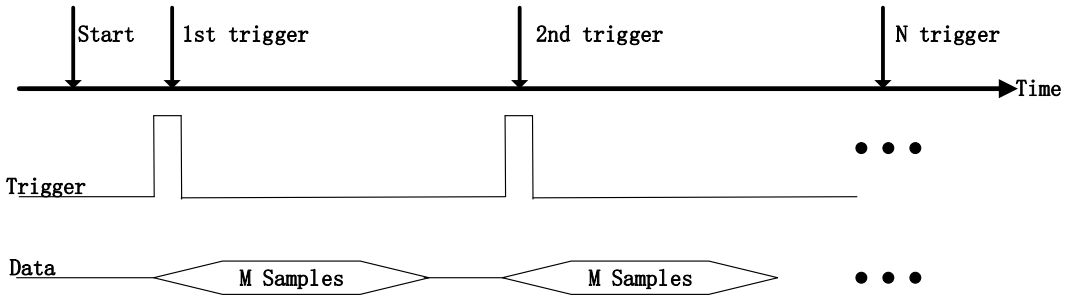


Figure 29 ReTrigger

### Learn by Example 8.5

Connect the signal source's positive terminal to PCIe-5114-H7 AI Ch0 (AI0+, Pin#18), the negative terminal to the ground (AI_GND, Pin#17) as shown in Figure 3 and 错误!未找到引用源。. (AI0+, AI_GND) consists of an RSE input.

- Set a sinewave signal (f=4Hz, Vpp=5V).
- Open **Analog Input-->Winform AI Finite Analog Trigger**, set the following numbers as shown.

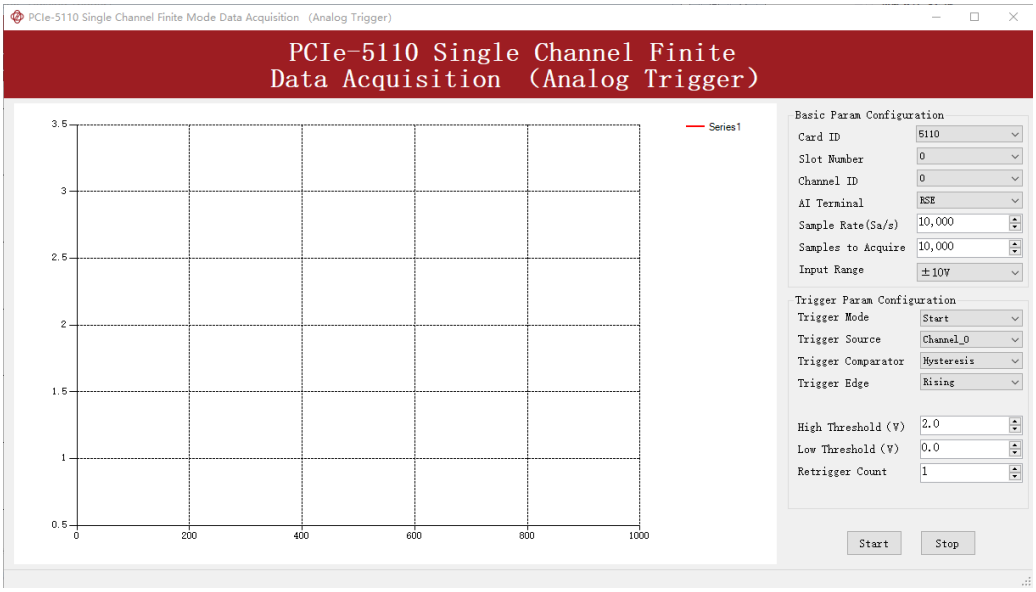


Figure 30 Retrigger Paraments

- You can use three different kinds of triggers in this program as mentioned in **8.5**. *Start Trigger* and *Reference Trigger* can be set by **Trigger Mode**. For *ReTrigger* can be used by changing the numbers in **Retrigger Count**.

- PretriggerSamples is set by **Pretrigger Samples**.

- Now the trigger is a **Start Trigger**. Click **Start** to begin the data acquisition, the result is shown below:

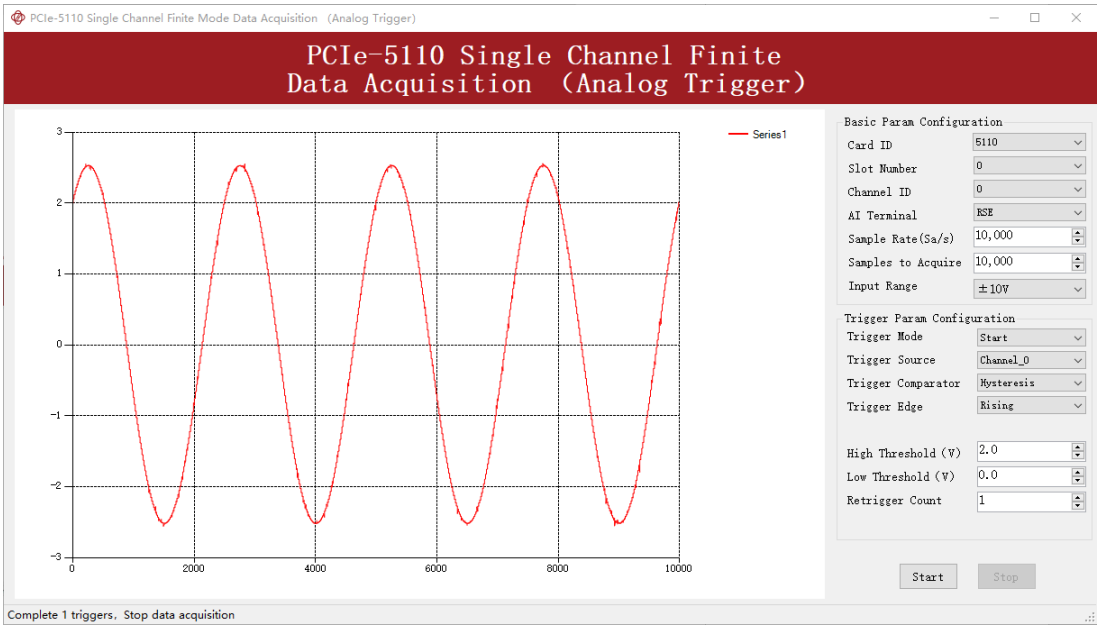


Figure 31 Retrigger In Start Trigger Mode

- Now change the **Trigger Mode** to **Reference** mode with **Pretrigger Samples** 1000. A different result shows below:

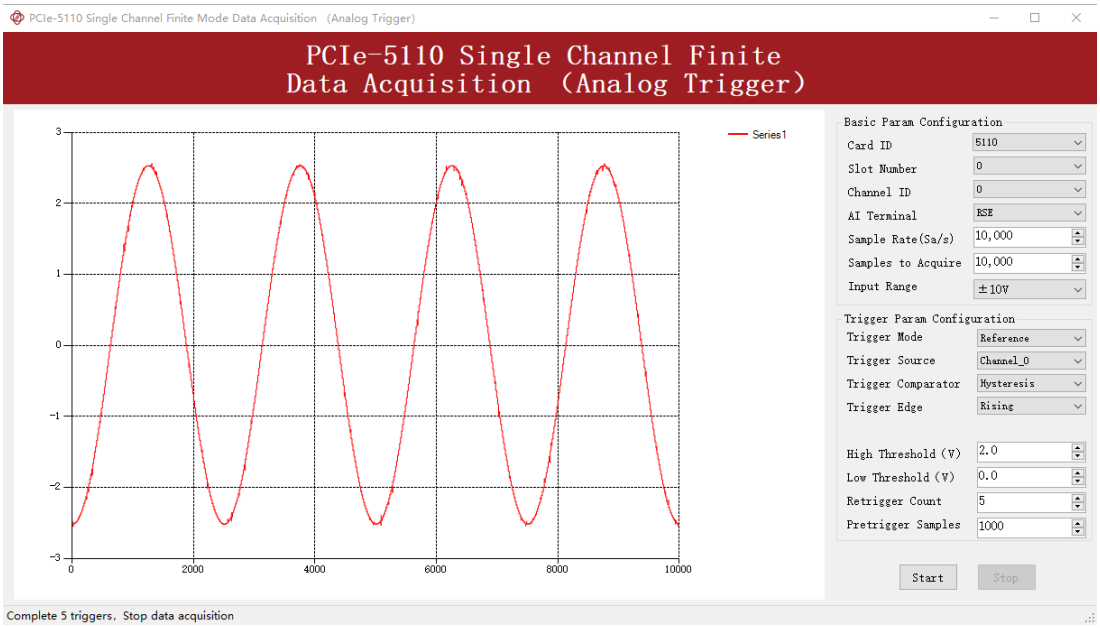


Figure 32 Retrigger In Reference Trigger Mode

- You can see the horizontal movement between two signals due to the change of **Trigger Mode**.

- Now change the mode of trigger to *Retrigger* through giving **Retrigger Count** a number other than 0 and click **Start**. A message will appear in the lower left corner: "Complete the $n^{th}$ trigger".

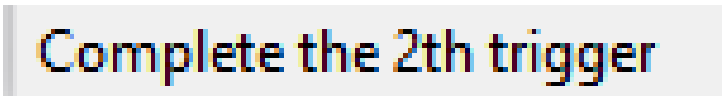


Figure 33 Complete Retrigger Count

  - It shows the acquisition process through every trigger signal.

## 8.6. AO Operations

The PCIe/PXIe-5114-H7 AO provides 16-bit simultaneous outputs. The analog output has three modes of operation: Finite, ContinuousWrapping, and ContinuousNoWrapping.

### 8.6.1. Finite Output

The finite output requires the user to write a piece of data. After starting the AO, it starts to output the written data until the output is completed.

**Learn by Example 8.6.1**

- Connect PCIe-5114-H7 AO Ch0 (AO0, Pin #49) to AI Ch0 (AI0+, Pin#18), Ground of AO0 (AO_GND, Pin#15) to Ground of AI0 (AI_GND, Pin#17). (AI0+, AI_GND) consists of a RSE input; (AO0, AO_GND) consists of an output.
- PCIe-5114-H7 sends an analog signal through (AO0, AO_GND) and reads back the signal from (AI0+, AI_GND).
- Open **Analog Input-->Winform AI Continuous**, set the following numbers as shown.

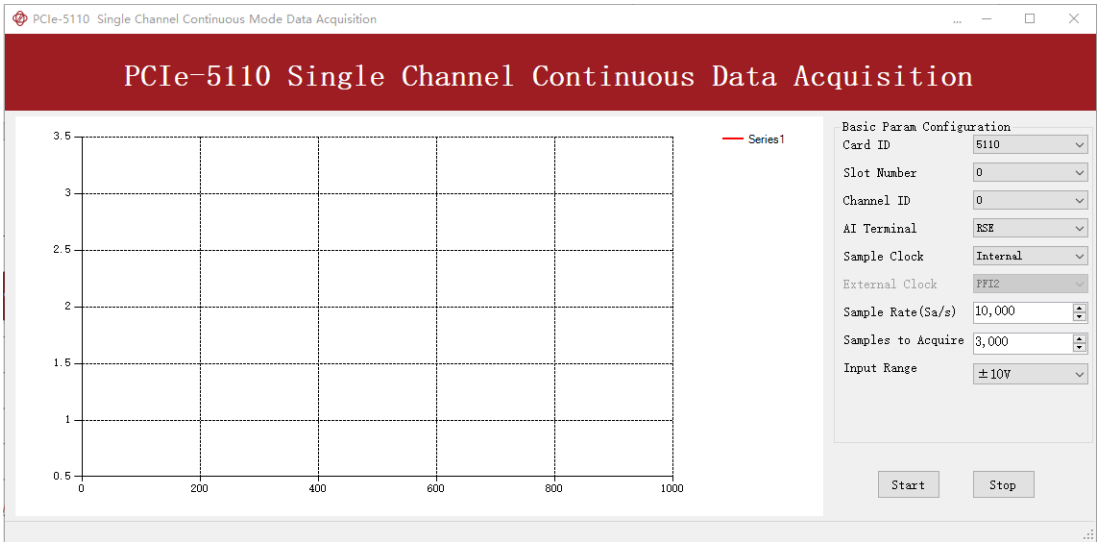


Figure 34 AI Continuous Paraments

- Click **Start** to start the data acquisition.

- Open **Analog Output-->Winform AO Finite**, set the following numbers as shown:

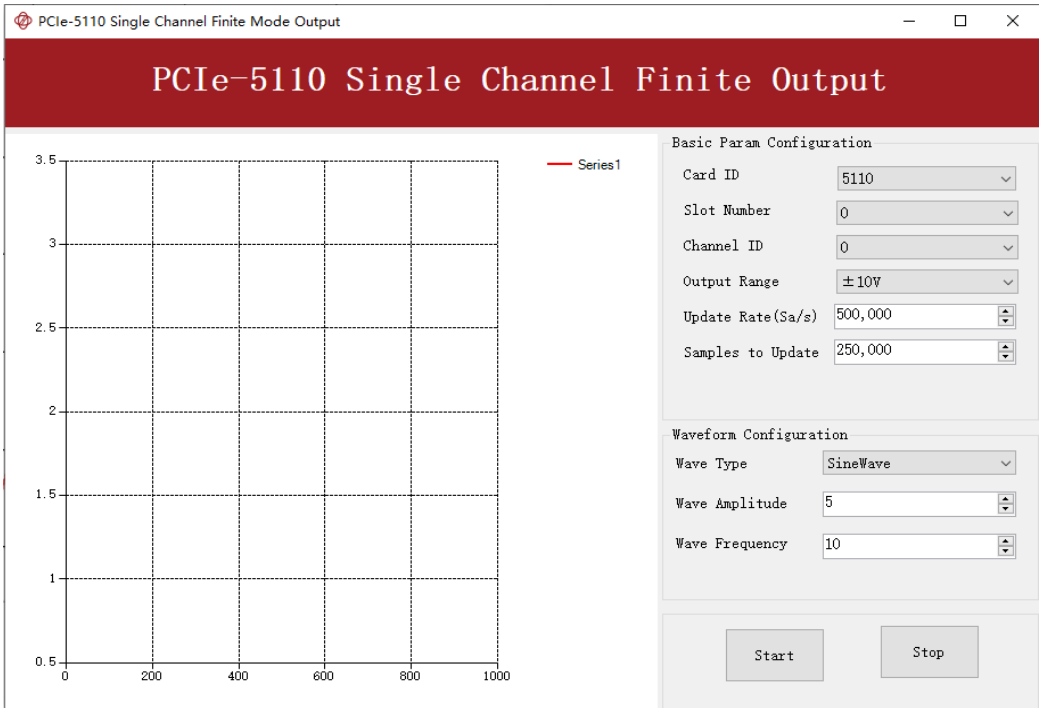


Figure 35 AO Finite Output Paraments

- Click **Start** to generate a **SineWave**. The generated signal is shown below:

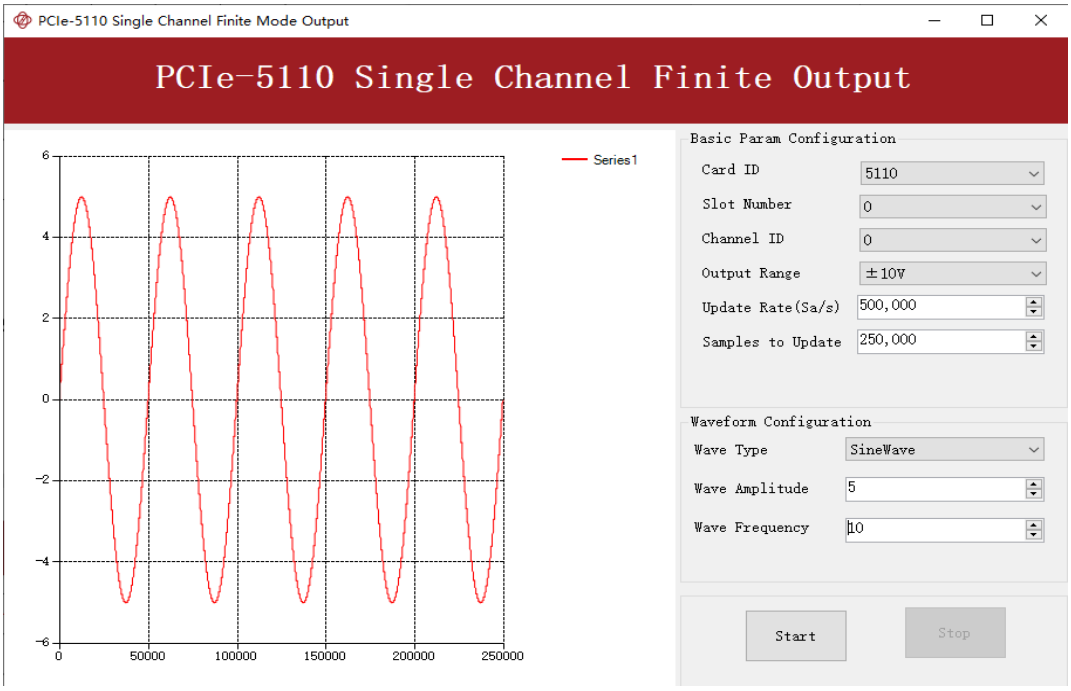


Figure 36 AO Finite Signal

- And the received signal is shown below.

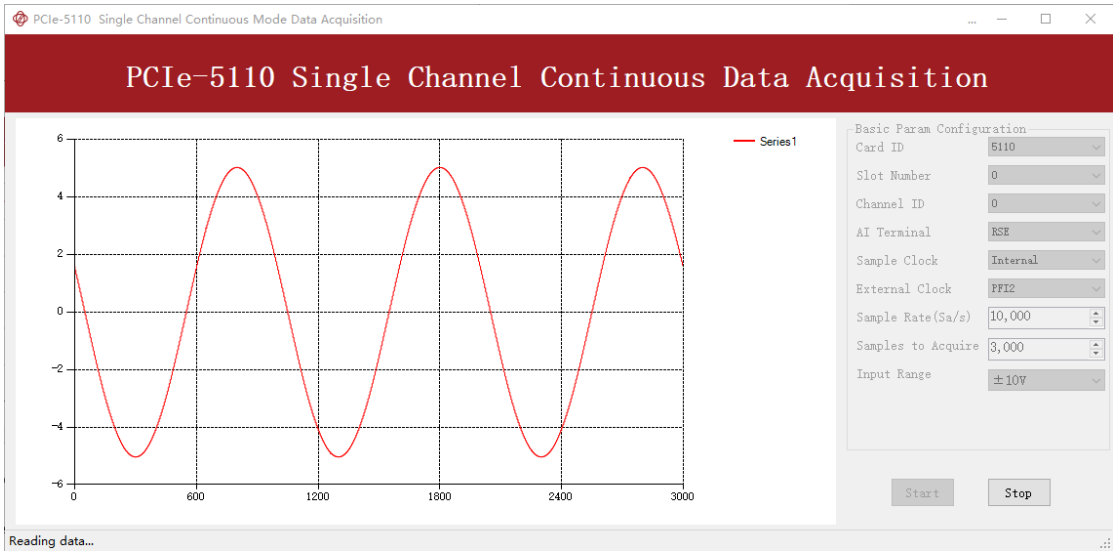


Figure 37 AI Acquisition Signal

- The analog signal is successfully generated and received by PCIe-5114-H7.

### 8.6.2. Continuous NoWrappping Output

The continuous acyclic output needs to write a piece of data before starting the AO. After the AO starts, user needs to continuously write new data to ensure the continuous output of the AO.

**Learn by Example 8.6.2**

- Connect PCIe-5114-H7 AO Ch0 (AO0, Pin #49) to AI Ch0 (AI0+, Pin#18), Ground of AO0 (AO_GND, Pin#15) to Ground of AI0 (AI_GND, Pin#17). (AI0+, AI_GND) consists of a RSE input; (AO0, AO_GND) consists of an output.
- PCIe-5114-H7 sends an analog signal through (AO0, AO_GND) and reads back the signal from (AI0+, AI_GND).
- Open **Analog Input-->Winform AI Continuous**, set the following numbers as shown.

