1. JY-7131 Specifications

1.1 Overview

JY-7131 series modules are multi-function isolated digital I/O module, which can provide 64 channels of DIO (32 Sinking/Sourcing Inputs, 32 Sinking/Sourcing/Push-pull Outputs, Bank-Isolated) or 8 channels of counter, supporting counting, frequency measurement, encoder, pulse output and other functions. Industrial DIO modules, like the JY-7131 series, are ideal for motor, valve control and automation control.

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

1.2 Main Features

- DI：
  - 32 channels DI, 0~55 VDC
  - Support Sourcing and Sinking input mode
- DO：
  - 32 channels DO,10~50 VDC, 200 kHz
  - Suport Sourcing, Sinking and Push-pull tri-state output mode
- Counter：
  - High operating voltage counter up to 50V
  - Up to 100 MHz internal clock rate
  - Counter Functions：

    Edge counting / Frequency measurement /Period measurement / Two-Edge separation/Quadrature/(x1/x2/x4) encoder/Two-Pulse encoder
  - Dynamic reconfigurable counter output
- Complies with IEC 61131-2 standard

1.3 Hardware Specifications

1.3.1 Isolated Digital Input

Number of Channel	32
Input Voltage	Maximum input voltage: 55 V DC Input logic low voltage (VIL): 0~±8.5 VDC Input logic high voltage (VIH): ±10~±55 VDC
Input Type	Sourcing/Sinking
Input Isolated Voltage	2500 Vrms

Table 1 Isolated Digital Input

- **Input Current**

Minimum	2.23 mA
Maximum	2.8 mA
Typical	2.48 mA

Table 2 Input Current

1.3.2 Isolated Digital Output

Number of Channel	32
V_{DD}(Output Supply Voltage)	10~50 VDC
Output Type	Sourcing/Sinking/Push-Pull
Output Voltage	10~50 VDC

Maximum Output Current	Out=High: 640 mA
	Out=Low: 440 mA

Output Isolated Voltage	3000 Vrms

Table 3 Isolated Digital Output

- **DO Output Mode**

DO Output Mode	DO Value	Output Status
Sourcing	1	High Level (VDD)
	0	High Z
Sinking	1	Low Level (GND)
	0	High Z
Push-Pull	1	High Level (VDD)
	0	Low Level (GND)

Table 4 DO Output Mode

1.3.3 Counter Specifications

Number of counters	8
Resolution	32 bits
Internal Clock Rate	100 MHz
Counter Input Functions	Edge counting Period measurement Frequency measurement Two-edge separation measurement Quadrature(×1/×2/×4) encoder Two-pulse encoder
Counter Input Frequency	1MHz(High Level ≥300ns)
Counter Output Functions	Single Finite Continuous pulse PWM
Counter Output Frequency	200 kHz
FIFO	16 M Samples

Counter Input Signal	Gate(Z)
	Source(A)
	Aux(B)
	Star Trigger
	External Sample Clock

Counter Output Signal	OUT

Table 5 Counter Specifications

1.3.4 External Digital Trigger

Trigger functions	Trigger source	PXI_TRIG<0..7> PFI_In<0..31>
	Polarity	Rising Edge/Falling Edge
	Counter /Timer functions	Start trigger
Module to module Trigger bus	Input source	PXI_TRIG<0..7>
	Output destination	PXI_TRIG<0..7>

Table 6 External Digital Trigger

1.3.5 Bus Interface

Bus interface	PXIe standard	x4 PXI Express peripheral module Specification V1.0 compliant
	Slot supported	x1 and x4 PXI Express or PXI Express hybrid slots

Table 7 Bus Interface

1.3.6 Physical Size

Dimensions	3U PXIe
Weight	213.6 g

Table 8 Physical Size

1.3.7 Operating Environment

Ambient temperature range	0°C~50°C
Relative humidity range	20% to 80%, none-condensing

Table 9 Operating Environment

1.3.8 Storage Environment

Ambient temperature range	-20 °C to 50 °C
Relative humidity range	10% to 90%, none-condensing

Table 10 Storage Environment

Table of Contents

2. Introduction

This chapter presents the information how to use this manual and quick start if you are already familiar with Microsoft Visual Studio and C# programming language.