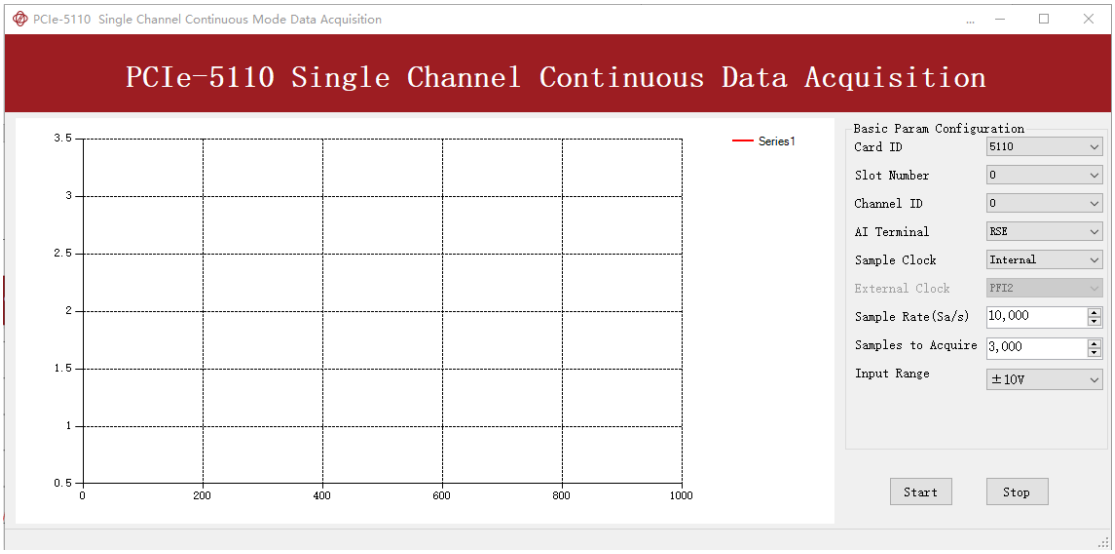


Figure 38 AI Continuous Paraments

- Click **Start** to start the data acquisition.
- Open **Analog Output-->Winform AO Continuous NoWrapping**, set the following numbers as shown:

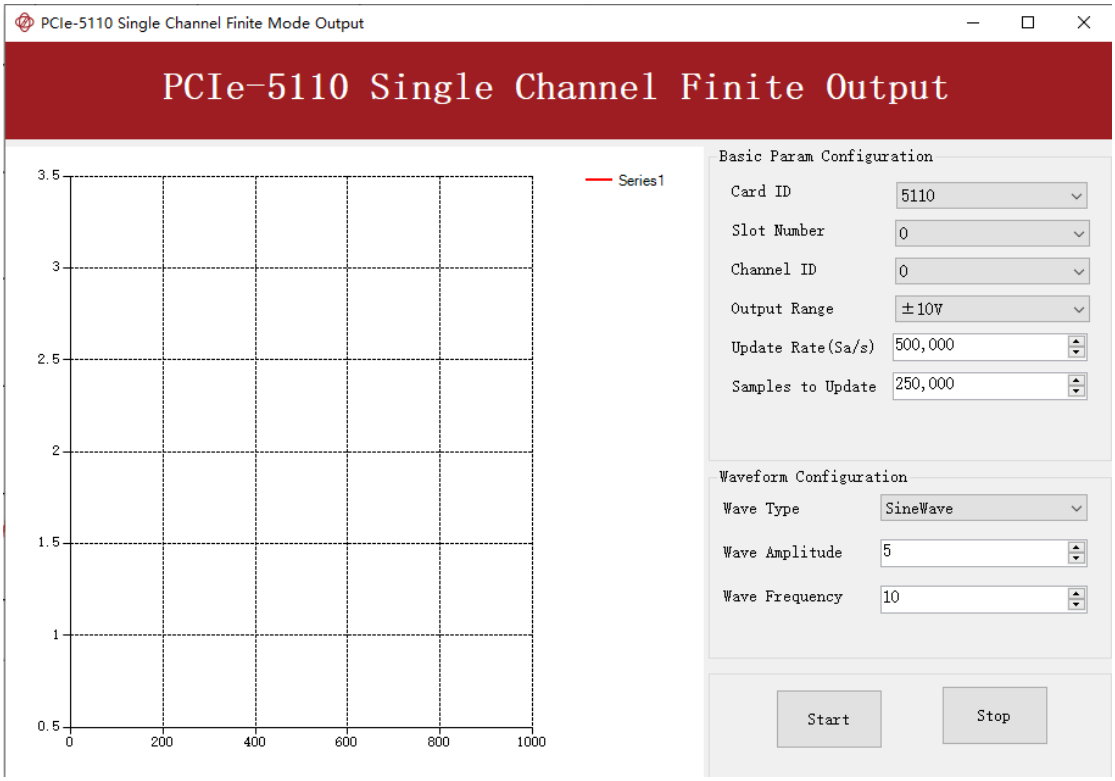


Figure 39 AO ContinuousNoWrapping Output Paraments

- In no wrapping analog output you can change the parameter of the signal whenever you want in **Waveform Configuration** when generating the wave. After the configuration you should click **Update** to apply the changes.

- Click **Start** to generate a sine wave first. The result is shown below.

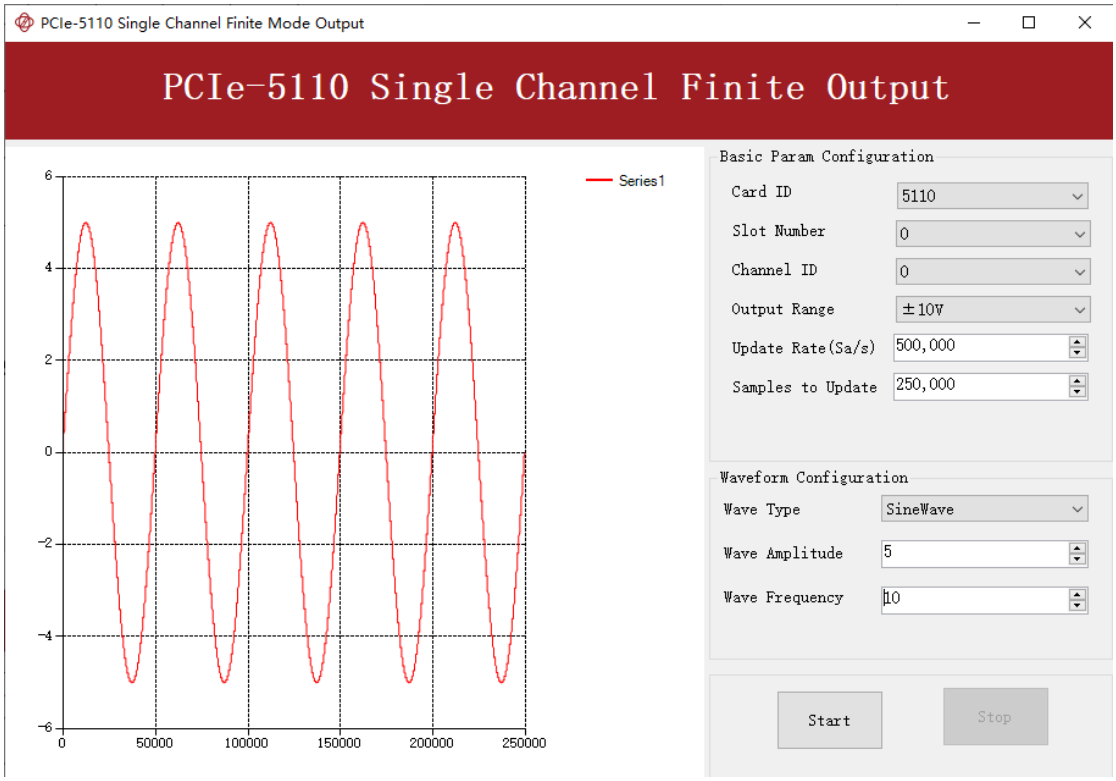


Figure 40 AO ContinuousNoWrapping Signal

- And the received signal is shown below.

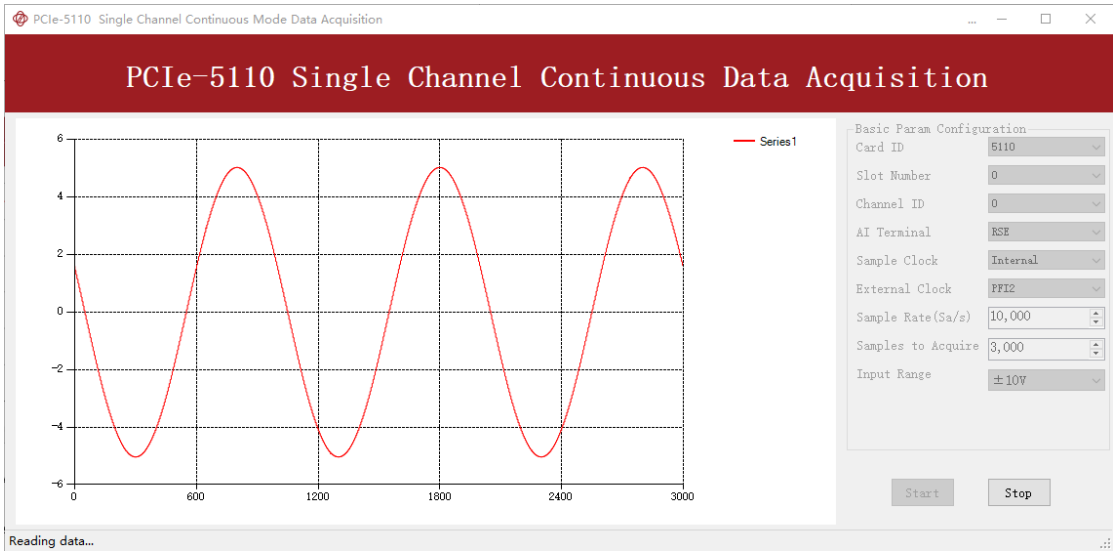


Figure 41 AI Acquisition AO Sin Signal

- Now change the **Wave Type** to **SquareWave** and click **Update** to generate it. The result is shown below.

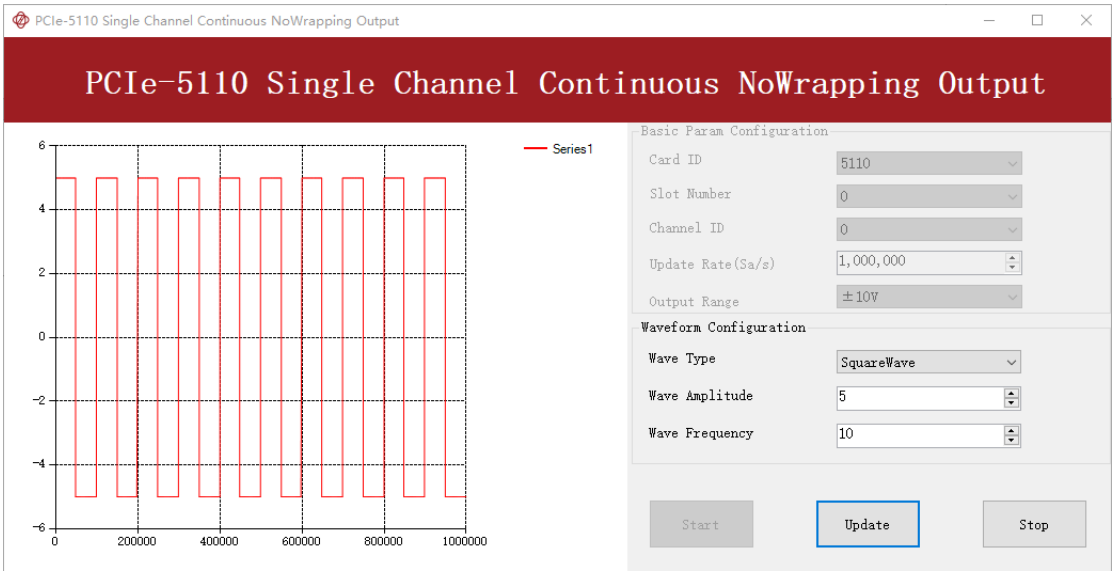


Figure 42 Update AO Square Signal

■ And the received signal is shown below.

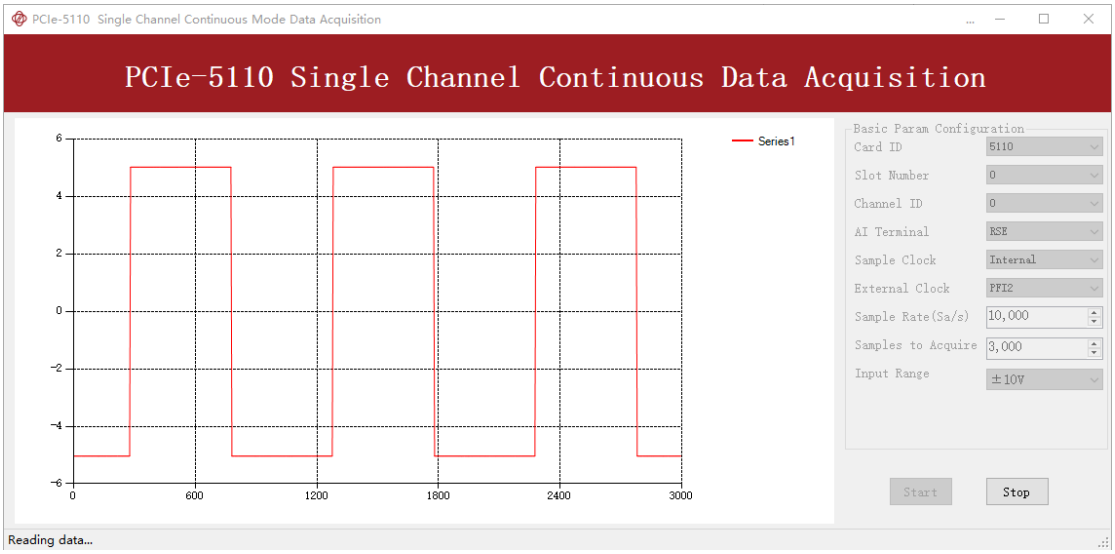


Figure 43 AI Acquisition AO Square Signal

➢ The analog signal is successfully generated and received by PCIe-5114-H7.

### 8.6.3. Continuous Wrapping Output

The continuous loop output first writes a piece of data before starting the AO. After the AO starts, the board will repeatedly output this data until user sends a stop command.

**Learn by Example 8.6.3**

- Connect PCIe-5114-H7 AO Ch0 (AO0, Pin #49) to AI Ch0 (AI0+, Pin#18), Ground of AO0 (AO_GND, Pin#15) to Ground of AI0 (AI_GND, Pin#17). (AI0+, AI_GND) consists of a RSE input; (AO0, AO_GND) consists of an output.
- PCIe-5114-H7 sends an analog signal through (AO0, AO_GND) and reads back the signal from (AI0+, AI_GND).
- Open **Analog Input-->Winform AI Continuous**, set the following numbers as shown.

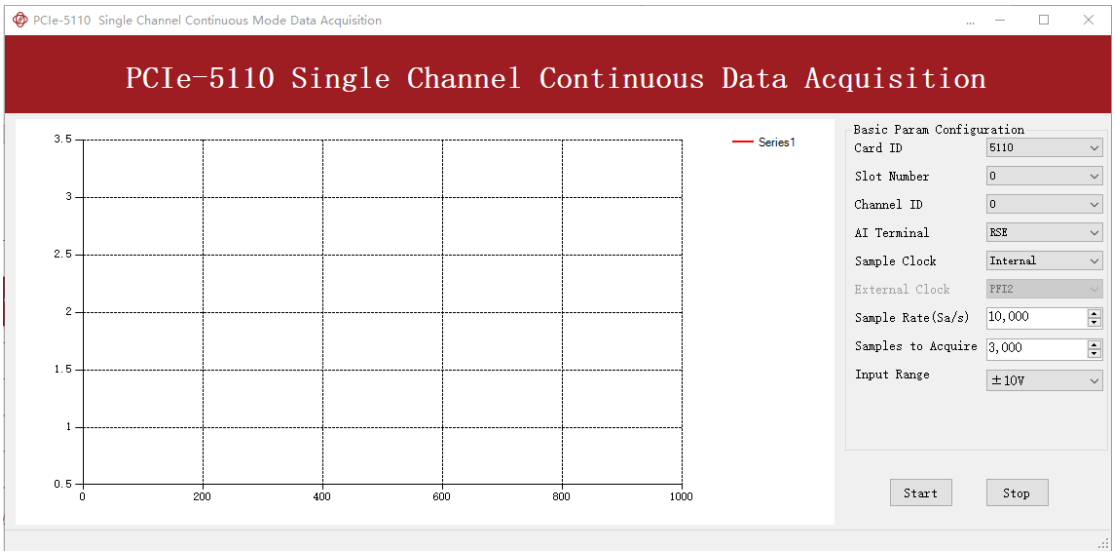


Figure 44 AI Continuous Paraments

- Click **Start** to start the data acquisition.
- Open **Analog Output-->Winform AO Continuous Wrapping**, set the numbers as shown.

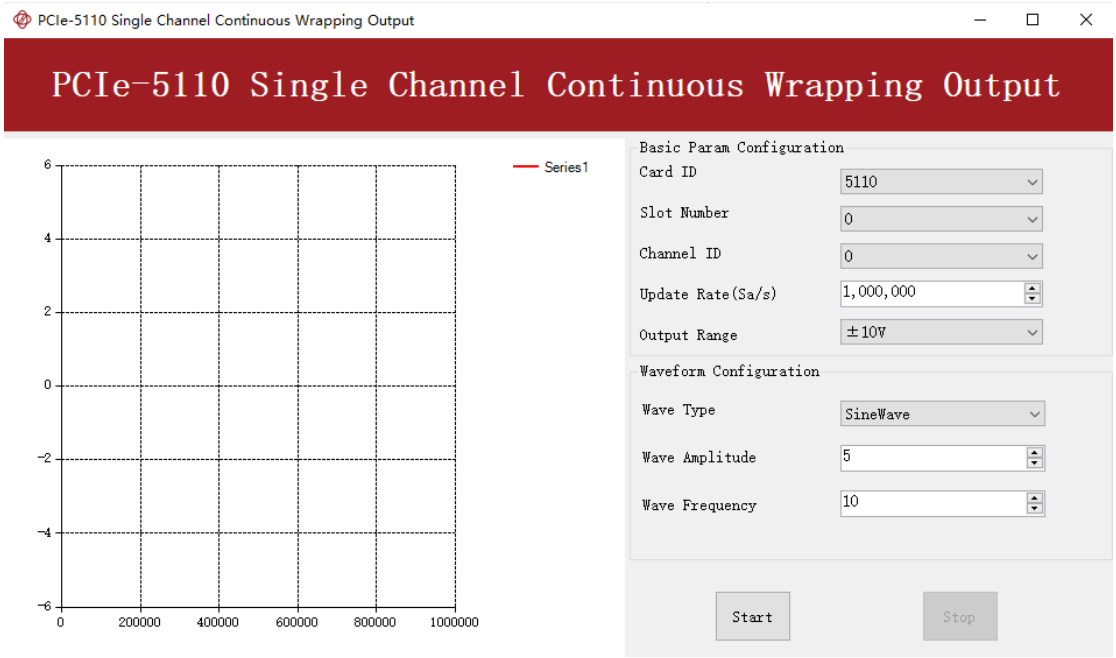


Figure 45 AO Continuous Wrapping Paraments

- Click **Start** to generate the signal. The result is shown below.

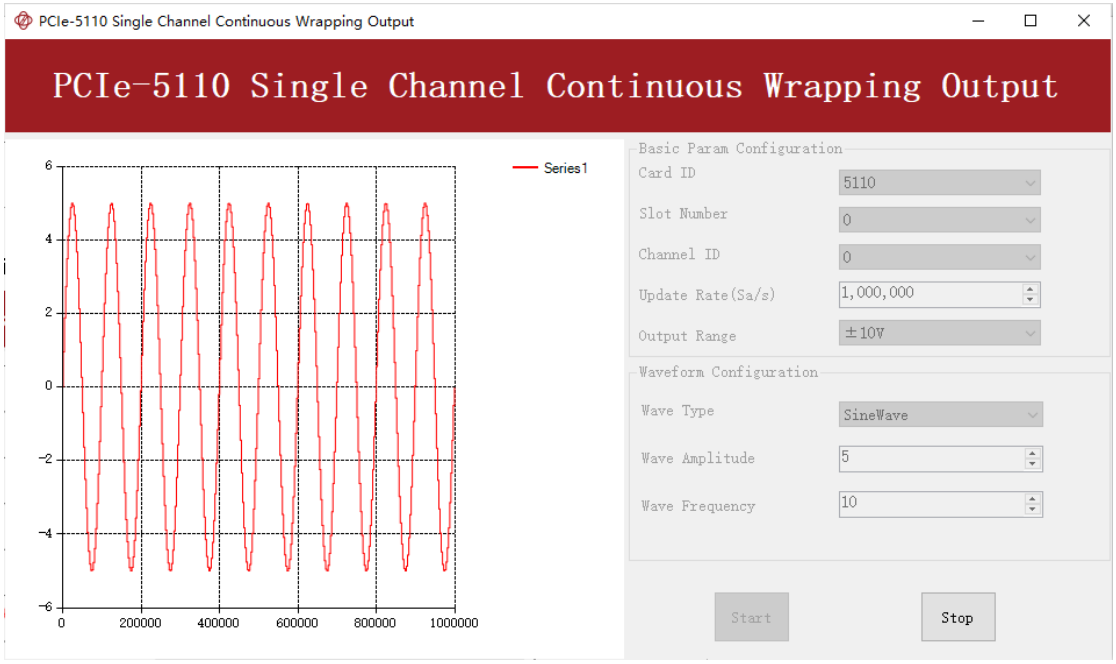


Figure 46 AO Continuous Wrapping Signal

- And the received signal is shown below.

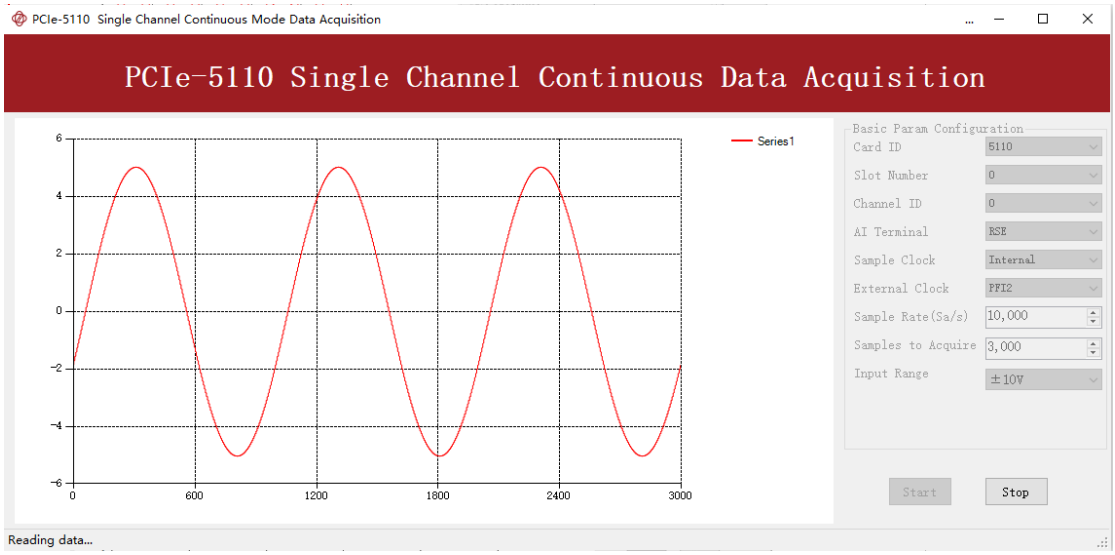


Figure 47 AI Acquisition AO Signal

➢ The analog signal is successfully generated and received by PCIe-5114-H7.

## 8.7. Digital I/O Operations

The PCIe/PXIe-5114-H7 provides powerful programmable digital I/O functions.

### 8.7.1. Static DI/DO

Programmable I/O supports static TTL, 24 digital I/O channels. User can access these I/O information through software polling.

**Learn by Example 8.7.1**

- In this example PCIe-5114-H7 outputs a digital signal by its DO function and reads it back by its DI function.
- Connect Connector0 of PCIe-5114-H7 to the TB-68 terminal block according to 错误!未找到引用源。.
- Connect DIO1 (Pin#35) to DIO3 (Pin#36).PCIe-5114-H7 sends a digital signal through DIO1 and reads the signal back from DIO3.
- Open the software **JYDM-->Test Panel-->DIO.**
- Select **Ch_3** in DI Line, Select **Ch_1** in DO Line. Click **Start** to set DIO3 as a DI and set DIO1 as a DO.

■ Set DIO1 in High-Level positions, The result is shown below.

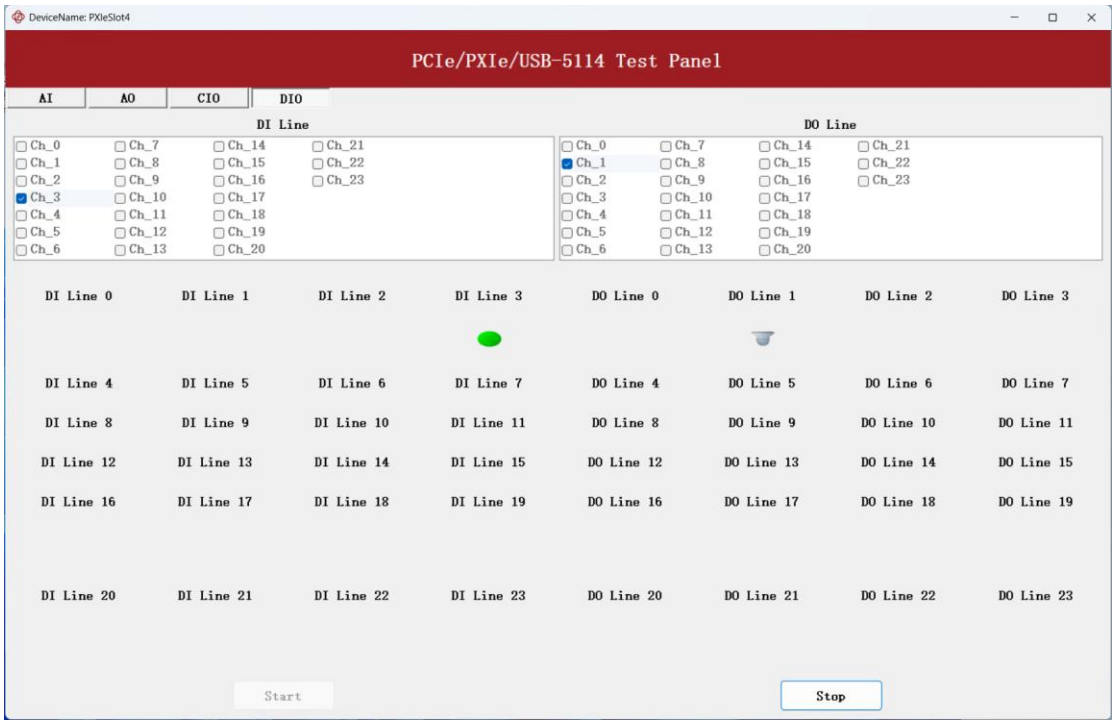


Figure 48 Single Digital Output/ Input

➢ The result matches the high and low levels set before.

### 8.7.2. Dynamic DI/DO

The PCIe/PXIe-5114-H7 supports both dynamic DI/DO operation with a maximum sample rate (update rate) of up to 10MHz. User can acquire or output digital waveforms in this way.

**Learn by Example 8.7.2**

■ In this example PCIe-5114-H7 outputs a squarewave by its DO function and reads it back by its DI function.

■ Connect Connector0 of PCIe-5114-H7.

■ Connect PCIe-5114-H7 Connect DIO1 (Pin#35) to DIO3 (Pin#36).

■ PCIe-5114-H7 sends digital signals through DIO1 and reads them back from DIO3.

- Open **Digital Input-->Winform DI Continuous** and set the numbers as shown. Set **Channel ID** as 3 (DIO3).

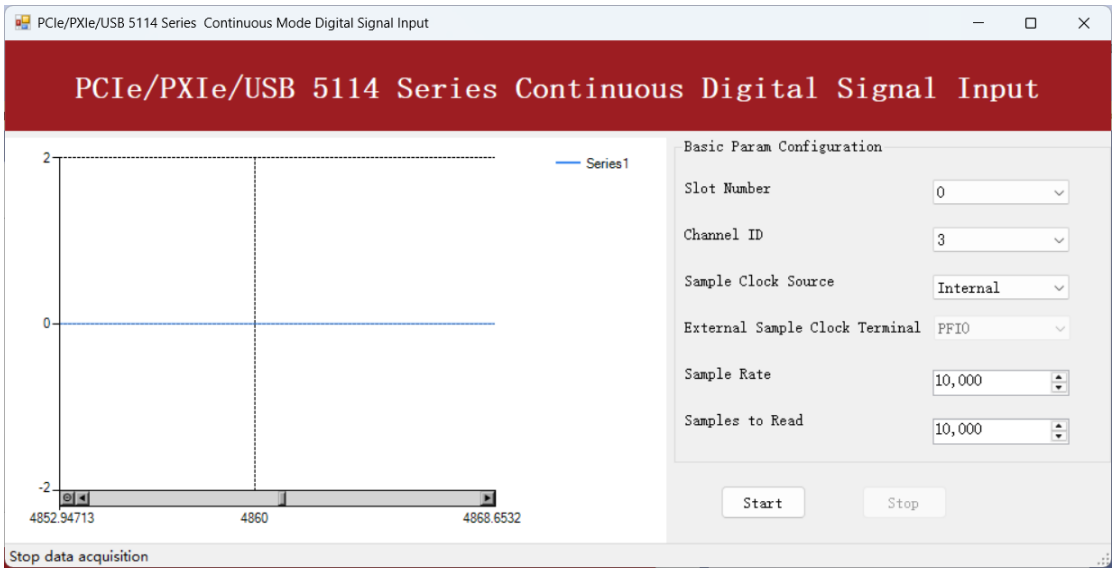


Figure 49 DI Continuous Paraments

- Click **Start** to begin the data acquisition.
- Open **Digital Output--> Winform DO Continuous NoWrapping** and set **Channel ID** as 1 (DIO1).
- Click **Start** to generate the signal. The result is shown below.

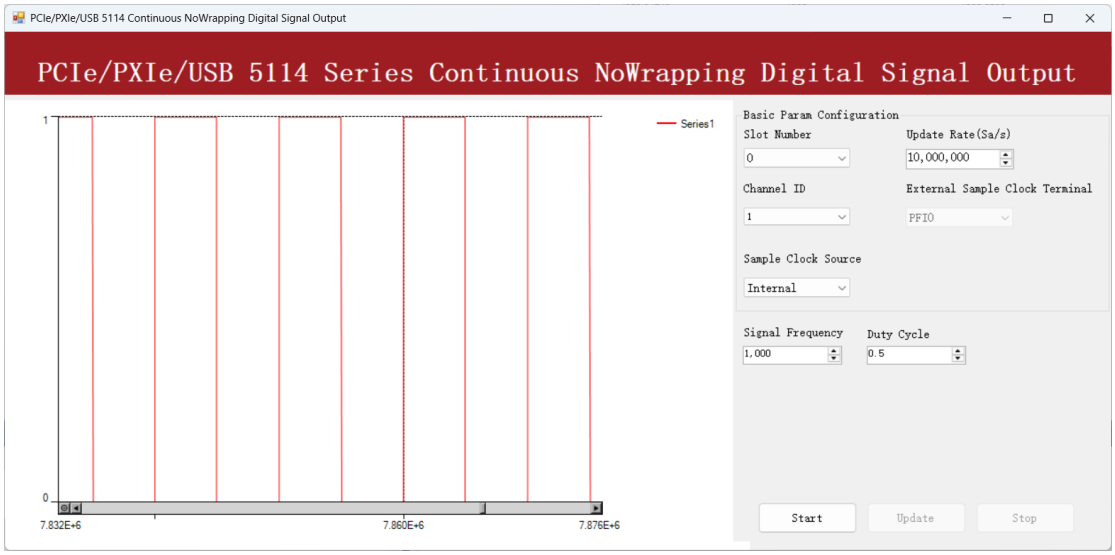


Figure 50 DO ContinuousNoWrapping Output

■ In program **Winform DI Continuous**, you can see the acquired signal.

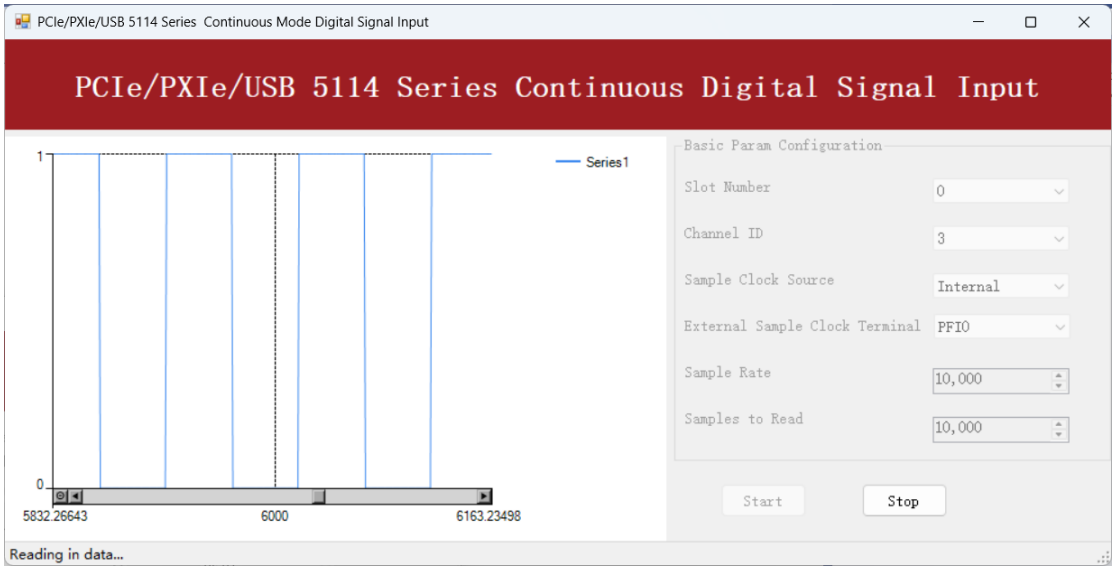


Figure 51 DI Continuous Acquisition

➢ The digital signal is successfully generated and acquired by PCIe-5114-H7.

## 8.8. Counter Input Operations

The PCIe/PXIe-5114-H7 has two identical 32 bits timers/counters.

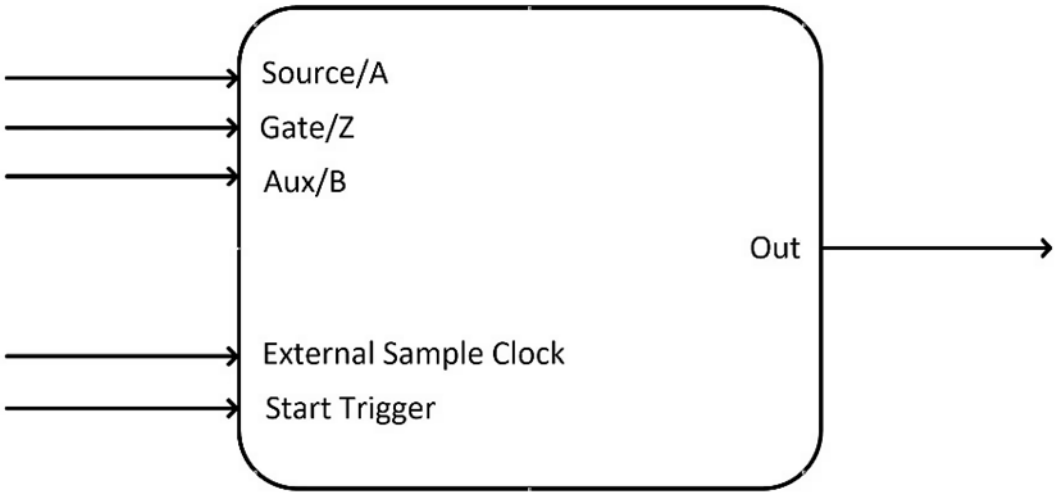


Figure 52 Counter Terminal

Each counter has five input terminals and one output terminal, and these terminals have different functions in different counter input application types, including:

- Edge Counting
- Pulse Measurement
- Frequency Measurement
- Period Measurement
- Two-Edge Separation
- Quadrature Encoder (X1, X2, X4)
- Two-Pulse Encoder

For buffered acquisition, each counter has a separate DDR storage space and requires a sample clock.

For each counter input application type, the measured signal needs to be connected to different terminals, as shown in the following table.

| Measured Signal | Terminal |
|---|---|
| Edge Counting | Source |
| Pulse Measurement | Gate |
| Frequency Measurement | Gate |
| Period Measurement | Gate |
| Two-Edge Separation | Gate、Aux |
| Quadrature Encoder (X1, X2, X4) | A、B、Z |
| Two-Pulse Encoder | A、B |

Figure 53 Counter Signal Wiring Instruction

### 8.8.1. Edge Counting

The counter counts the number of active edges of input signal.

**Timing**

1) Single Mode

The count value is written to the register on each rising edge or falling edge of the signal to measure as shown in Figure 54.

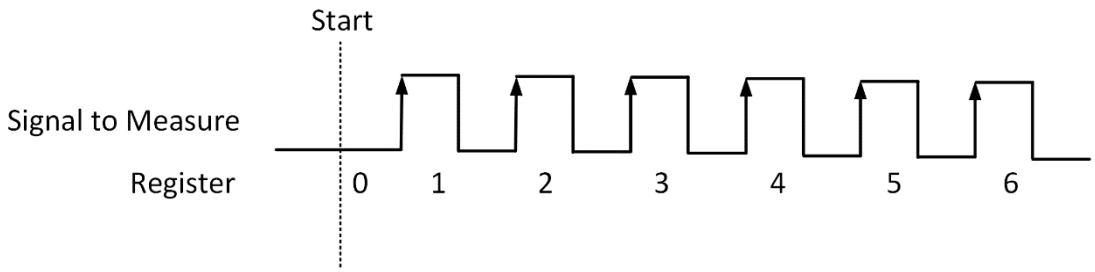


Figure 54 Simple Edge Counting in Single Mode

2) Finite/Continuous Mode with Internal Sample Clock

The count value is stored into the buffer on each rising edge or falling edge of the sample clock as shown in Figure 55.

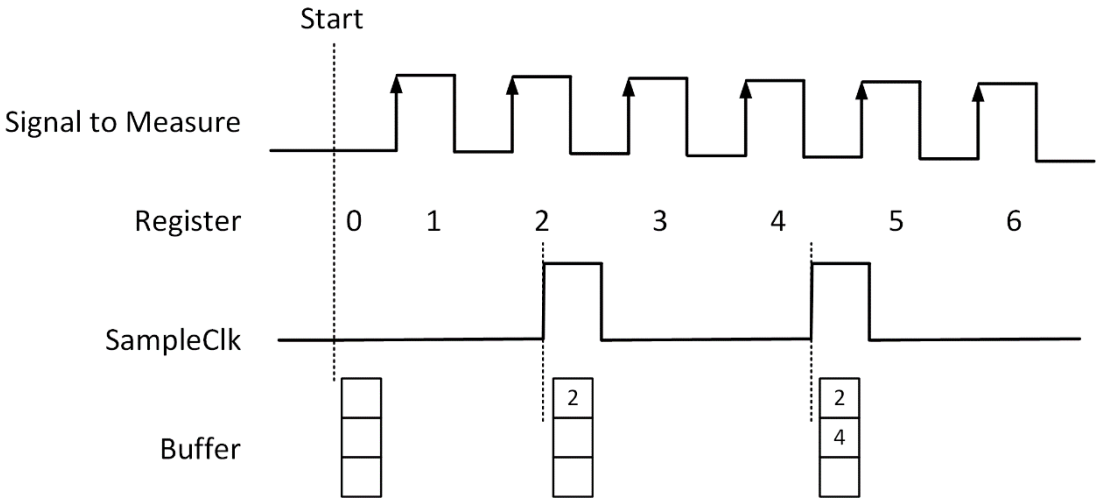


Figure 55 Buffered Edge Counting with Internal Sample Clock

3) Finite/Continuous Mode with Implicit Sample Clock

The count value is stored into the buffer on each rising edge or falling edge of the signal to measure as shown in Figure 56.

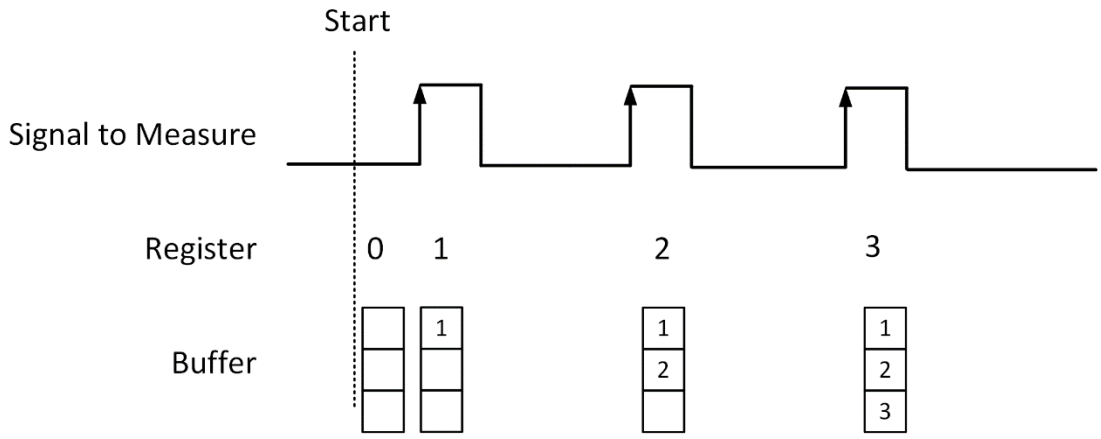


Figure 56 Simple Edge Counting with Implicit SampleClk

**Counting Direction**

User can control the counting direction through software configuration or by an input signal with Gate terminal. When using an input signal to control the counting direction, the counter counts up when the signal is high and counts down when the signal is low as shown in Figure 57.

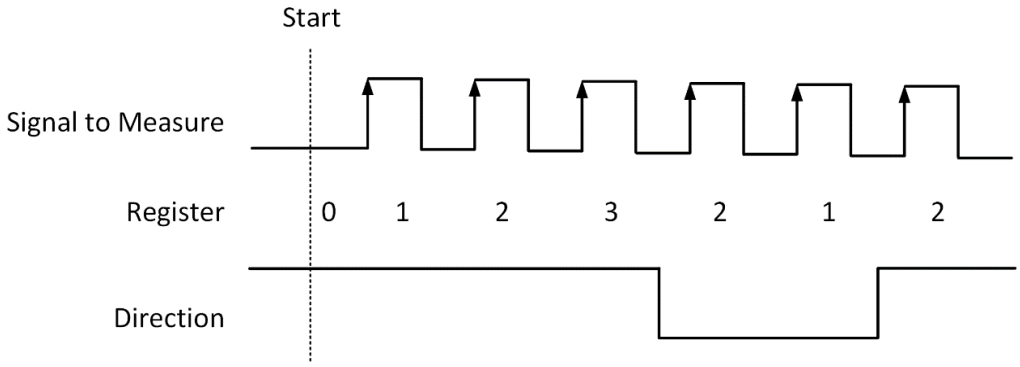


Figure 57 Count Direction

**Learn by Examples8.8.1**

- Connect the signal source's positive terminal of a signal source to PCIe-5114-H7 counter0's edge counting source (CTR0_Source/A, Pin#1), negative terminal to the ground (DGND, Pin#9) as shown in Figure 3 and 错误!未找到引用源。. (CTR0_Source, DGND) consists of an edge counting counter input and they share the same ground.
- Set a squarewave signal (f=1Hz, Vpp=5V).