2.1 Abbreviations

DI: Digital Input

DO: Digital Output

CI: Counter Input

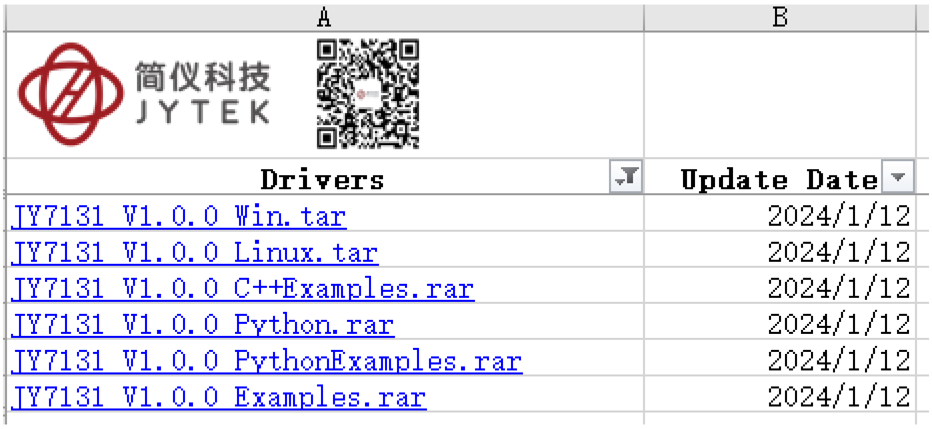
CO: Counter Output

DAQ: Data AcQuisition

PFI: Programmable Function Interface

2.2 Learn by Example

JYTEK has added **Learn by Example** in this manual. We provide many sample programs for this device. Please download and install the sample programs for this device. You can download a JYPEDIA excel file from our web www.jytek.com. Open JYPEDIA and search for JY7131 in the driver sheet, select **JY7131.Examples.zip**. This will lead you to download the sample program for this device. In addition to the download information, JYPEDIA also has a lot of other valuable information, JYTEK highly recommend you use this file to obtain information from JYTEK.



A	B
Drivers	Update Date
JY7131_V1.0.0_Win.tar	2024/1/12
JY7131_V1.0.0_Linux.tar	2024/1/12
JY7131_V1.0.0_C++Examples.rar	2024/1/12
JY7131_V1.0.0_Python.rar	2024/1/12
JY7131_V1.0.0_PythonExamples.rar	2024/1/12
JY7131_V1.0.0_Examples.rar	2024/1/12

Figure 1 JYPEDIA Information

In a **Learn by Example** section, the sample program is in bold style such as **Counter Input-->Winform CI Single Edge Counting**; the property name in the sample program is also in bold style such as **SamplesToAcquire**; the technical names used in the manual is in italic style such as *SampleRate*. You can easily relate the property names in the example program with the manual documentation.

In an **Learn by Example** section, the experiment is set up as follow. A JY-7131 card is plugged in a desktop computer. The JY-7131 is connected to a DIN-100 terminal block. A signal source is also connected to the same terminal block.

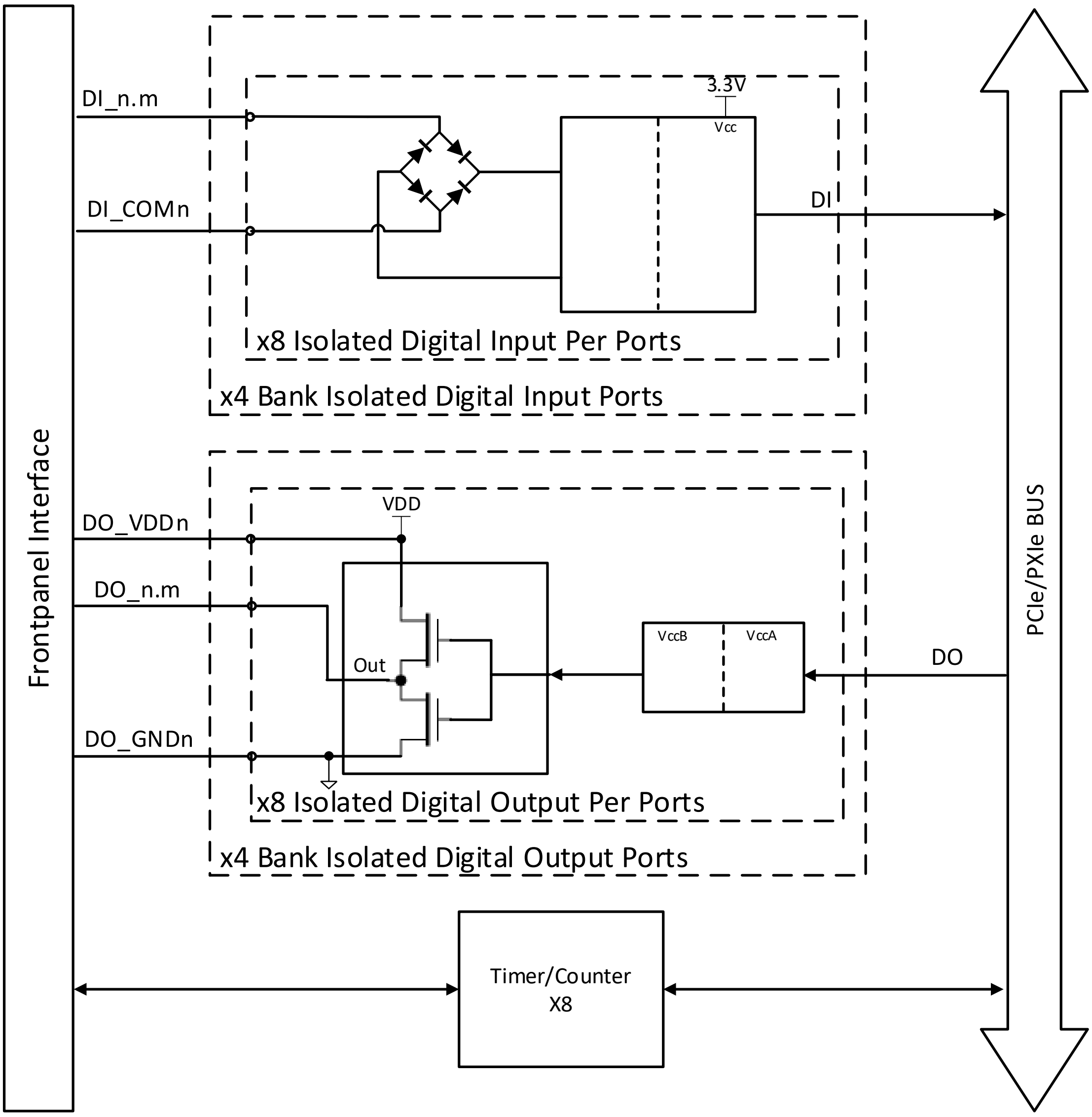
The DIN-100 has 4 terminal columns, J1 – J4. In the rest of this manual, the wire connection in each **Learn by Example** section will be given by the pin numbers only.