## Single Mode

- Open **Counter Input-->Winform CI Single EdgeCounting**, set the following numbers as shown:

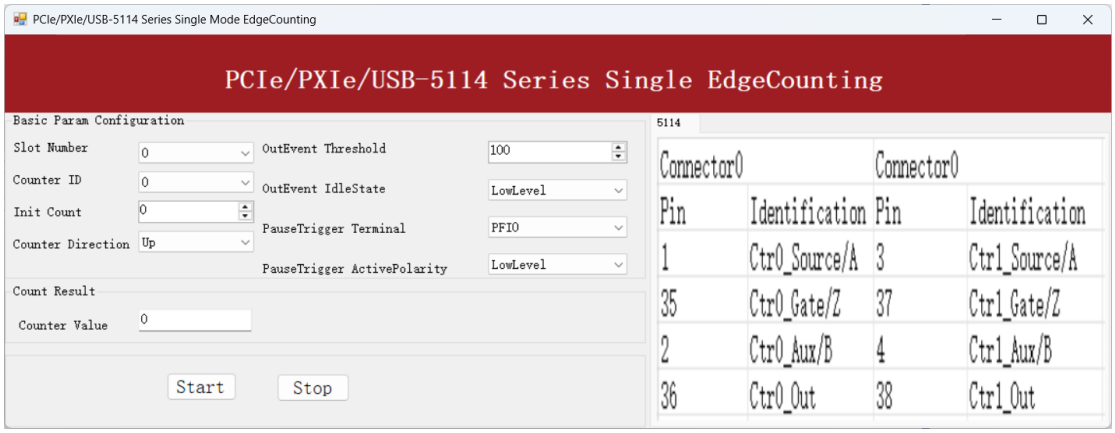


Figure 58 EdgeCounting For Single Mode

- Counter Direction is set by **Counter Direction**.
- The table in the sample program is a connection diagram for your convenience.

  - The *rising edge counter* works when **Start** is clicked.
  - The result is shown by **Counter Value**. In this example the **Counter Value** increases by 1 every second for a 1Hz sinewave.

## Finite/Continuous Mode

- Change the squarewave frequency to 50 Hz.
- Open **Counter Input-->Winform CI Finite/Continuous EdgeCounting**, set the following numbers as shown:

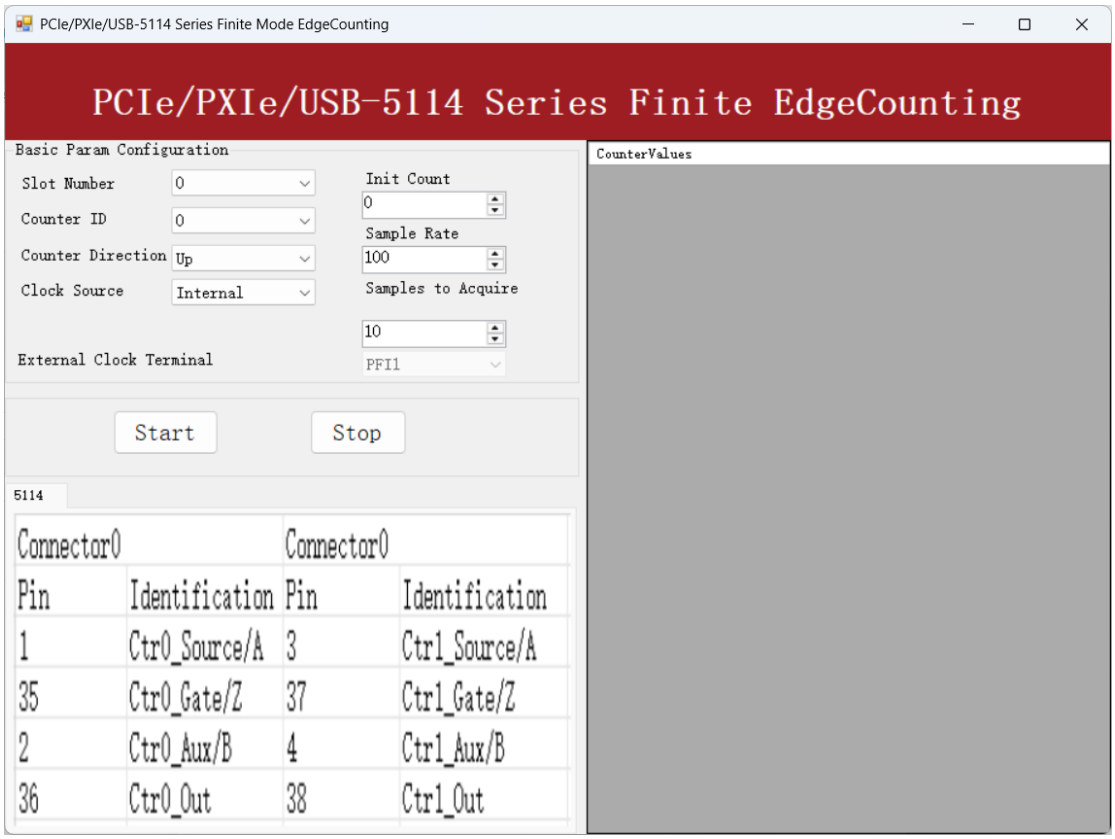


Figure 59 EdgeCounting For Finite Mode

- The table in the sample program is a connection diagram for your convenience.
- Counter Direction is set by **Counter Direction**.
- There are two clock sources in PCIe-5114-H7 *Internal* and *Implicit*: This example uses **Internal** mode set by **Clock Source**.

  - Click **Start** to start counting by rising edge. The result is shown below:

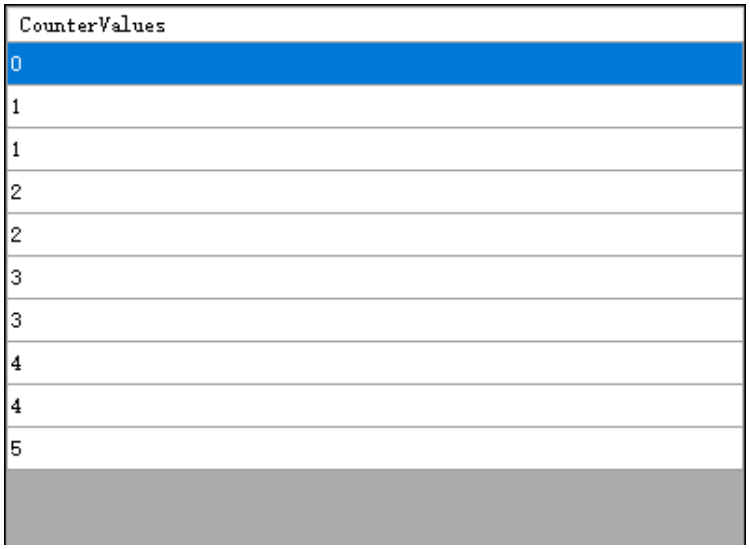


Figure 60 Counter Values For Internal Clock

- The numbers are stored in a buffer **CounterValues**.

- Change the **Clock Source** to **Implicit**:

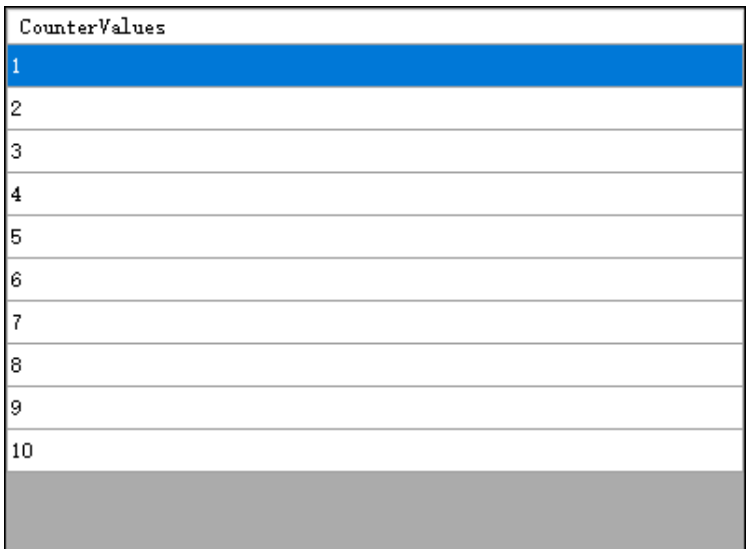


Figure 61 Counter Values For Implicit Clock

- The numbers are stored in a buffer **CounterValues.**
- The counter values are different as before because of the change from **Clock Source**.

### 8.8.2. Pulse Measurement

The counter measures the high-level and low-level duration of signal.

**Timing**

1) Single Mode

The count value of the duration of the high-level or low-level is written to the register on each rising or falling edge of the pulse to measure, as shown in Figure 62.

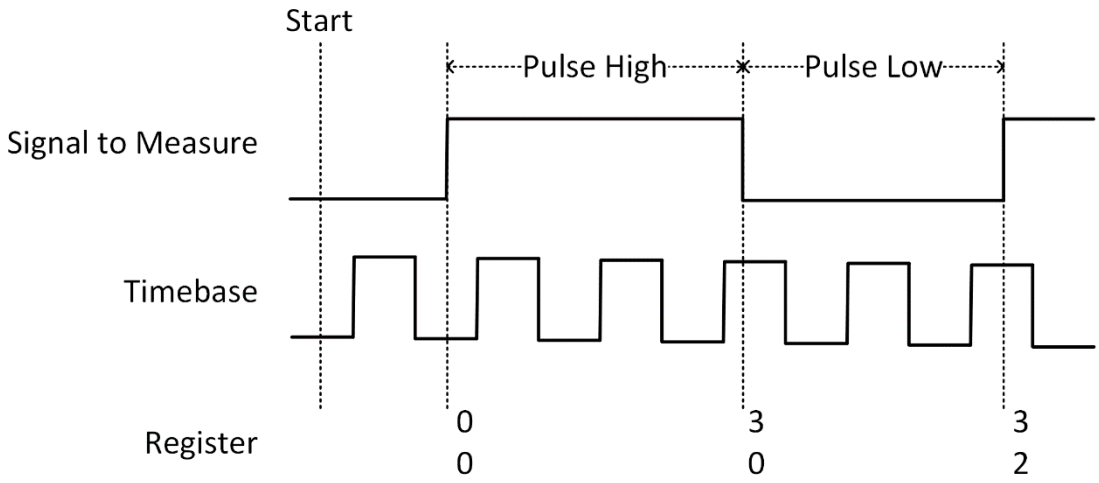


Figure 62 Pulse Measurement in Single Mode

2) Finite/Continuous Mode with Internal Sample Clock

The count value of the duration of the high or low level is stored into the buffer on each rising or falling edge of the sample clock, as shown in Figure 63.

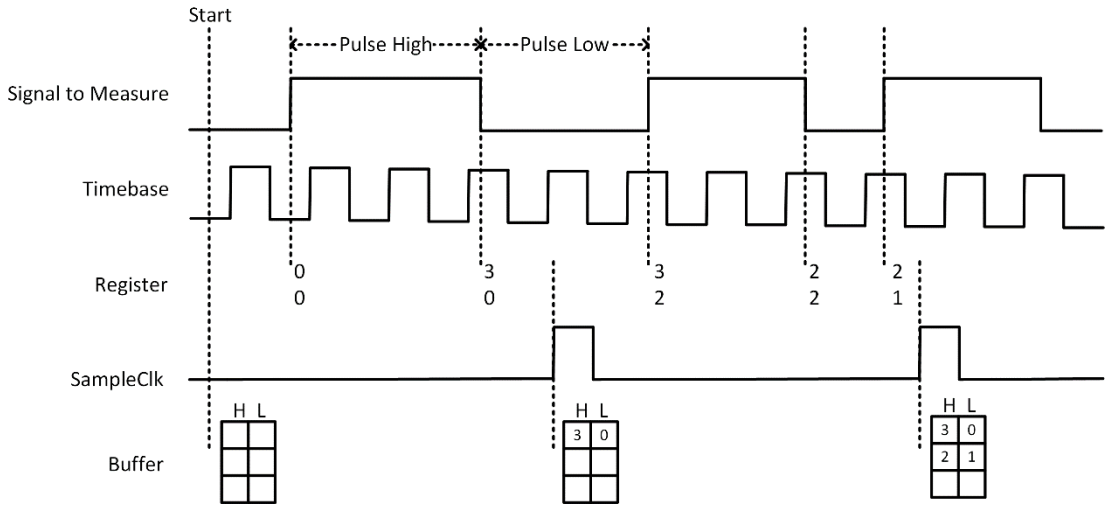


Figure 63 Pulse Measurement with Internal SampleClk

3) Finite/Continuous Mode with Implicit Sample Clock

The count value of the duration of the high-level or low-level is stored into the buffer on each rising or falling edge of the pulse to measure, as shown in Figure 64.

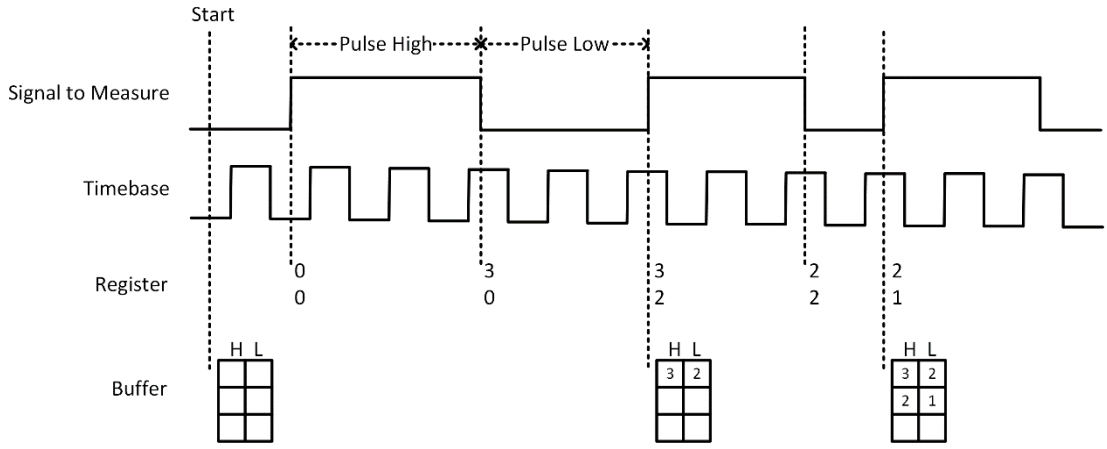


Figure 64 Pulse Measurement with Implicit SampleClk

**Learn by Examples 8.8.2**

- Connect the signal source's positive terminal to PCIe-5114-H7 counter0's pulse measure source (CTR0_Gate/Z, Pin#35), negative terminal to the ground (DGND, Pin#9) as shown in Figure 3 and 错误!未找到引用源。. (CTR0_Gate/Z, DGND) consists of a pulse measure counter input and they share the same ground.

- Set a squarewave signal (f=1Hz, Duty Cycle=50%, Vpp=5V).

**Single Mode**

- Open **Counter Input-->Winform CI Single PulseMeasure**, set the following numbers as shown:

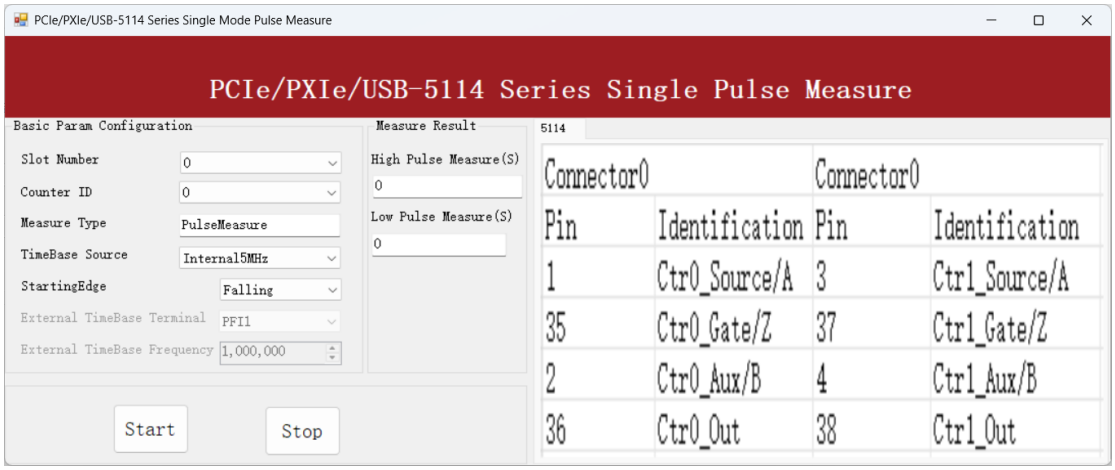


Figure 65 Pulse Measure For Single Mode

➢ The table in the sample program is a connection diagram for your convenience.

- Click **Start** to start measuring the pulses. The result is shown by **High Pulse Measure(S)** and **Low Pulse Measure(S)**:

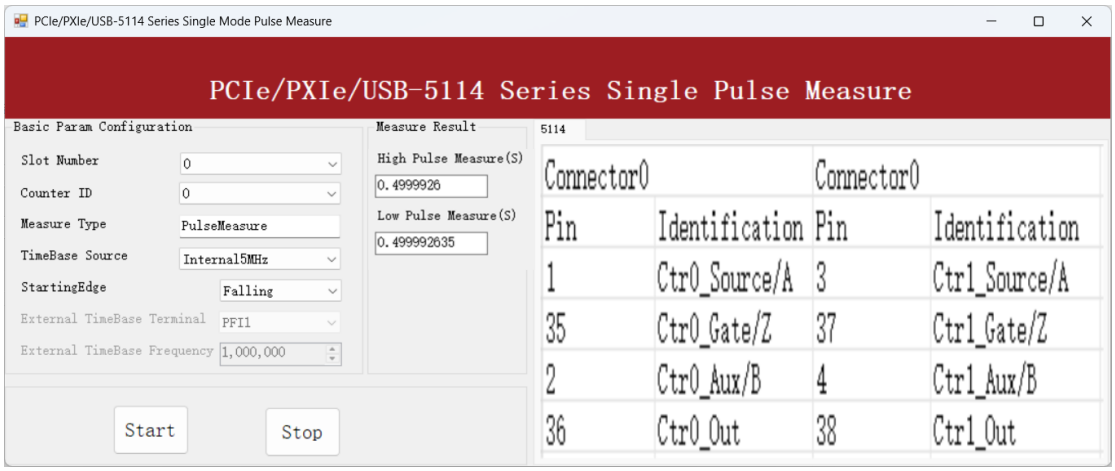


Figure 66 Pulse Measure Value For Single Mode

➢ The numbers show the duration of **High/Low Pulse** in one signal period and match the duty cycle set before.

**Finite/Continuous Mode**

- Change the frequency of Squarewave to 50 Hz.
- Open **Counter Input-->Winform CI Finite/Continuous PulseMeasure**, set the following numbers as shown:

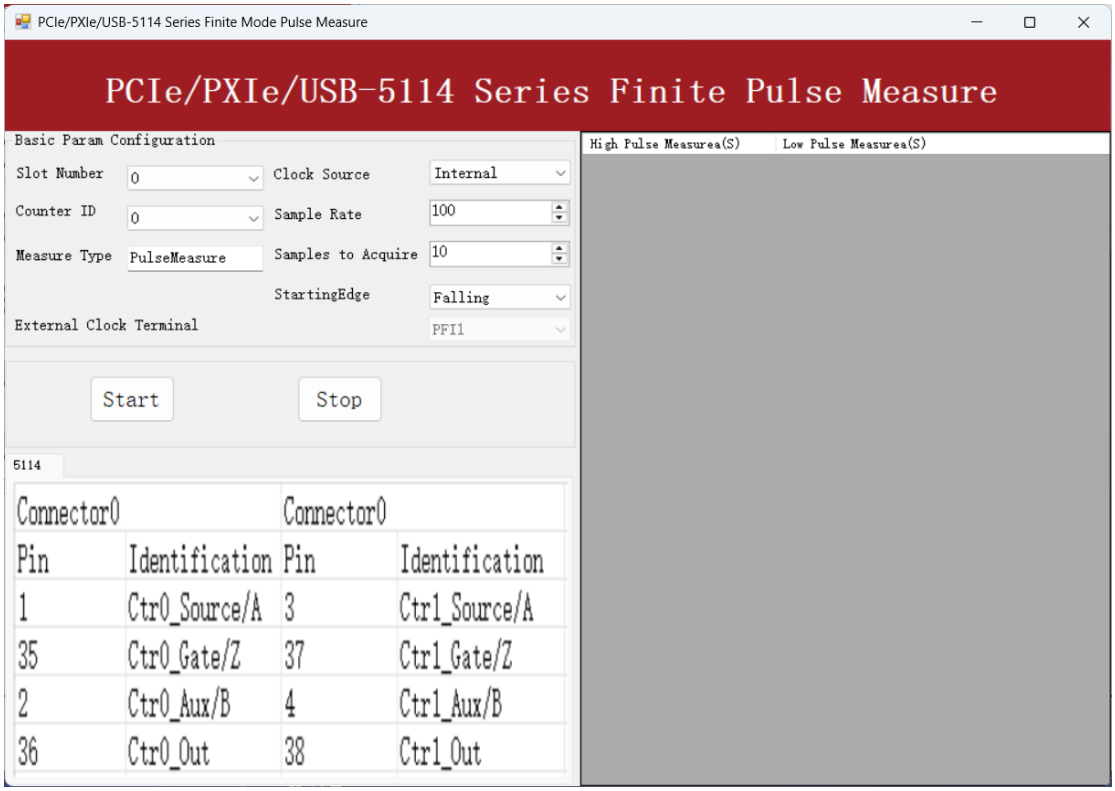


Figure 67 Pulse Measure For Finite Mode

➢ The table in the sample program is a connection diagram for your convenience.