Tip: JY-7131 also has the counter output capability. If you do not have a signal source, you can use the outputs of JY-7131 as the signal source. In this case you need first run example program ***Counter Output--> Winform CO Continuous Wrapping*** to generate the output.

3. Hardware Specifications

3.1 System Diagram

The system block diagram of JY-7131 series is shown in Figure 2.



Frontpanel Interface
DI_n.m
DI_COMn
3.3V
Vcc
DI
x8 Isolated Digital Input Per Ports
x4 Bank Isolated Digital Input Ports
DO_VDDn
DO_n.m
DO_GNDn
VDD
Out
VccB
VccA
DO
x8 Isolated Digital Output Per Ports
x4 Bank Isolated Digital Output Ports
Timer/Counter
X8
PCIe/PXIe BUS

Figure 2 JY-7131 Series System Block Diagram

3.2 Front Panel and Pin Definition

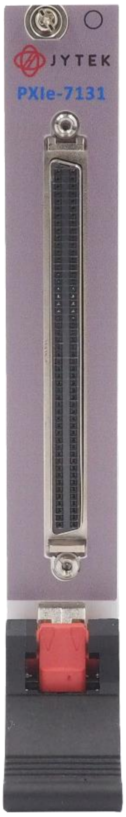


Figure 3 Front Pannel

Pin Number	Signal	Pin Number	Signal
1	DI_0.0 / PFI_In_0	26	DO_0.0 / PFI_Out_0
2	DI_0.1 / PFI_In_1	27	DO_0.1 / PFI_Out_1
3	DI_0.2 / PFI_In_2	28	DO_0.2 / PFI_Out_2
4	DI_0.3 / PFI_In_3	29	DO_0.3 / PFI_Out_3
5	DI_0.4 / PFI_In_4	30	DO_0.4 / PFI_Out_4
6	DI_0.5 / PFI_In_5	31	DO_0.5 / PFI_Out_5
7	DI_0.6 / PFI_In_6	32	DO_0.6 / PFI_Out_6
8	DI_0.7 / PFI_In_7	33	DO_0.7 / PFI_Out_7
9	DI_COM0	34	DO_VDD0
10	DI_COM0	35	DO_GND0
11	DI_COM0	36	DO_GND0
12	DI_COM0	37	DO_GND0
13	DI_2.0 / PFI_In_16	38	DO_2.0 / PFI_Out_16
14	DI_2.1 / PFI_In_17	39	DO_2.1 / PFI_Out_17
15	DI_2.2 / PFI_In_18	40	DO_2.2 / PFI_Out_18
16	DI_2.3 / PFI_In_19	41	DO_2.3 / PFI_Out_19
17	DI_2.4 / PFI_In_20	42	DO_2.4 / PFI_Out_20
18	DI_2.5 / PFI_In_21	43	DO_2.5 / PFI_Out_21
19	DI_2.6 / PFI_In_22	44	DO_2.6 / PFI_Out_22
20	DI_2.7 / PFI_In_23	45	DO_2.7 / PFI_Out_23
21	DI_COM2	46	DO_VDD2
22	DI_COM2	47	DO_GND2
23	DI_COM2	48	DO_GND2
24	DI_COM2	49	DO_GND2
25	NC	50	NC
51	DI_1.0 / PFI_In_8	76	DO_1.0 / PFI_Out_8
52	DI_1.1 / PFI_In_9	77	DO_1.1 / PFI_Out_9
53	DI_1.2 / PFI_In_10	78	DO_1.2 / PFI_Out_10
54	DI_1.3 / PFI_In_11	79	DO_1.3 / PFI_Out_11
55	DI_1.4 / PFI_In_12	80	DO_1.4 / PFI_Out_12
56	DI_1.5 / PFI_In_13	81	DO_1.5 / PFI_Out_13
57	DI_1.6 / PFI_In_14	82	DO_1.6 / PFI_Out_14
58	DI_1.7 / PFI_In_15	83	DO_1.7 / PFI_Out_15
59	DI_COM1	84	DO_VDD1
60	DI_COM1	85	DO_GND1
61	DI_COM1	86	DO_GND1
62	DI_COM1	87	DO_GND1
63	DI_3.0 / PFI_In_24	88	DO_3.0 / PFI_Out_24
64	DI_3.1 / PFI_In_25	89	DO_3.1 / PFI_Out_25
65	DI_3.2 / PFI_In_26	90	DO_3.2 / PFI_Out_26
66	DI_3.3 / PFI_In_27	91	DO_3.3 / PFI_Out_27
67	DI_3.4 / PFI_In_28	92	DO_3.4 / PFI_Out_28
68	DI_3.5 / PFI_In_29	93	DO_3.5 / PFI_Out_29
69	DI_3.6 / PFI_In_30	94	DO_3.6 / PFI_Out_30
70	DI_3.7 / PFI_In_31	95	DO_3.7 / PFI_Out_31
71	DI_COM3	96	DO_VDD3
72	DI_COM3	97	DO_GND3
73	DI_COM3	98	DO_GND3
74	DI_COM3	99	DO_GND3
75	NC	100	NC

Legend:
“n” stands for port number, "m" stands for line number
DI_COMn：Common terminal for input port n
DO_VDDn： Power line for port n
DO_GNDn：Ground for port n

Table 11 Pin Defination

3.3 Default Routing for Counter Input/Output Signals

All counter input and output terminals are routed to a certain PFI by default as shown in Table 12 and Table 13

Functions	Signal Type	Ctr0	Ctr1	Ctr2	Ctr3
Edge Counting	Signal To Measure(Source)	PFI_In_0 (1)	PFI_In_4 (5)	PFI_In_8 (51)	PFI_In_12 (55)
	Pause Trigger(Gate)	PFI_In_1 (2)	PFI_In_5 (6)	PFI_In_9 (52)	PFI_In_13 (56)
	Count Direction(Aux)	PFI_In_2 (3)	PFI_In_6 (7)	PFI_In_10 (53)	PFI_In_14 (57)
	SampleClock Terminal	PFI_In_3 (4)	PFI_In_7 (8)	PFI_In_11 (54)	PFI_In_15 (58)
	Output(Out)	PFI_Out_0 (26)	PFI_Out_1 (27)	PFI_Out_2 (28)	PFI_Out_3 (29)
Frequency Measurement	Signal To Measure(Gate)	PFI_In_1 (2)	PFI_In_5 (6)	PFI_In_9 (52)	PFI_In_13 (56)
	SampleClock Terminal	PFI_In_3 (4)	PFI_In_7 (8)	PFI_In_11 (54)	PFI_In_15 (58)
Period Measurement	Signal To Measure(Gate)	PFI_In_1 (2)	PFI_In_5 (6)	PFI_In_9 (52)	PFI_In_13 (56)
	SampleClock Terminal	PFI_In_3 (4)	PFI_In_7 (8)	PFI_In_11 (54)	PFI_In_15 (58)
Two-Edge Separation Measurement	First Signal to Measure(Gate)	PFI_In_1 (2)	PFI_In_5 (6)	PFI_In_9 (52)	PFI_In_13 (56)
	Second Signal to Measure(Aux)	PFI_In_2 (3)	PFI_In_6 (7)	PFI_In_10 (53)	PFI_In_14 (57)
	SampleClock Terminal	PFI_In_3 (4)	PFI_In_7 (8)	PFI_In_11 (54)	PFI_In_15 (58)
Quadrature Encoder	A Signal(Source)	PFI_In_0 (1)	PFI_In_4 (5)	PFI_In_8 (51)	PFI_In_12 (55)
	B Signal(Aux)	PFI_In_2 (3)	PFI_In_6 (7)	PFI_In_10 (53)	PFI_In_14 (57)
	Z Signal(Gate)	PFI_In_1 (2)	PFI_In_5 (6)	PFI_In_9 (52)	PFI_In_13 (56)
	SampleClock Terminal	PFI_In_3 (4)	PFI_In_7 (8)	PFI_In_11 (54)	PFI_In_15 (58)
Two-Pulse Encoder	A Signal(Source)	PFI_In_0 (1)	PFI_In_4 (5)	PFI_In_8 (51)	PFI_In_12 (55)
	B Signal(Aux)	PFI_In_2 (3)	PFI_In_6 (7)	PFI_In_10 (53)	PFI_In_14 (57)
	SampleClock Terminal	PFI_In_3 (4)	PFI_In_7 (8)	PFI_In_11 (54)	PFI_In_15 (58)

Culumn "Ctr0" to "Ctr3": Pin Number is shown in (*).