- Click Start to begin the finite/continuous pulse measurement. The result is shown below:

| High Pulse Measurea(S) | Low Pulse Measurea(S) |
|---|---|
| 0 | 0 |
| 0.009999835 | 0 |
| 0.009999835 | 0.009999865 |
| 0.009999835 | 0.009999865 |
| 0.009999835 | 0.00999986 |
| 0.009999835 | 0.00999986 |
| 0.009999835 | 0.009999865 |
| 0.009999835 | 0.009999865 |
| 0.009999835 | 0.009999865 |
| 0.009999835 | 0.009999865 |

Figure 68 Pulse Measure Values For Finite Mode

- The numbers show the duration of **High/Low Pulse** in one signal period and match the duty cycle set before.

- Please refer to **Learn by Examples8.8.1 Finite/Continuous** Mode about the difference between Internal and Implicit.

### 8.8.3. Frequency Measurement

The counter measures the frequency of signal to measure.

**Timing**

1) Single Mode

Frequency Measurement without sample clock is actually using Pulse Width Measurement internally, refer to chapter 8.8.2 for more information.

Every time the user reads the data, driver will automatically calculate the frequency ($f_x$) according to the HighTick ($tick_h$), LowTick ($tick_l$) values and known frequency of the timebase ($f_{base}$) according to the formula 1 and return the result to the user.

$$f_x = f_{base} \times \frac{1}{tick_h + tick_l}$$

To configure the counter to work in this mode, set JY5114CITask.Mode to CIMode.Single.

2) Finite/Continuous Mode with Internal Sample Clock *(Averaging)*

Between every two rising edges of the sample clock, the counter counts the numbers of full periods ($T1$) of the signal to measure, and the number of rising edges of timebase ($T2$) during those full periods. These two values are stored into the buffer on each rising edge of the sample clock, as shown in Figure 69.

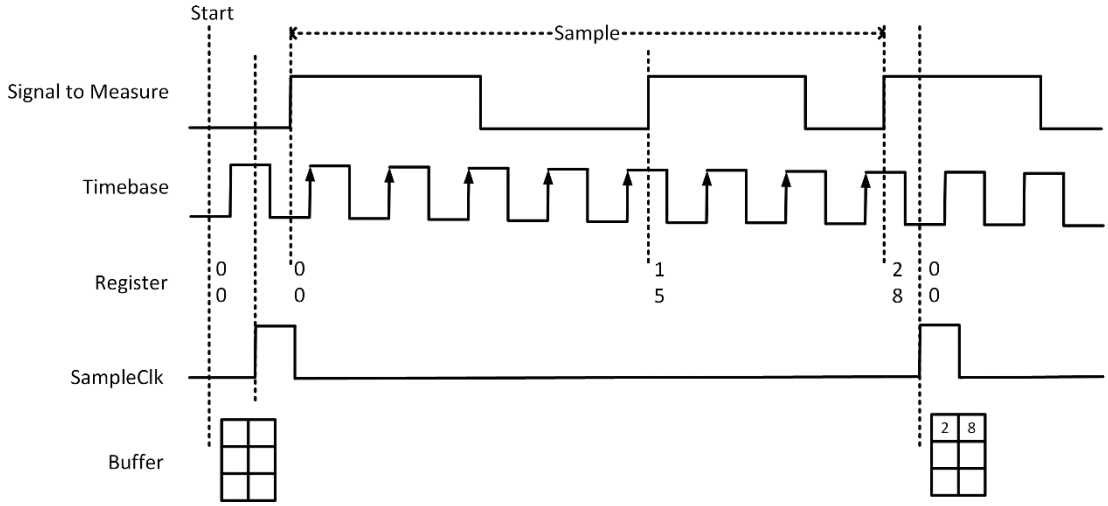


Figure 69 Frequency Measurement with Internal Sample Clock

Every time the user reads the data, driver will automatically calculate the frequency ($f_x$) according to the buffered values and known frequency of the timebase ($f_{base}$) by using following formula and return the result to user.

$$f_x = f_{base} \times \frac{T1}{T2}$$

3) Finite/Continuous Mode with Implicit Sample Clock

Frequency Measurement with implicit sample clock is actually using Pulse Measurement internally. refer to chapter 8.8.2 for more information.

Every time the user reads the data, driver will automatically calculate the frequency ($f_x$) according to the HighTick ($T_h$) and LowTick ($T_l$) values according to the formula 1 and return the result to the user.

$$f_x = \frac{1}{T_h + T_l}$$

**Learn by Examples 8.8.3**

- Connect the signal source's positive terminal to PCIe-5114-H7 counter0's frequency measure source (CTR0_Gate/Z, Pin#35), negative terminal to the ground (DGND, Pin#9) as shown in Figure 3 and 错误!未找到引用源。. (CTR0_Gate/Z, DGND) consists of a frequency measure counter input and they share the same ground.

- Set a squarewave signal (f=50Hz, Duty Cycle=50%, Vpp=5V).

**Single Mode**

- Open **Counter Input-->Winform CI Single Frequency Measure** and click **Start**. The result is shown below by **Frequency Measure (Hz)**:

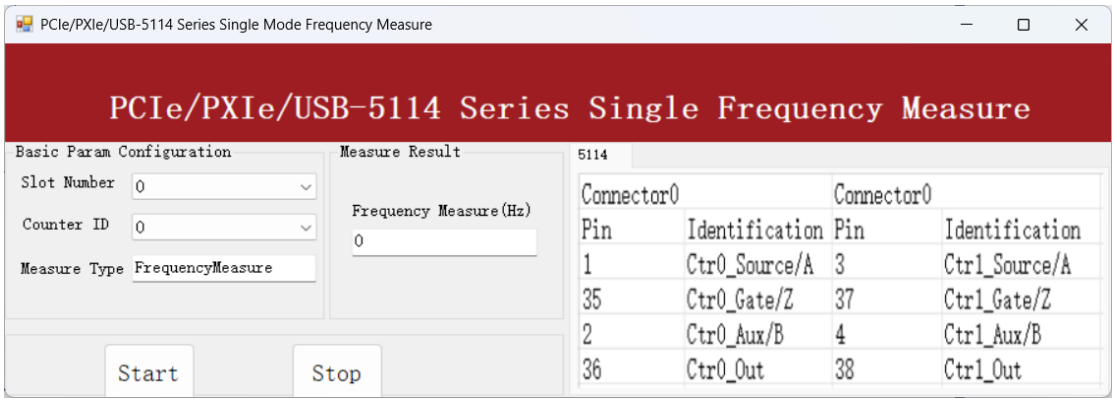


Figure 70 Frequency Measure For Single Mode

- The table in the sample program is a connection diagram for your convenience.
- The result matches the frequency set before.

**Finite/Continuous Mode**

- Open **Counter Input-->Winform CI Finite/Continuous Frequency Measure.**

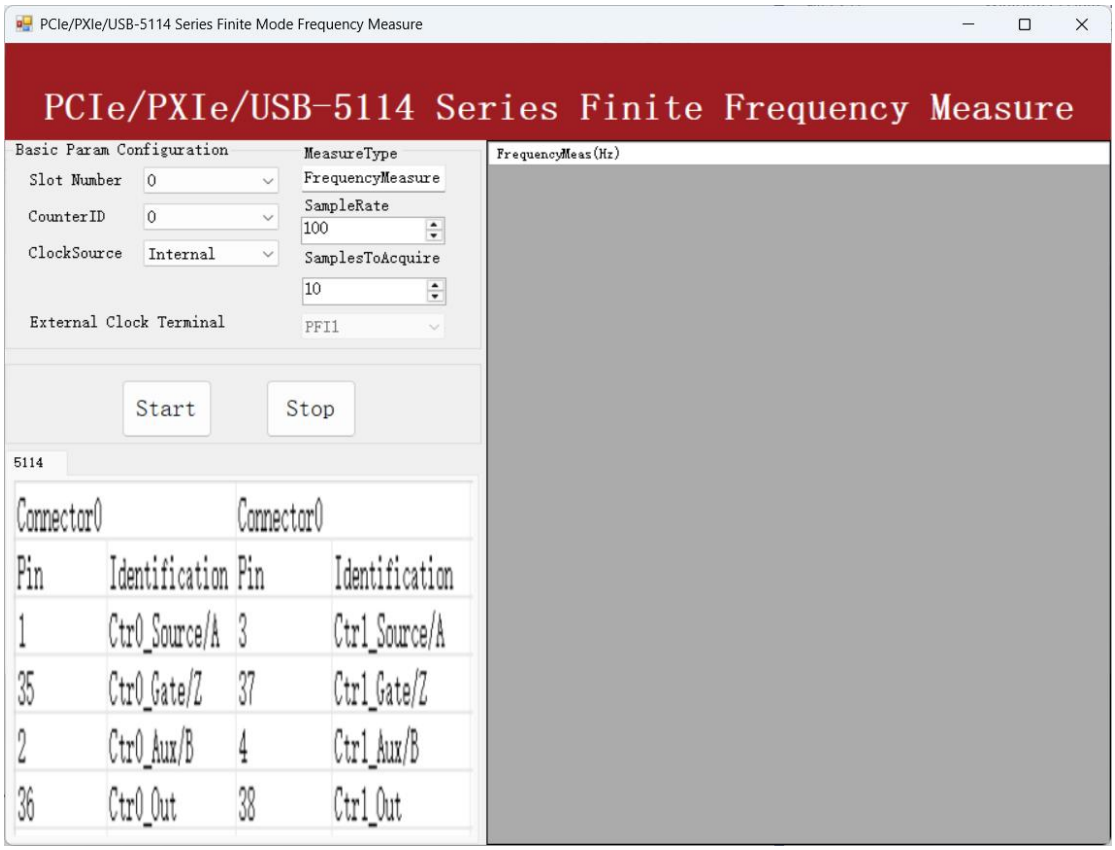


Figure 71 Frequency Measure For Continuous Mode

- The table in the sample program is a connection diagram for your convenience.
- Internal and Implicit Sample Clocks are set by **Clock Source** as before. (Please refer to Finite/Continuous Mode for more information.)

- Click Start and it will show the frequency 50 as set in the signal resource.

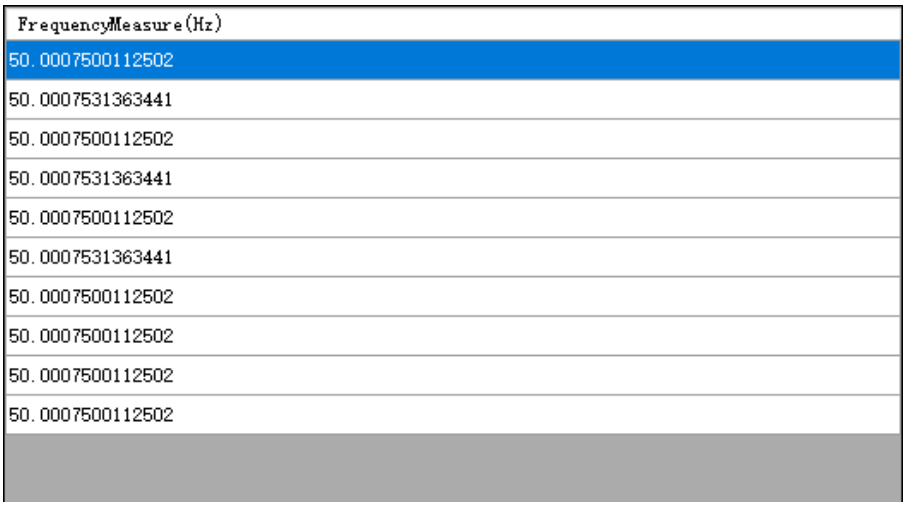


Figure 72 Frequency Measure Values

### 8.8.4. Period Measurement

The counter measures the period of signal to measure. Period Measurements is using Frequency Measurement internally and returns the inverse result of Frequency Measurement. Refer to chapter 8.8.3 for more information.

**Learn by Examples 8.8.4**

- Connect the signal source's positive terminal to PCIe-5114-H7 counter0's period measure source (CTR0_Gate/Z, Pin#35), negative terminal to the ground (DGND, Pin#9) as shown in Figure 3 and 错误!未找到引用源。. (CTR0_Gate/Z, DGND) consists of a period measure counter input and share the same ground.
- Set a squarewave signal (f=200Hz, Duty Cycle=50%, Vpp=5V).

**Single Mode**

- Open **Counter Input-->Winform CI Single Period Measure** and click **Start**. The result is shown below by **Period Measure(S)**:

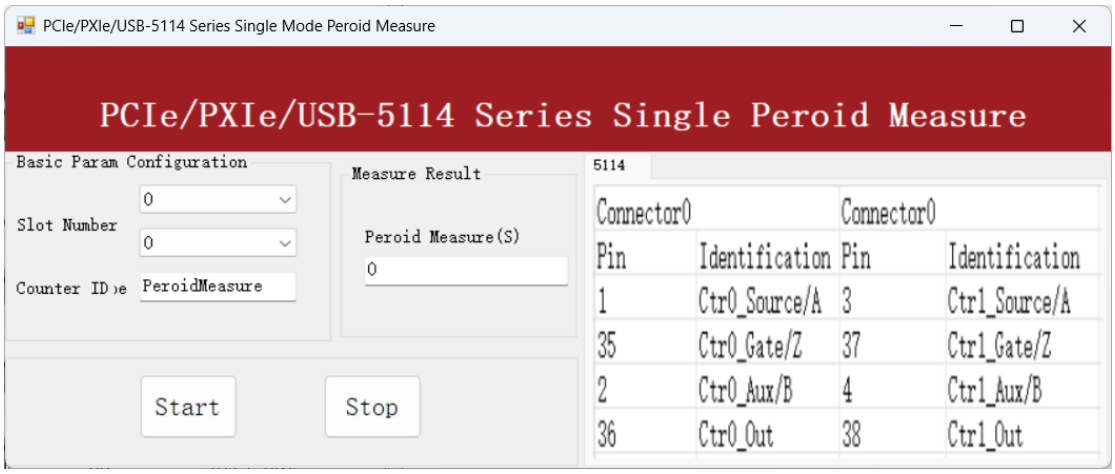


Figure 73 Peroid Measure For Single Mode

- The table in the sample program is a connection diagram for your convenience.
- The result of **Period Measure(S)** shows the correspond to the frequency set before.

**Finite/Continuous Mode**

- Open **Counter Input-->Winform CI Finite/Continuous Period Measure** and click **Start**. The result is shown below by **PeriodMeasure (S).**

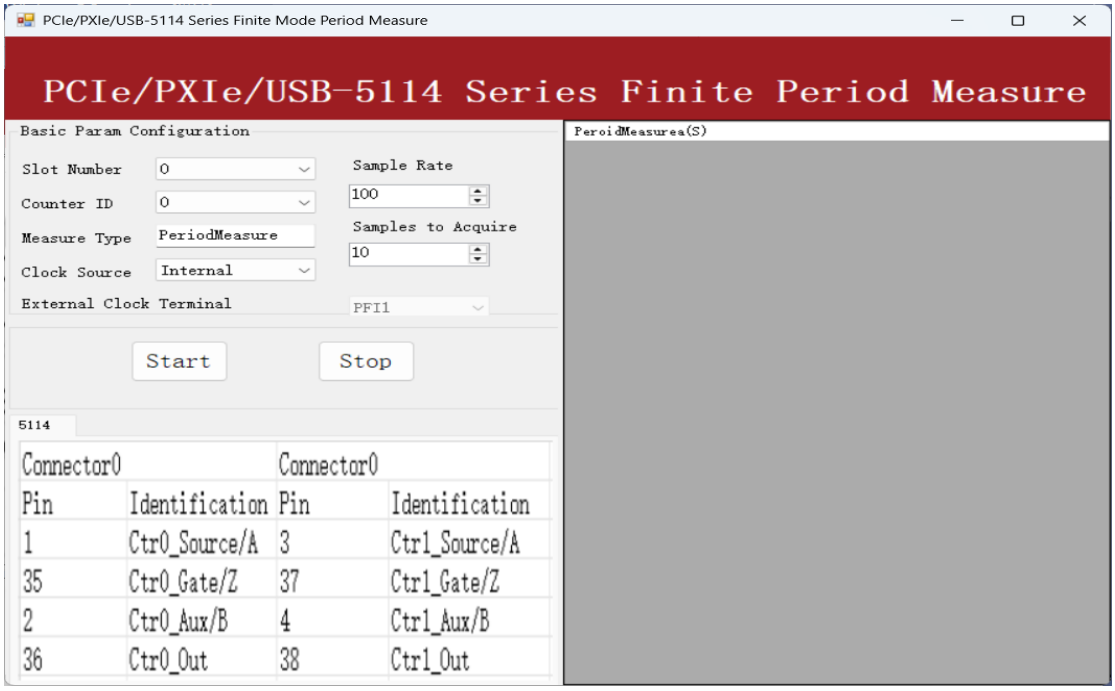


Figure 74 Peroid Measure For Continuous Mode

- The table in the sample program is a connection diagram for your convenience.
- The result of **Period Measure(S)** shows the correspond to the frequency set before.

### 8.8.5. Two-Edge Separation

The counter measures the separation between the rising edges of two signals.

**Timing**

1) Single Mode

The number of rising edges of timebase between the rising edge of the first signal and the rising edge of the second signal is written to the register on each rising edge of the second signal.

The number of rising edges of timebase between previous rising edge of the second signal and current rising edge of the first signal is written to the register on each rising edge of the first signal as shown in Figure 75.

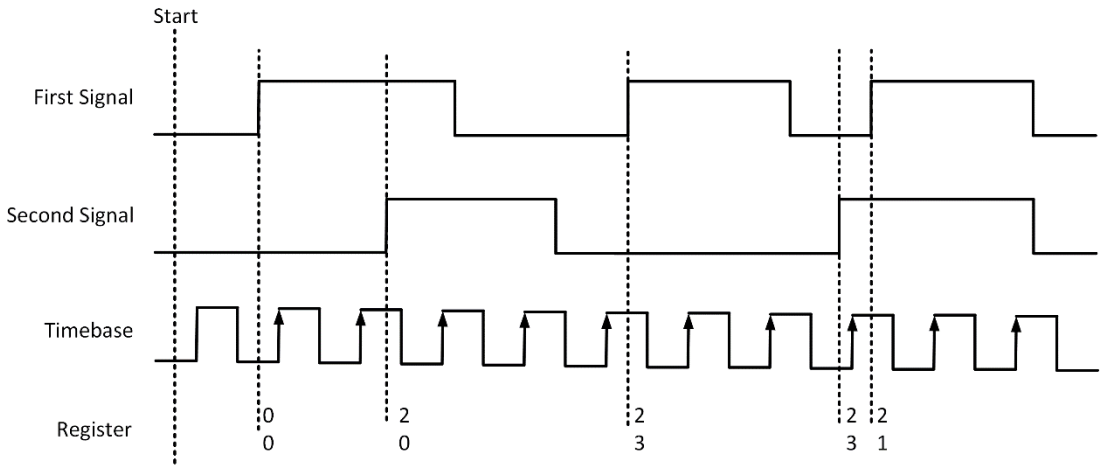


Figure 75 Two-Edge Separation in Single Mode

2) Finite/Continuous Mode with Internal Sample Clock:

The count values of rising edges of timebase between first signal and second signal are stored into buffer on each rising edge of the sample clock, as shown in Figure 76.

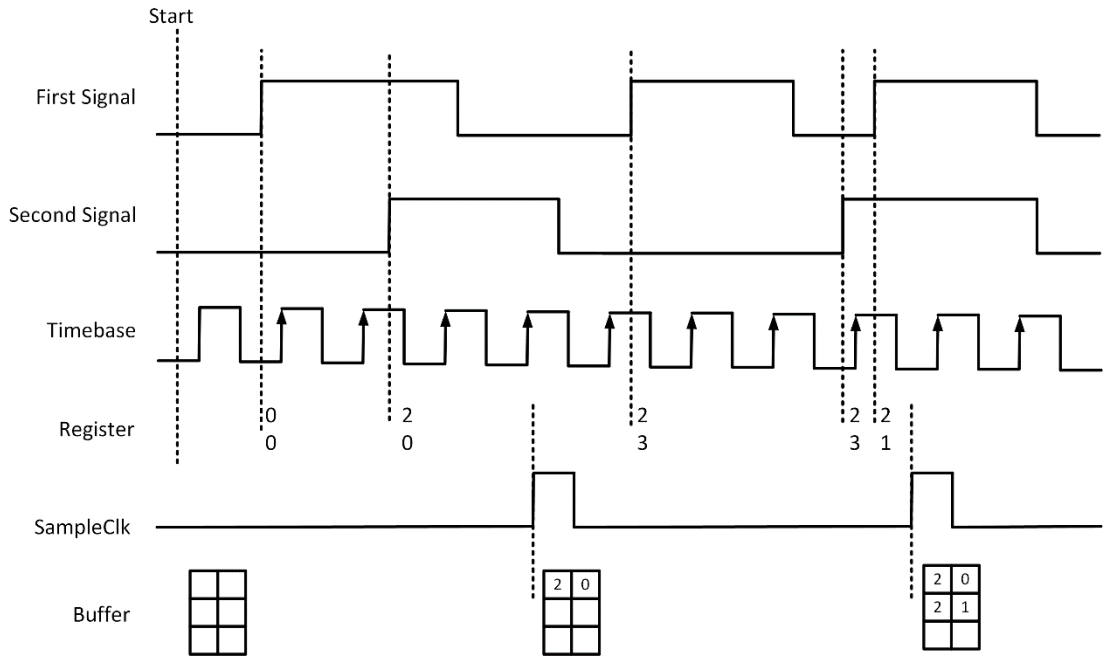


Figure 76 Two-Edge Separation with Internal Sample Clock

3) Finite/Continuous Mode with Implicit Sample Clock

The count values of rising edges of timebase between first signal and second signal are stored into buffer on each rising edge of the first signal, as shown in Figure 77.

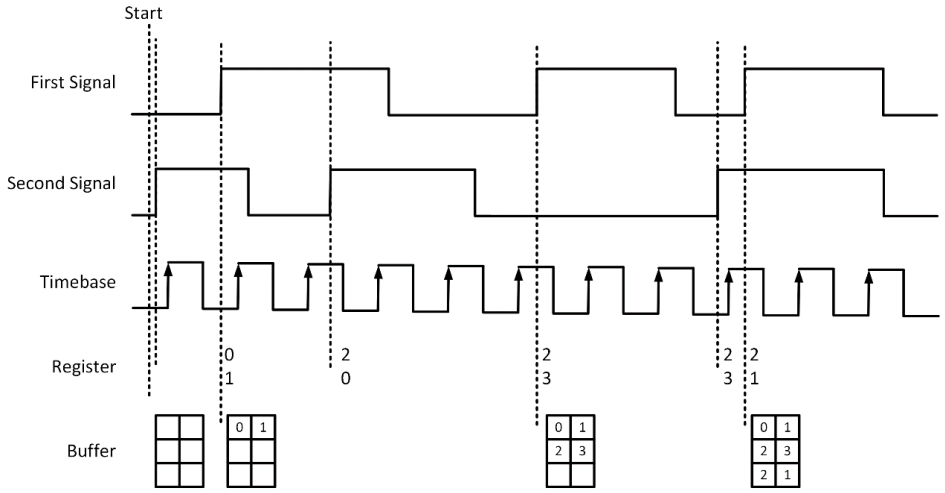


Figure 77 Two-Edge Separation with Implicit Sample Clock

**Learn by Examples 8.8.5**

- Connect the signal source's two positive terminals to PCIe-5114-H7 first signal input (squarewave, CTR0_Gate/Z, Pin #35) and second signal input (squarewave, CTR0_AUX/B, Pin#2), two negative terminals to the ground (DGND, Pin#9) and (DGND, Pin#43) as shown in Figure 3 and 错误!未找到引用源。.
- Set a squarewave signal (f=1Hz, Phase=0°) and a squarewave signal (f=1Hz, Phase=135°).

**Single Mode**

- Open **Counter Input-->Winform CI Single TwoEdgeSeparation Measure** and click **Start**. The result is shown below by **First to Second(S)** and **Second to First(S)**, which represent the time difference between the rising edges of the two signals:

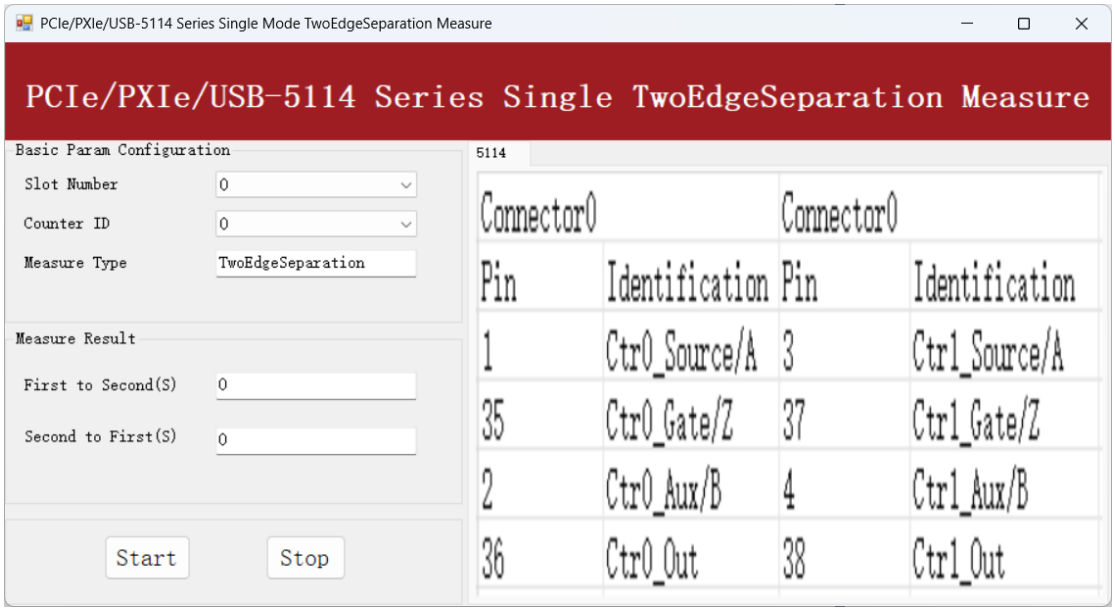


Figure 78 Two-EdgeSeparation Measure For Single Mode

- The table in the sample program is a connection diagram for your convenience.
- Due to the phase-difference between First Signal and Second Signal, **First to Second** and **Second to First** are different and summarize as 1.

**Finite/Continuous Mode**

- Open **Counter Input-->Winform CI Finite/Continuous TwoEdge Separation Measure** and click **Start**. The result is shown below by **First to Second(S)** and **Second to First(S)**, which represent the time difference between the rising edges of the two signals:

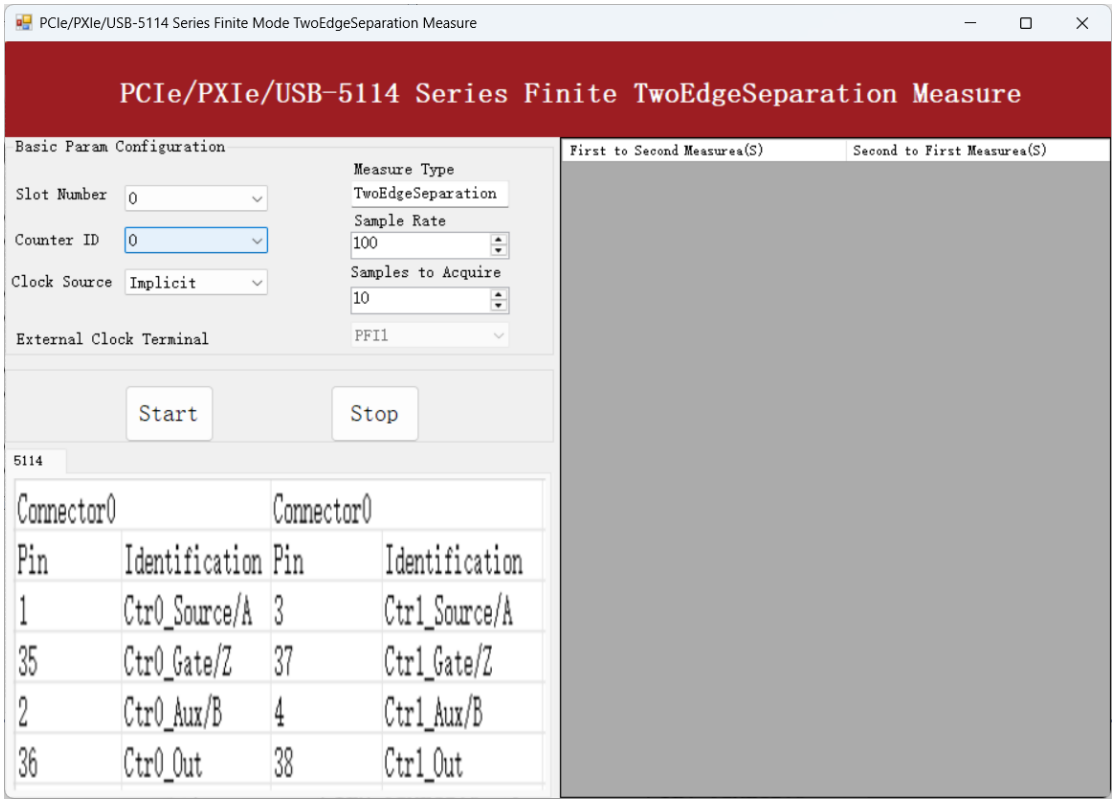


Figure 79 Two-EdgeSeparation Measure For Finite Mode

- The result in this picture is similar to the result in **Single Mode** before.
- The table in the sample program is a connection diagram for your convenience.

### 8.8.6. Quadrature Encoder

The quadrature encoder includes three encoding types: x1, x2, and x4.

Encoding Type

1) x1 Encoding

When A leads B, the count increase occurs on the rising edge of A; when B leads A, the count decrease occurs on the falling edge of A as shown in Figure 80.

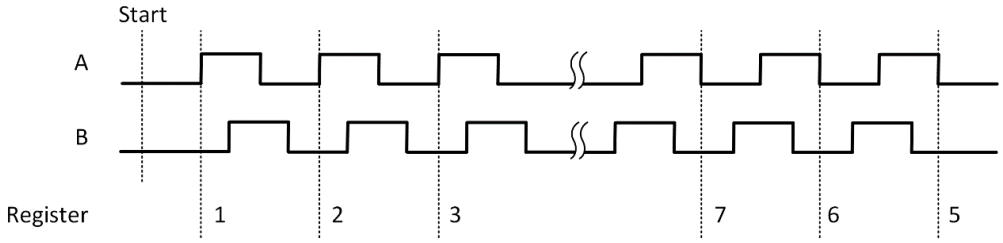


Figure 80 Quadrature Encoder x1 Mode

2) x2 Encoding

When A leads B, the count increase occurs on the rising edge and the falling edge of A; when B leads A, the count reduction occurs on the rising edge and falling edge of A as shown in Figure 81.

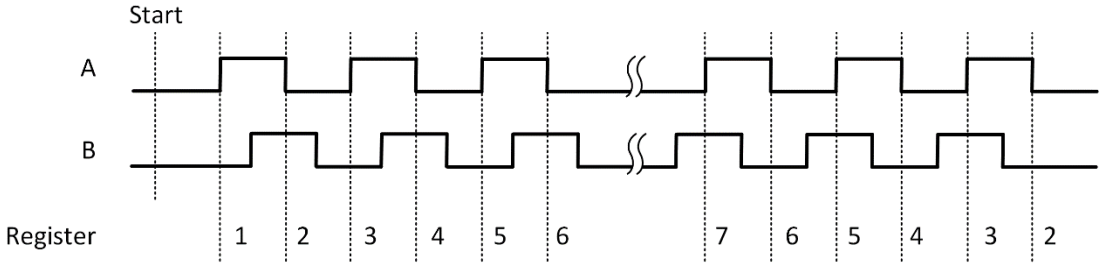


Figure 81 Quadrature Encoder x2 Mode

3) x4 Encoding

When A leads B, the increase of count occurs on the rising and falling edges of A and B. When B leads A, the decrease in count occurs on the rising and falling edges of A and B. As shown in Figure 82.

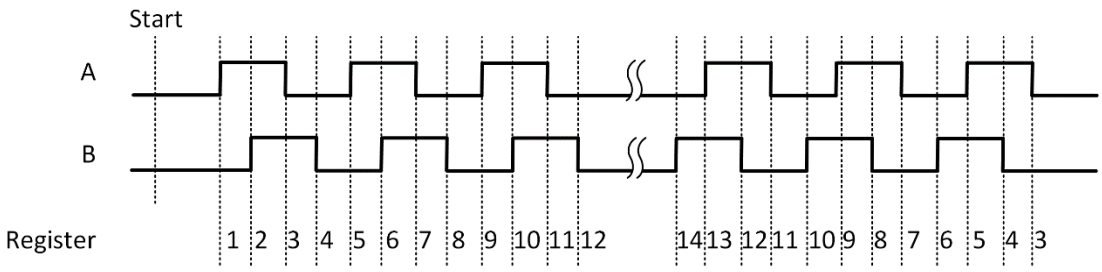


Figure 82 Quadrature Encoder x4 mode

**Channel Z Behavior**

The phase is reloaded when channel Z is high, A and B are low.

**Timing**

Take Encoding x1 mode as an example.

1) Single Mode

The count value is written to the register on each rising edge of the signal A, as shown in Figure 54.

To configure the counter to work in this mode, set JY5114CITask. Mode to CIMode.Single.

2) Finite/Continuous Mode with Internal Sample Clock

The count value is stored into the buffer on each rising edge of the sample clock, as shown in Figure 83.

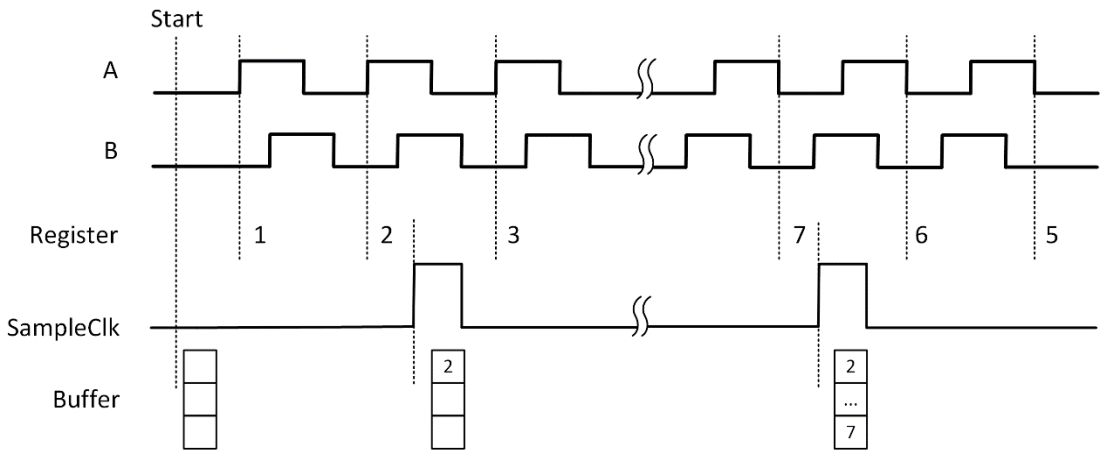


Figure 83 Quadrature Encoder x1 with Sample Clock

3) Finite/Continuous Mode with Implicit Sample Clock

The count value is stored into the buffer every time it changes, as shown in Figure 84.

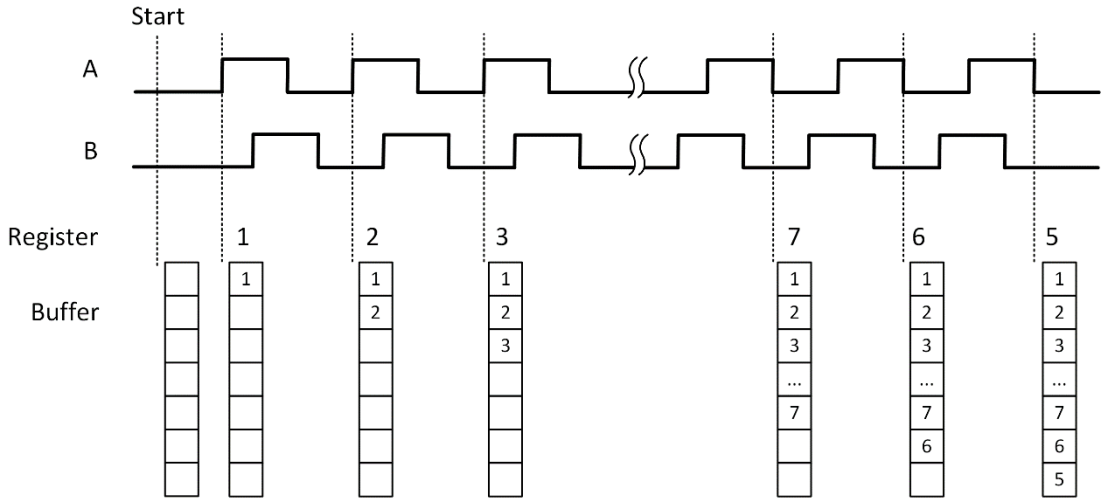


Figure 84 Quadrature Encoder x4 with Implicit Sample Clock

**Learn by Examples 8.8.6**

- Connect the signal source's two positive terminals to PCIe-5114-H7 first signal input (sinewave, CTR0_Source/A, Pin #1) and second signal input (squarewave , CTR0_AUX/B, Pin#2), two negative terminals to the ground (DGND, Pin#9) and (DGND, Pin#43) as shown in Figure 3 and 错误!未找到引用源。.
  (CTR0_Source/A, DGND) consists of the first signal to be measured;
  (CTR0_AUX/B, D_GND) consists of the second signal to be measured.
- Set a sqaurewave signal (f=10Hz, Phase=90°) and a squarewave signal (f=10Hz, Phase=0°).

**Single Mode**

- Open **Counter Input--> Winform CI Single QuadEncoder** and click **Start**. The result is shown below by **CounterValue** according to the counting rules explained in 8.8.6:

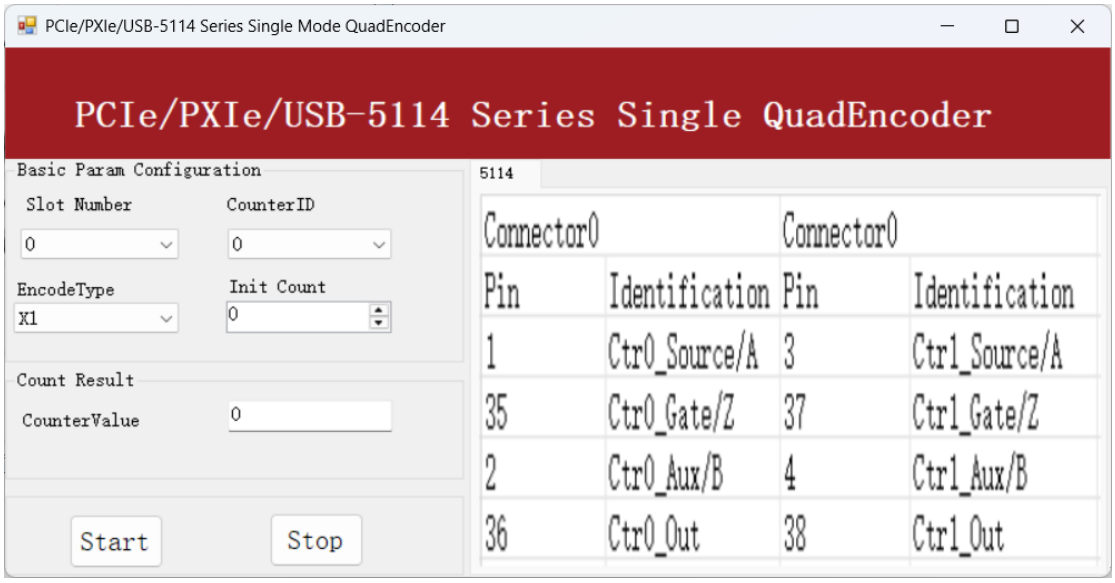


Figure 85 QuadEncoder For Single Mode

- The table in the sample program is a connection diagram for your convenience.
- *Encoding Type* is set by **Encode Type (x1, x2, x4)**.
- When the *encode type* is changed from x1 to x2 and x4, you can see the rising speed of **CounterValue** is twice and four times than x1Mode.

**Continuous Mode**

- Open **Counter Input--> Winform CI Continuous QuadEncoder** and click **Start**. The result is shown below by **CounterValues**.

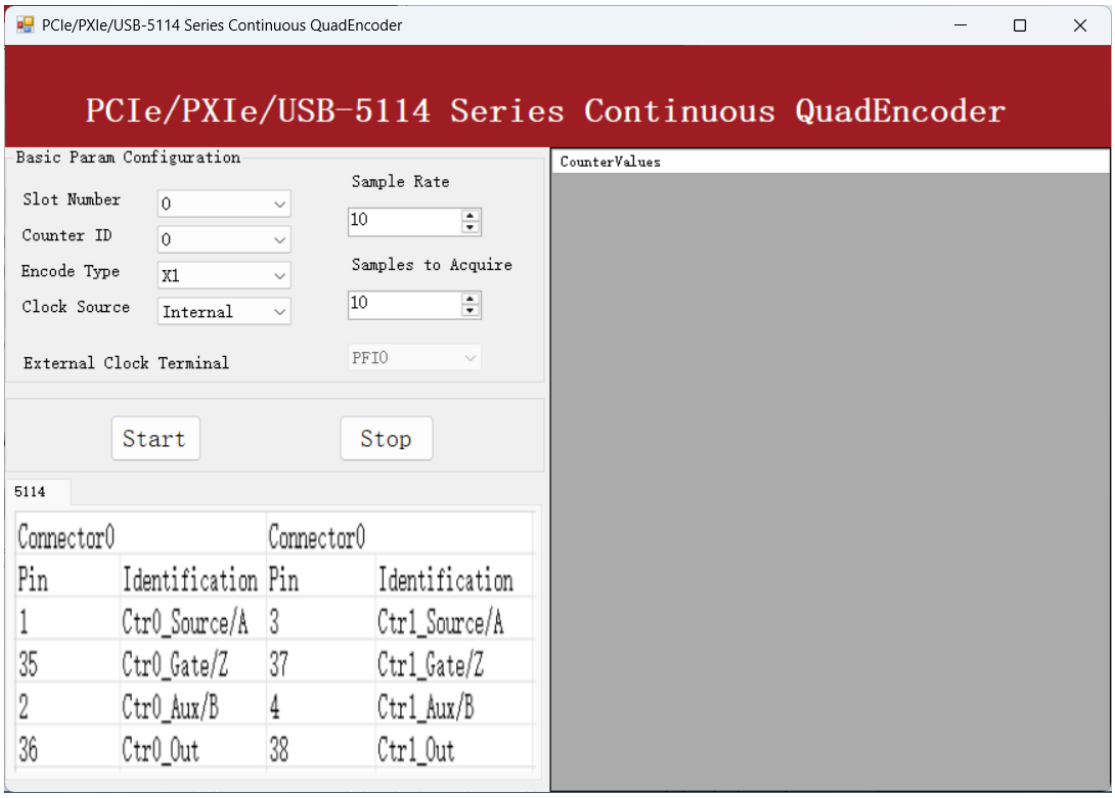


Figure 86 QuadEncoder For Continuous Mode

- The table in the sample program is a connection diagram for your convenience.
- *Encoding Type* is set by **Encode Type (x1, x2, x4)**.
- When the *encode type* is changed from x1 to x2 and x4, you can see the rising speed of **CounterValue** is twice and four times than x1Mode.