Table 12 Counter Input/Output Default Routing 1

Functions	Signal Type	Ctr4	Ctr5	Ctr6	Ctr7
Edge Counting	Signal To Measure(Source)	PFI_In_16 (13)	PFI_In_20 (17)	PFI_In_24 (63)	PFI_In_28 (67)
	Pause Trigger(Gate)	PFI_In_17 (14)	PFI_In_21 (18)	PFI_In_25 (64)	PFI_In_29 (68)
	Count Direction(Aux)	PFI_In_18 (15)	PFI_In_22 (19)	PFI_In_26 (65)	PFI_In_30 (69)
	SampleClock Terminal	PFI_In_19 (16)	PFI_In_23 (20)	PFI_In_27 (66)	PFI_In_31 (70)
	Output(Out)	PFI_Out_4 (30)	PFI_Out_5 (31)	PFI_Out_6 (32)	PFI_Out_7 (33)
Frequency Measurement	Signal To Measure(Gate)	PFI_In_17 (14)	PFI_In_21 (18)	PFI_In_25 (64)	PFI_In_29 (68)
	SampleClock Terminal	PFI_In_19 (16)	PFI_In_23 (20)	PFI_In_27 (66)	PFI_In_31 (70)
Period Measurement	Signal To Measure(Gate)	PFI_In_17 (14)	PFI_In_21 (18)	PFI_In_25 (64)	PFI_In_29 (68)
	SampleClock Terminal	PFI_In_19 (16)	PFI_In_23 (20)	PFI_In_27 (66)	PFI_In_31 (70)
Two-Edge Separation Measurement	First Signal to Measure(Gate)	PFI_In_17 (14)	PFI_In_21 (18)	PFI_In_25 (64)	PFI_In_29 (68)
	Second Signal to Measure(Aux)	PFI_In_18 (15)	PFI_In_22 (19)	PFI_In_26 (65)	PFI_In_30 (69)
	SampleClock Terminal	PFI_In_19 (16)	PFI_In_23 (20)	PFI_In_27 (66)	PFI_In_31 (70)
Quadrature Encoder	A Signal(Source)	PFI_In_16 (13)	PFI_In_20 (17)	PFI_In_24 (63)	PFI_In_28 (67)
	B Signal(Aux)	PFI_In_18 (15)	PFI_In_22 (19)	PFI_In_26 (65)	PFI_In_30 (69)
	Z Signal(Gate)	PFI_In_17 (14)	PFI_In_21 (18)	PFI_In_25 (64)	PFI_In_29 (68)
	SampleClock Terminal	PFI_In_19 (16)	PFI_In_23 (20)	PFI_In_27 (66)	PFI_In_31 (70)
Two-Pulse Encoder	A Signal(Source)	PFI_In_16 (13)	PFI_In_20 (17)	PFI_In_24 (63)	PFI_In_28 (67)
	B Signal(Aux)	PFI_In_18 (15)	PFI_In_22 (19)	PFI_In_26 (65)	PFI_In_30 (69)
	SampleClock Terminal	PFI_In_19 (16)	PFI_In_23 (20)	PFI_In_27 (66)	PFI_In_31 (70)

Table 13 Counter Input/Output Default Routing 2

Culumn "Ctr4" to "Ctr7": Pin Number is shown in (*).

4. Order Information

- PXIe-7131 (PN: JY5046684-01)

  64-Channel, 50 VDC, 32 Sink/Source Inputs, 32 Push-pull Outputs, Bank-Isolated PXIe Digital I/O Module

- Accessories:

  Cable:

  ACL-1020100-1 1M 100pin SCSI twisted pair cable (PN: JY4910923-01)

  ACL-1020100-2 2M 100pin SCSI twisted pair cable (PN: JY7155665-01)

  Terminal Block:

  DIN-100-1 100-Pin SCSI Terminal block (PN: JY7739162-01)

5. Software

5.1 System Requirements

JY-7131 modules 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 will at the best support the following Linux versions.

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 14 Supported Linux Versions

5.2 System Software

When using the JY-7131 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 JY-7131 with .NET Framework 4.0 with Microsoft Visual Studio 2015. JYTEK relies on Microsoft to maintain the compatibility for the newer versions.

5.3 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.

5.4 JY-7131 Series Hardware Driver

After installing the required application development environment as described above, you need to install the JY-7131 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 JY-7131 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 module works, you should be able to know how to use all other DAQ hardware by using the same methods.

5.5 Install the SeeSharpTools from JYTEK

To efficiently and effectively use JY-7131 boards, 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 JY-7131 hardware. Please register and down load the latest SeeSharpTools from our website, www.jytek.com.

5.6 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.

6. Operating JY-7131

This chapter provides the operation guides for JY-7131, including Timer and programmable I/O interface, etc.

JYTEK provides extensive examples, on-line help and documentation to assist you to use the JY-7131 module. 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.

6.1 Quick Start

After you have installed the driver software and the SeeSharpTools, you are ready to use Microsoft Visual Studio C# to operate the JY-7131 products.

If you are already familiar with Microsoft Visual Studio C#, the quickest way to use JY-7131 boards 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.

6.2 Digital I/O Operations

The JY-7131 provides 32 channels of isolated digital input and 32 channels of isolated digital output.

6.2.1 Digital Input Connection Guide

The JY-7131 device supports 32 channels of isolated digital input. Here's a guide to connect an input signal to a channel.