### 8.8.7. Two-Pulse Encoder

The count value increases on the rising edge of A and decreases on the rising edge of B.

**Timing**

1) Single Mode

The count value is written to the register on each rising edge of the signal A, and signal B, as shown in Figure 87.

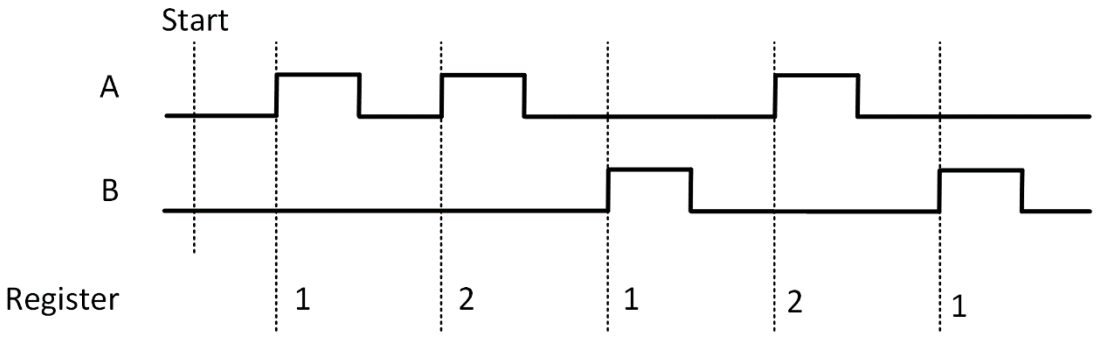


Figure 87 Two-Pulse Encoder in Single Mode

2) Finite/Continuous Mode with Internal Sample Clock

The count value is stored into the buffer on each rising edge of the sample clock, as shown in Figure 88.

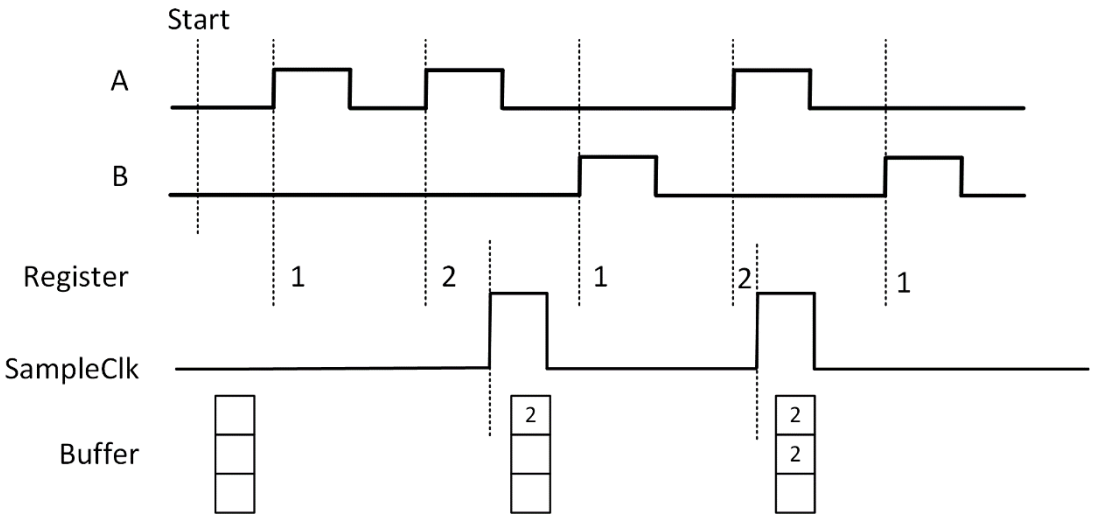


Figure 88 Two-Pulse Encoder with Internal Sample Clock

3) Finite/Continuous Mode with Implicit Sample Clock

The count value is stored into the buffer every time it changed, as shown in Figure 89.

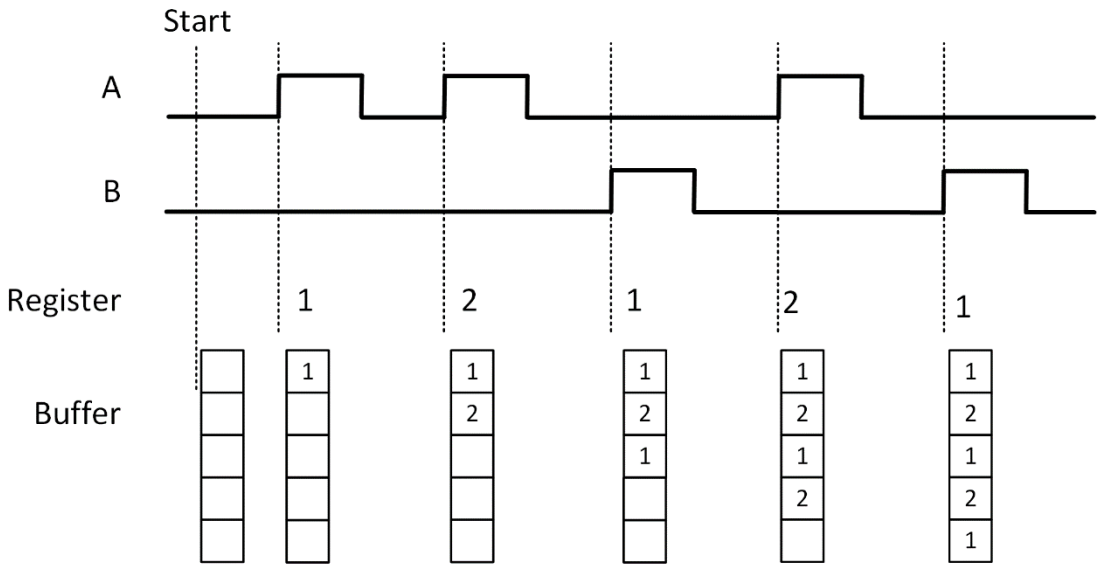


Figure 89 Two-Pulse Encoder with Implicit Sample Clock

**Learn by Examples 8.8.7**

- Connect the signal source's positive terminal to PCIe-5114-H7 signal input (squarewave, CTR0_Source/A, Pin #1)， negative terminal to the ground (DGND, Pin#9)。
- Connect the PCIe-5114-H7 signal input(CTR0_AUX/B, Pin#2)to ground (DGND, Pin#9).
- Set a sqaurewave signal (f=40Hz)

**Single Mode**

- Open **Counter Input-->Winform CI Single Two PulseEncoder** and set the numbers as shown.

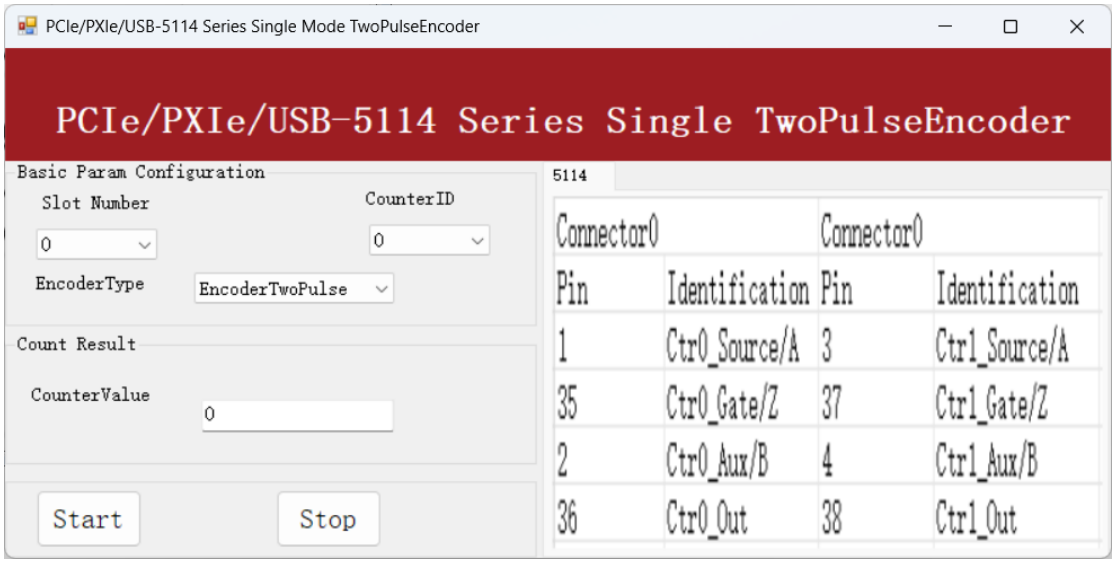


Figure 90 Two-PulseEncoder For Single Mode

➢ The table in the sample program is a connection diagram for your convenience.

■ Click **Start** to start counting. You can see a continuously rising of the **Counter Value,** which follows the counting rules explained in this chapter.

**Finite Mode**

■ Connect the signal source's positive terminal to PCIe-5114-H7 signal input (squarewave, CTR0_AUX/B, Pin#2)， negative terminal to the ground (DGND, Pin#9)。

■ Connect the PCIe-5114-H7 signal input(CTR0_Source/A, Pin #1)to ground (DGND, Pin#9).

■ Set a sqaurewave signal (f=40Hz).

■ Open **Counter Input-->Winform CI Finite Two PulseEncoder** and set the numbers as shown.

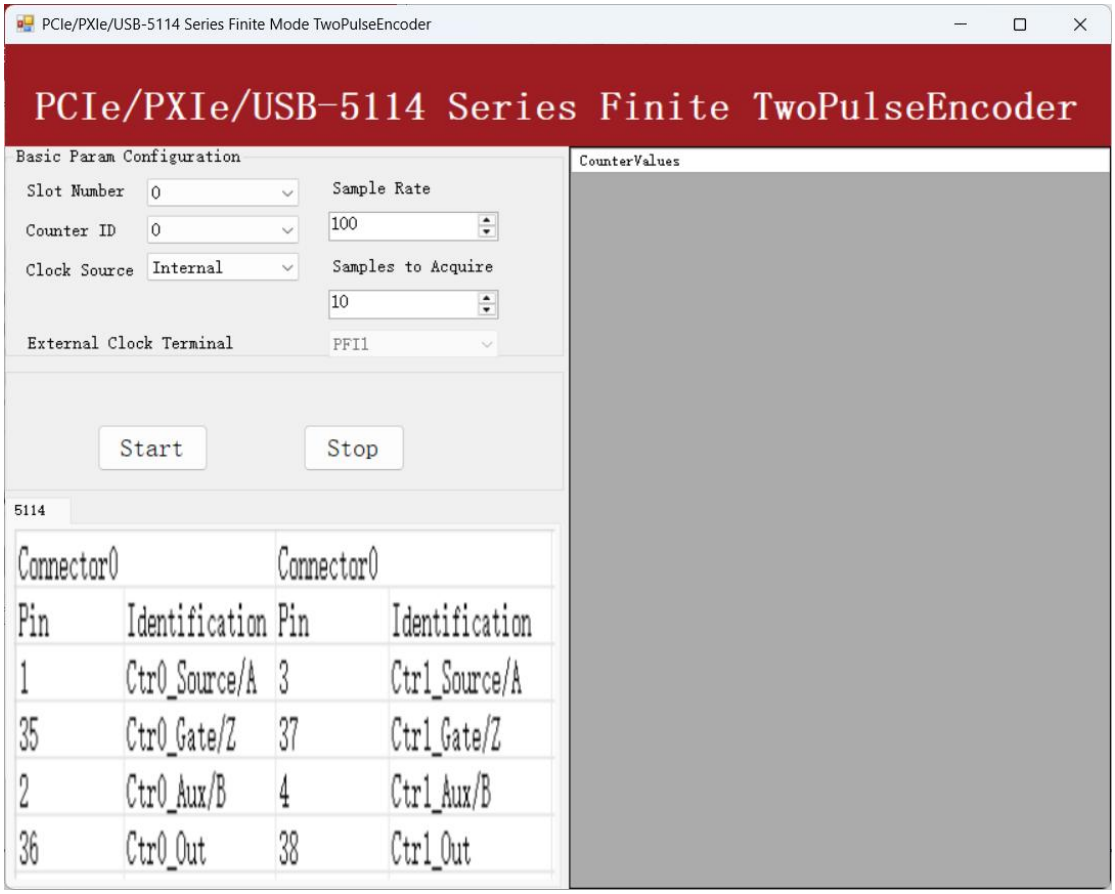


Figure 91 Two-PulseEncoder For Finite Mode

➢ The table in the sample program is a connection diagram for your convenience.

■ Click **Start** to start counting. You can see that the **CounterValue** is decreasing**,** which follows the counting rules explained in this chapter.

**Continuous Mode**

■ Open **Counter Input-->Winform CI Continuous Two PulseEncoder** and set the numbers as shown.

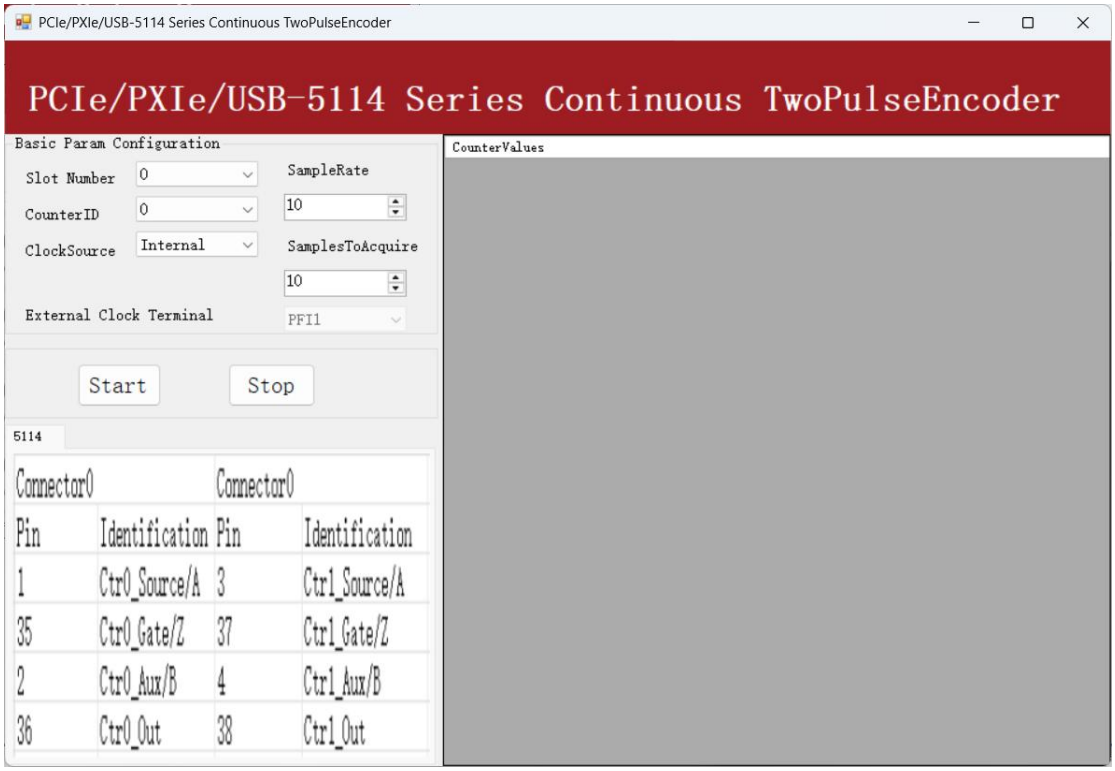


Figure 92 Two-PulseEncoder For Continuous Mode

- The table in the sample program is a connection diagram for your convenience.

- Click **Start** to start counting. You can see that the **CounterValue** is decreasing**,** which follows the counting rules explained in this chapter.

## 8.9. Counter Output Operations

### 8.9.1. Single Pulse Output

The PCIe/PXIe-5114-H7 timer/counter can output a single pulse with a specified pulse width. The timing diagram of the pulse output is shown in Figure 93.

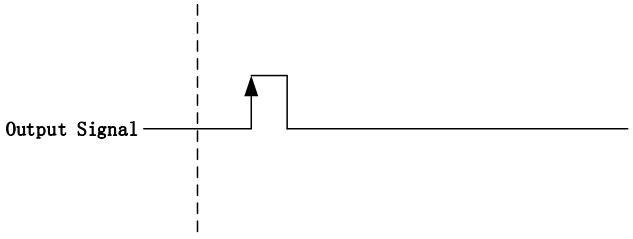


Figure 93 Single Pulse Output

In single pulse output mode, the user could set up the pulse width by configuring the frequency and duty cycle.

If you want to generate a single pulse with 1 ms pulse width, the parameter, frequency should be setup 500Hz and the duty cycle is 50%. Here is the formula for frequency setting:

Frequency = 1 / (1ms / 0.5) = 500Hz

**Learn by Example 8.9.1**

- To see the signal that PCIe-5114-H7 Counter Output generates, it is recommended to connect PCIe-5114-H7 Counter Output (CTR0_OUT, Pin#36) to PCIe-5114-H7 AI Ch0 input (AI0+, Pin#18). Please note Counter Output and AI Ch0 input share the same ground so only one connection is needed.
- Open **Counter Output-->Winform CO Single** and click **Start** and set the numbers as follow:

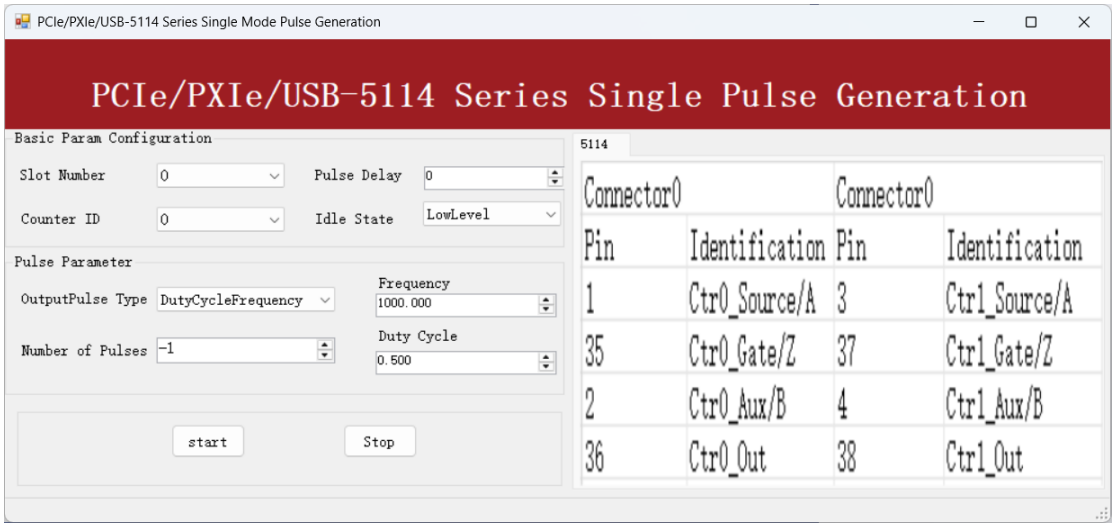


Figure 94 Single Pulse Generation

- The table in the sample program is a connection diagram for your convenience.
- The frequency and duty cycle of the pulse are set by **Frequency** and **Duty Cycle**.

- Please refer

- Learn by **Example** to configure an analog input to receive the signal from Counter Output.
- Click **Start** to generate a single pulse as shown.