- **Sinking Configuration**

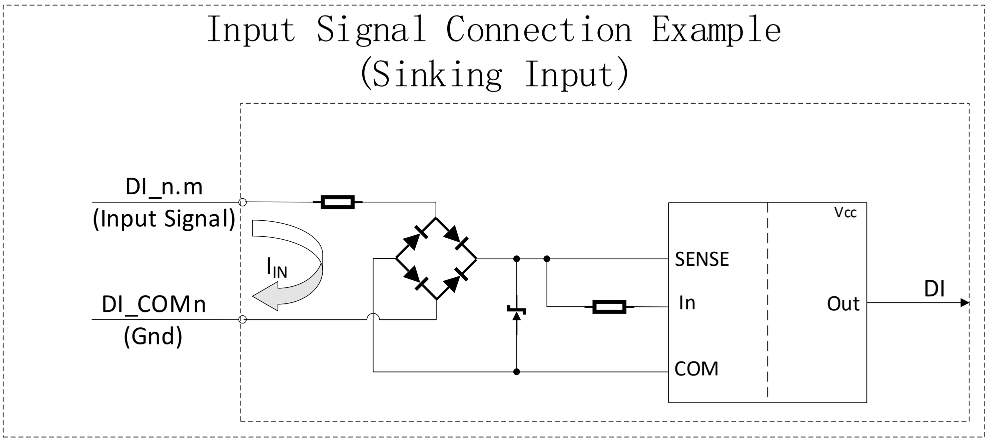


Figure 4 Input Signal Connection Example（Sinking Input）

1. **Identify the Input Terminal:** Locate the input channel on the device you intend to use, typically labeled as DI_n.m where “n” represents the port number and “m” represents the line number.

2. **Connect the Input Signal:** Connect the positive input signal to the input terminal (DI_n.m) of the module.
3. **Connect the COM:** Connect the ground of your signal source to the COM terminal (DI_COMn) of the module.
4. **Operation:** Applying a DC voltage of at least 10 V across the two input terminals (DI_n.m and DI_COMn) represents a logic high for that input.
   - If no voltage is present (a voltage of 8.5 V or less), the module represents a logic low for that input.
   - DC voltages between 8.5 V and 10 V are invalid and represent an unreliable value.

- **Sourcing Configuration**

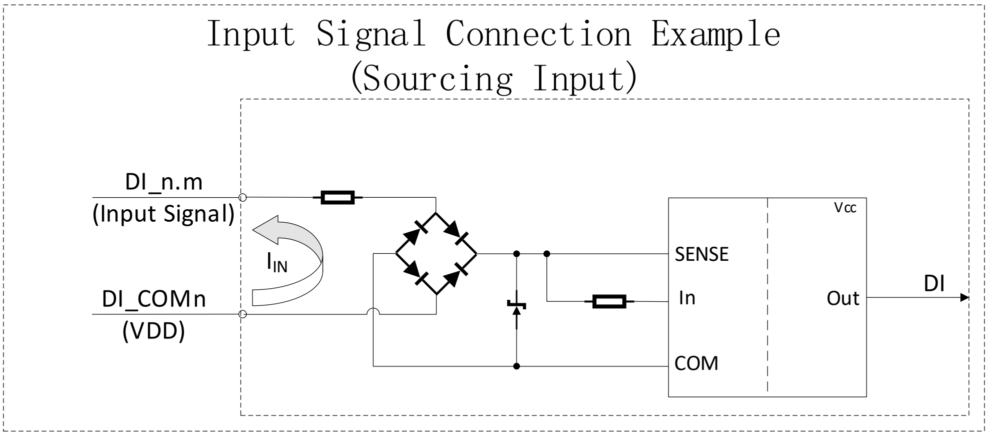


Figure 5 Input Signal Connection Example（Sourcing Input）

1. **Identify the Input Terminal:** Locate the input channel on the device you intend to use, typically labeled as DI_n.m where “n” represents the port number and “m” represents the line number.
2. **Connect the Input Signal:** Connect the negative input signal to the input terminal (DI_n.m) of the module.
3. **Connect the COM:** Connect the VDD of your signal source to the COM terminal (DI_COMn) of the module.
4. **Operation:** Applying a DC voltage of at least 10 V across the two input terminals (DI_n.m and DI_COMn) represents a logic high for that input.
   - If no voltage is present (a voltage of 8.5 V or less), the module represents a logic low for that input.
   - DC voltages between 8.5 V and 10 V are invalid and represent an unreliable value.

6.2.2 Digital Output Connection Guide

The JY-7131 device also supports 32 channels of isolated digital output. There are three configurations for digital output: Sourcing, Sinking and Push-pull output mode.

DO Output Mode	DO Value	Output Status
Sourcing	1	High Level (VDD)
	0	High Z
Sinking	1	Low Level (GND)
	0	High Z
Push-Pull	1	High Level (VDD)
	0	Low Level (GND)

Figure 6 Digital Output Mode Table

- **Sourcing Configuration**

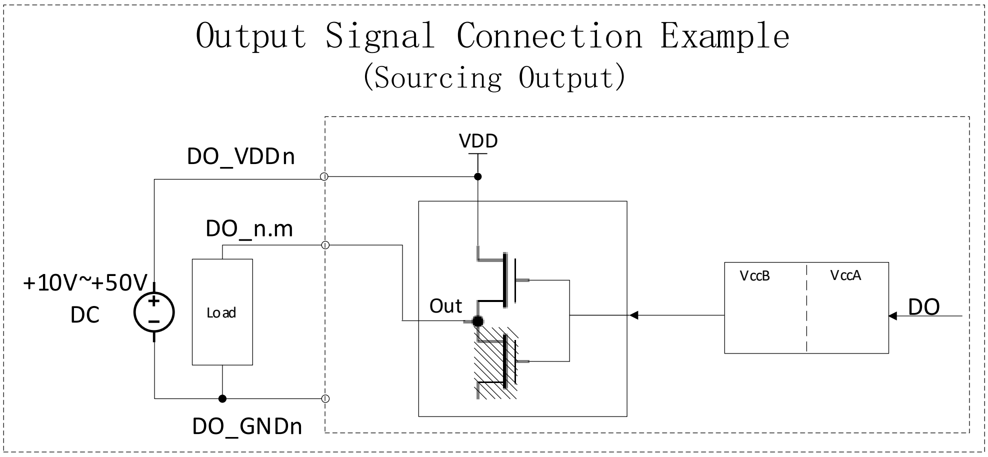


Figure 7 Output Signal Connection Example （Sourcing Output）

1. **Load Connection:** Connect the load to the DO_GNDn and DO_n.m terminals, ensuring correct polarity if the load is polarity sensitive.
2. **Power Source:** Connect a positive of power source (DC) to the DO_VDDn terminal.
3. **Ground Connection:** Connect the negative of the power source (DC) to the ground terminal (DO_GNDn) of the module.
4. Operation:
   - Writing a 1 (logic high) to an output line switches on one channel of the module and allows current to pass through the output line.
   - Writing a 0 (logic low) to an output line switches off one channel of the module and prohibits current from passing through the output line.

- **Sinking Configuration:**

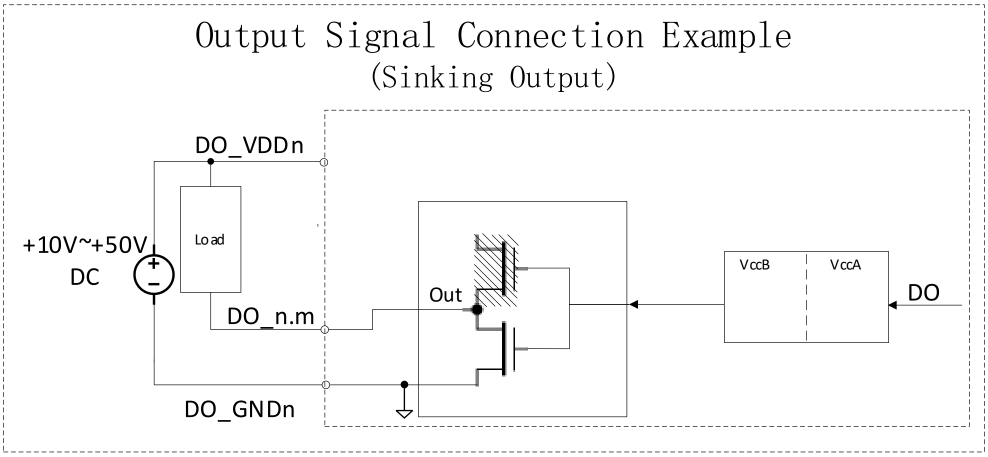


Figure 8 Output Signal Connection Example（Sinking Output）

1. **Load Connection:** Connect one end of the load to the DO_n.m terminal and the other end to the DO_VDDn terminal.
2. **Power Source:** Connect a power source (VDD) to the DO_VDDn terminal.
3. **Common Ground:** Connect the ground of the power source to the common ground terminal (DO_GNDn) of the module.
4. **Operation:** Use the module's software to control the output, which will complete the circuit and allow current to flow through the load.

- **Push-Pull Configuration:**

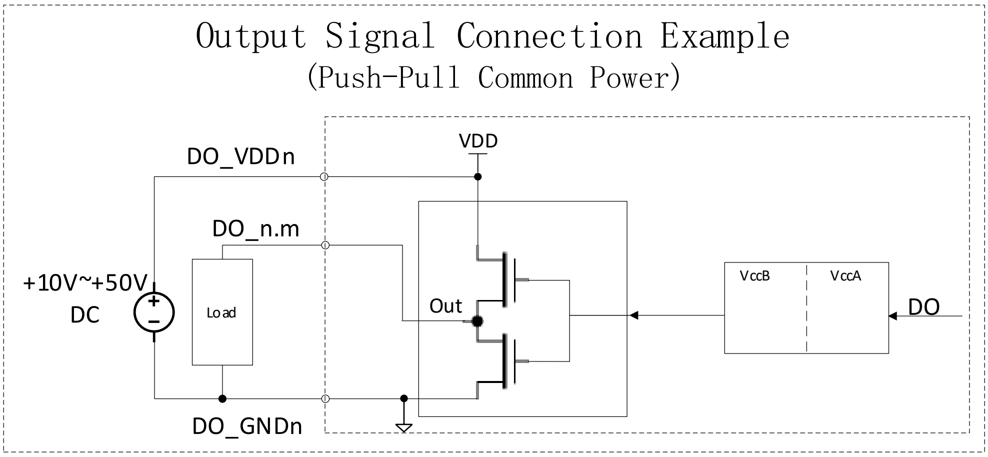


Figure 9 Output Signal Connection Example（Push-Pull Common Power）

1. **Load Connection:** Connect the load to the DO_GNDn and Do_n.m terminals, ensuring correct polarity if the load is polarity sensitive.
2. **Power Source:** Connect a positive of power source (DC) to the DO_VDDn terminal.
3. **Ground Connection:** Connect the negative of the power source (DC) to the ground terminal (DO_GNDn) of the module.