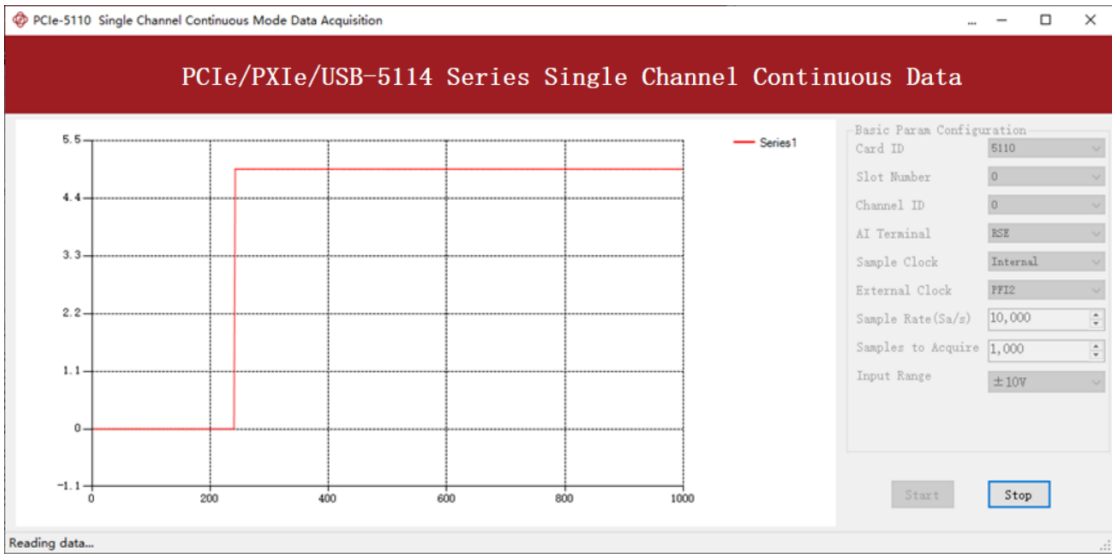


Figure 95 AI Acquisition Single Pulse

### 8.9.2. Finite Pulse Output

The pulse output timing is as shown in Figure 96.

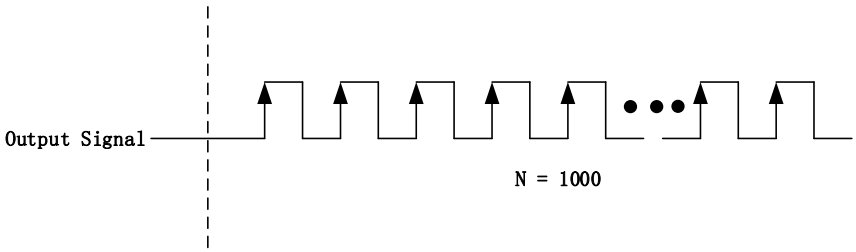


Figure 96 Finite Pulse Output

In finite pulse output mode, the user is required to configure the output frequency, duty cycle and the number of output pulses.

Assuming that the pulse width to be output by the user is 1ms, the frequency calculated according to the duty cycle of 50% is as follows:

Set frequency = 1 / (1ms / 0.5) = 500Hz

That is to say, when the user sets the frequency as 500Hz and the duty cycle as 0.5, a limited pulse of 1ms pulse width will be obtained.

**Learn by Example 8.9.2**

- To see the signal that PCIe-5114-H7 Counter Output generates, it is recommended to connect PCIe-5114-H7 Counter Output (CTR0_OUT, Pin#36) to PCIe-5114-H7 AI Ch0 input (AI0+, Pin#18). Please note Counter Output and AI Ch0 input share the same ground so only one connection is needed.
- Open **Counter Output-->Winform CO Finite** and click **Start** and set the numbers as follow:

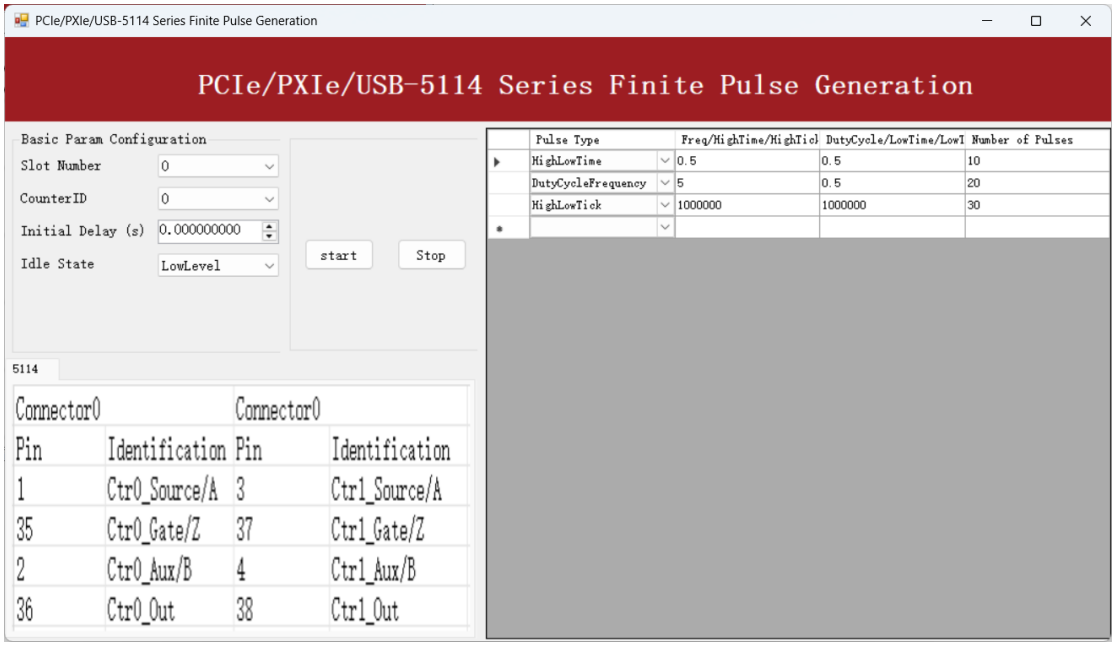


Figure 97 Finite Pulses Generation

- The table in the sample program is a connection diagram for your convenience.
- The frequency and duty cycle of the pulse are set by **Frequency** and **Duty Cycle**.

- Please refer
- Learn by **Example** to configure an analog input to receive the signal from Counter Output.

■ Click **Start** to generate the pulse shown below.


Figure 98 AI Acquisition Finite Pulse

➢ According to the picture, the *duty cycle* is 0.5 as set before.

### 8.9.3. Continuous Pulse Output

The pulse output timing is shown in Figure 99 below.

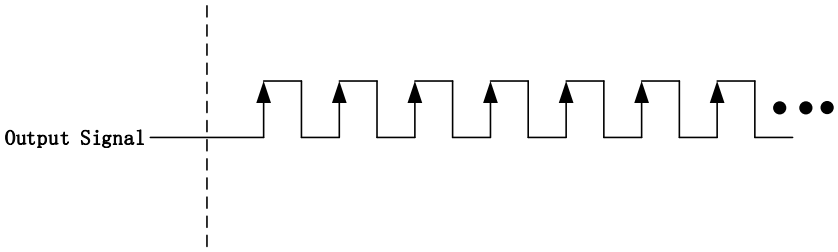


Figure 99 Continuous Pulse Output

In continuous output mode, you need to configure the output frequency and duty cycle. After starting the output, the pulse signal with fixed frequency and duty cycle will be output continuously.

**Learn by Example 8.9.3**

■ To see the signal that PCIe-5114-H7 Counter Output generates, it is recommended to connect PCIe-5114-H7 Counter Output (CTR0_OUT, Pin#36) to

PCIe-5114-H7 AI Ch0 input (AI0+, Pin#18). Please note Counter Output and AI Ch0 input share the same ground so only one connection is needed.

■ Open **Counter Output-->Winform CO Continuous** and click **Start** and set the numbers as follow:

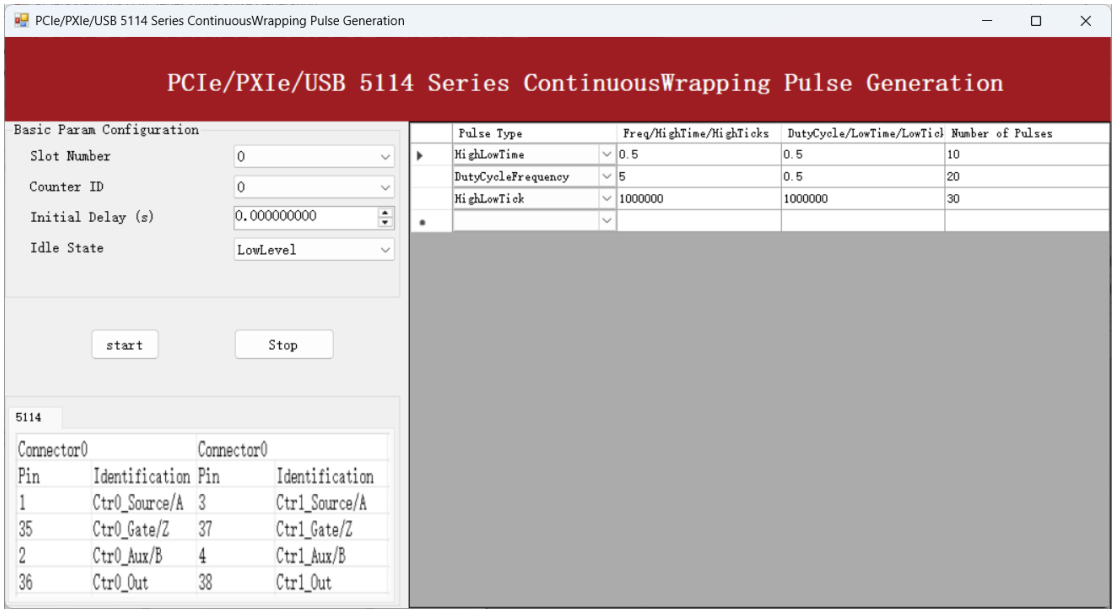


Figure 100 Continuous Pulse Generation

➢ The table in the sample program is a connection diagram for your convenience.

➢ The frequency and duty cycle of the pulse are set by **Frequency** and **Duty Cycle**.

■ Change the **Duty Cycle** to 0.7 for instance. The result is shown below.

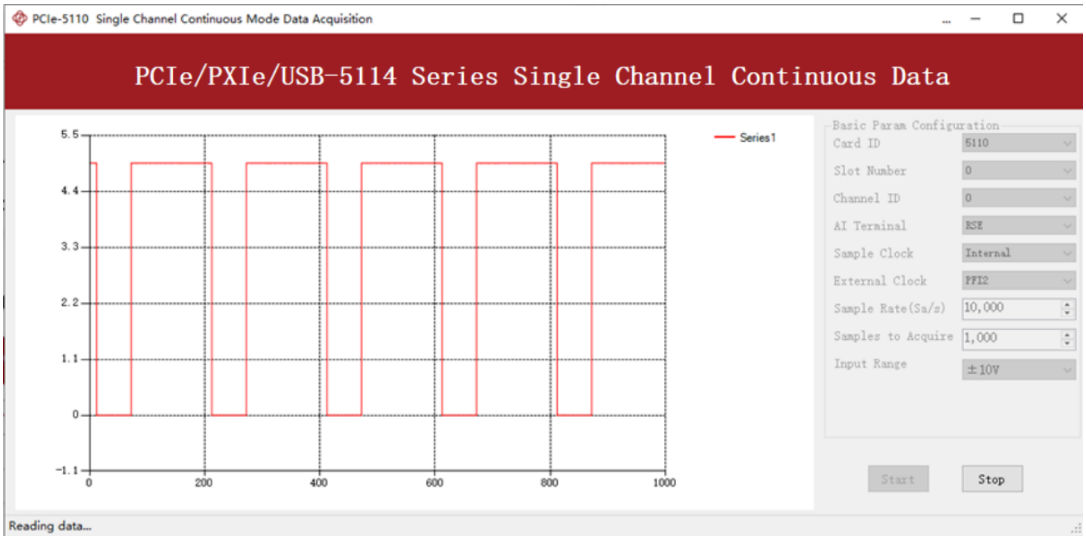


Figure 101 AI Acquisition Continuous Pulse

- According to the picture, the **duty cycle** is 0.7 as set before.

## 8.10. System Synchronization Interface（SSI）for PCIe Modules

The synchronization between PCIe modules are handled differently from the PXIe synchronization, it is implemented by the system synchronization interface (SSI). SSI is designed as a bidirectional bus and it can synchronize up to four PCIe modules. One PCIe module is designated as the master module and the other PCIe modules are designated as the slave modules.

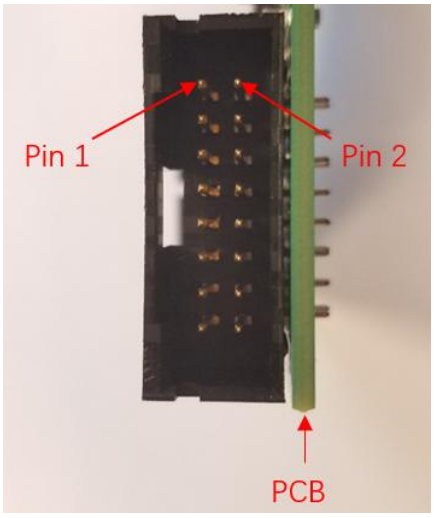


Figure 102 SSI Connector in PCIe-5114-H7

| Pin | Signal Name | Signal Name | Pin |
|---|---|---|---|
| 1 | PXI_TRIG0 | GND | 2 |
| 3 | PXI_TRIG1 | GND | 4 |
| 5 | PXI_TRIG2 | GND | 6 |
| 7 | PXI_TRIG3 | GND | 8 |
| 9 | PXI_TRIG4 | GND | 10 |
| 11 | PXI_TRIG5 | GND | 12 |
| 13 | PXI_TRIG6 | GND | 14 |
| 15 | PXI_TRIG7 | GND | 16 |

Table 20 SSI Connector Pin Assignment for PCIe-5114-H7

### 8.11. DIP Switch in PCIe-5114-H7

PCIe-5114-H7 modules have a DIP switch. The card number can be adjusted manually by changing the DIP switch setting, which is used to identify the boards with different slot positions.

# 9. Calibration

PCIe/PXIe-5114-H7 boards are precalibrated before the shipment. We recommend you recalibrate PCIe/PXIe-5114-H7 boards periodically to ensure the measurement accuracy. A commonly accepted practice is one year. If for any reason, you need to recalibrate your board, please contact JYTEK.

# 10. Appendix(Measurement Issues)

## 10.1. Performance and Tests

### 10.1.1. AI Bandwidth

| Analog Input Bandwidth | | |
|---|---|---|
| Nominal Range Full Scale (V) | Filter | -3dB Bandwidth (kHz) |
| All range | Narrow | 15 |
| 10V | Medium | 39 |
| 5V/2.5V | Medium | 33 |
| All range | Wide | 375 |

Table 21 AI Bandwidth

### 10.1.2. Analog filter group delay

| Analog filter group delay(us) | | | |
|---|---|---|---|
| Filter / Range(V) | Wide | Medium | Narrow |
| 10V | 0.9 | 5.2 | 13.2 |
| 5V | 0.9 | 6.2 | 13.2 |
| 2.5V | 0.94 | 6.2 | 13.3 |

Table 22 10.1.2. Analog filter group delay

## 10.2. Floating Signals and Ground Referenced Signals

Signals to be measured often fall into two categories: floating and ground referenced. The floating signals include battery output, isolated output, thermocouples etc; the ground referenced signals include most instrumentation output signals. Some instruments also offered isolated floating output.

## 10.3. Differential, RSE Modes

The DAQ boards have three measurement modes: differential (DIFF), and the referenced single end (RSE).

The two measurement modes and the two types of input signals, floating and ground referenced, form 6 different measurement scenarios as shown in the following.

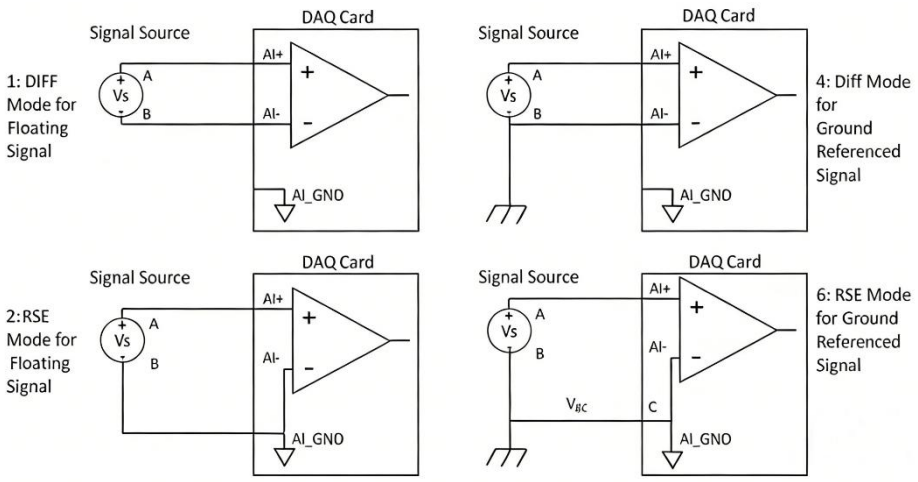


Figure 103 Six Measurement Scenarios

In the first 5 scenarios, $V_{AB}$ is measured voltage. But in the 6$^{th}$ scenario, both the measured signal and the DAQ have own grounds. The two ground may have a voltage difference $V_{BC}$. The actual measurement is $V_{AC}=V_{AB}+V_{BC}$, not $V_{AB}$. Due to the ground noise, $V_{BC}$ is quite noisy. This affects the measurement accuracy. The caution must be taken using 6th mode.

## 10.4. Reducing the Common Mode Voltage Effect

In the first 2 modes, the measured signal is floating. It is quite often that the common mode voltage will appear. To reduce this effect on the measurement accuracy, a resister can be added as shown. The value of this resister depends on the impedance of the signal source. As a rule of thumb, R should be 1000 times of the signal source output impedance, roughly 10K to 100KΩ. At this level, R has very little impact on the measurement.

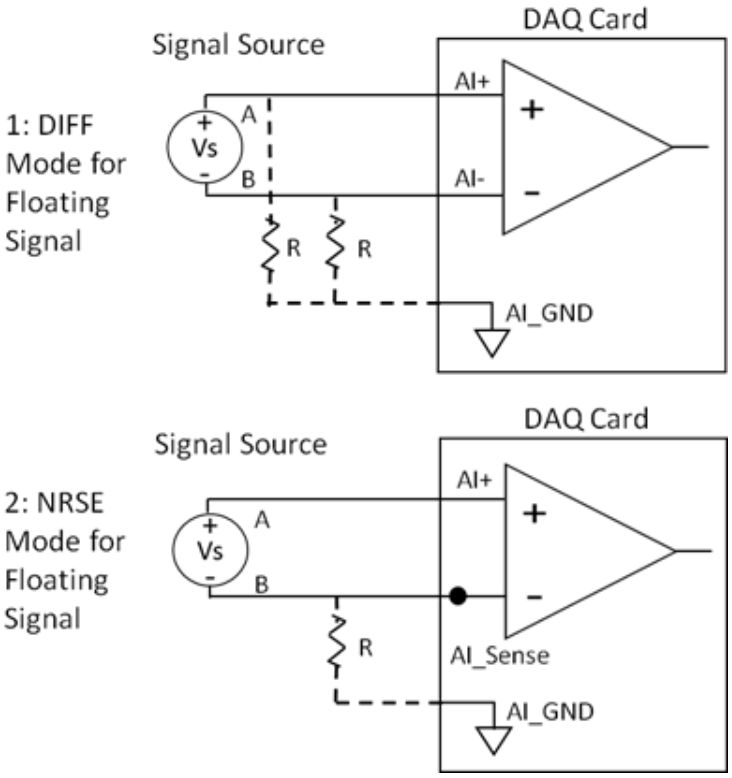


Figure 104 Using Resister to Reduce Common Mode Voltage Effect

# 11. About JYTEK

## 11.1. JYTEK China

Founded in June, 2016, JYTEK China is a leading Chinese test & measurement company, providing complete software and hardware products for the test and measurement industry. The company has evolved from re-branding and reselling PXI(e) and DAQ products to a fully-fledged product company. The company offers complete lines of PXI, DAQ, USB products. More importantly, JYTEK has been promoting open-sourced based ecosystem and offers complete software products. Presently, JYTEK is focused on the Chinese market. Our Shanghai headquarters and production service center have regular stocks to ensure timely supply; we also have R&D centers in Xi'an and Chongqing. We also have highly trained direct technical sales representatives in Shanghai, Beijing, Tianjin, Xi'an, Chengdu, Nanjing, Wuhan, Guangdong, Haerbin, and Changchun. We also have many patners who provide system level support in various cities.

## 11.2. JYTEK Software Platform

JYTEK has developed a complete software platform, SeeSharp Platform, for the test and measurement applications. We leverage the open sources communities to provide the software tools. Our platform software is also open sourced and is free, thus lowering the cost of tests for our customers. We are the only domestic vendor to offer complete commercial software and hardware tools.

## 11.3. JYTEK Warranty and Support Services

With our complete software and hardware products, JYTEK is able to provide technical and sales services to wide range of applications and customers. In most cases, our products are backed by a 1-year warranty. For technical consultation, pre-sale and after-sales support, please contact JYTEK of your country.

# 12. Statement

The hardware and software products described in this manual are provided by JYTEK China, or JYTEK in short.

This manual provides the product review, quick start, some driver interface explanation for PCIe/PXIe-5114-H7 of temperature sensor data acquisition cards. The manual is copyrighted by JYTEK.

No warranty is given as to any implied warranties, express or implied, including any purpose or non-infringement of intellectual property rights, unless such disclaimer is legally invalid. JYTEK is not responsible for any incidental or consequential damages related to performance or use of this manual. The information contained in this manual is subject to change without notice.

While we try to keep this manual up to date, there are factors beyond our control that may affect the accuracy of the manual.  Please check the latest manual and product information from our website.

Shanghai Jianyi Technology Co., Ltd.

Address: Room 201, Building 3, NO.300 Fangchun Road, Shanghai.

Post Code: 201203

Tel: 021-5047 5899

Website: www.jytek.com