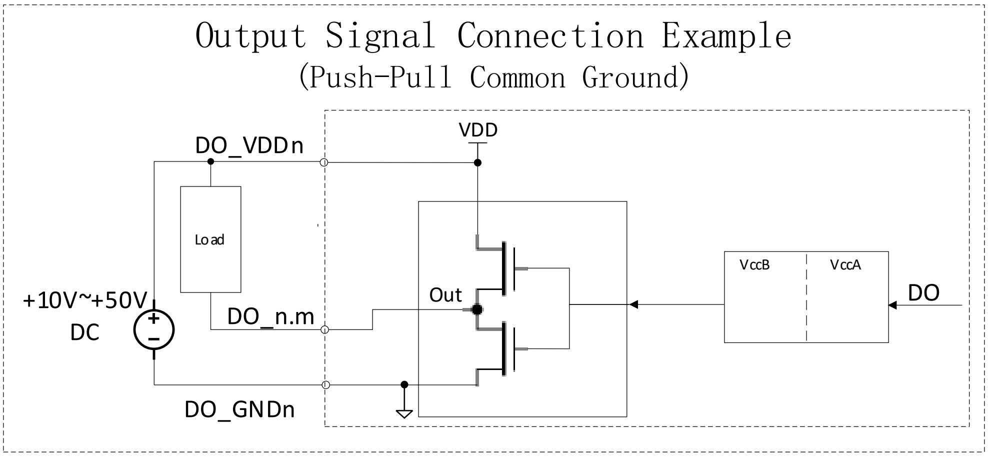


Figure 10 Output Signal Connection Example （Push-Pull Common Ground）

1. **Load Connection:** Connect one end of the load to the Don.m terminal and the other end to the DO_VDDn terminal.
2. **Power Source:** Connect a power source (VDD) to the DO_VDDn terminal.
3. **Common Ground:** Connect the ground of the power source to the common ground terminal (DO_GNDn) of the module.

Please ensure all connections are secure and that the module's power is off before making or changing connections to prevent damage. After setup, double-check all connections before turning of the module to ensure they match the configuration described in this guide.

6.3 Counter Measurement Operations

The JY-7131 has eight identical 32-bit channels of timer/counter as shown in Figure 11.

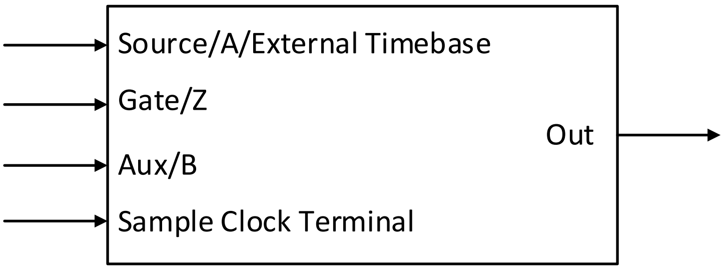


Figure 11 Counter Input Diagram

Each counter has eight input terminals and one output terminal, and these terminals have different functions in different counter measurement application described below:

- Edge Counting
- Pulse Measurement
- Frequency Measurement
- Period Measurement
- Two-Edge Separation Measurement
- 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 more information about sample clock, please refer to chapter 6.5.2.

6.3.1 Edge Counting

The counter counts the number of active edges of input signal. Default, the input signal must be connected to Counter Source terminal.

Set JY7131CITask.Type to CIType.EdgeCounting to use this function.

Timing

1) Single Mode

The counting value is written to the register on each rising edge or falling edge of the measured signal as shown in Figure 12.

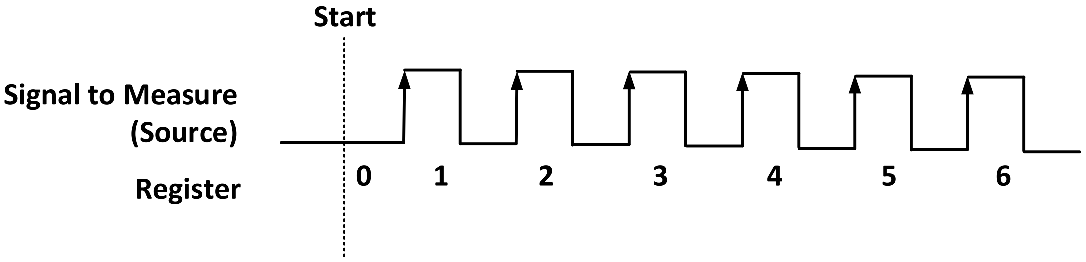


Figure 12 Simple Edge Counting in Single Mode

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Single.

2) Finite/Continuous Mode with Explicit Sample Clock

The counting value is stored into the buffer on each rising edge of the sample clock as shown in Figure 13.

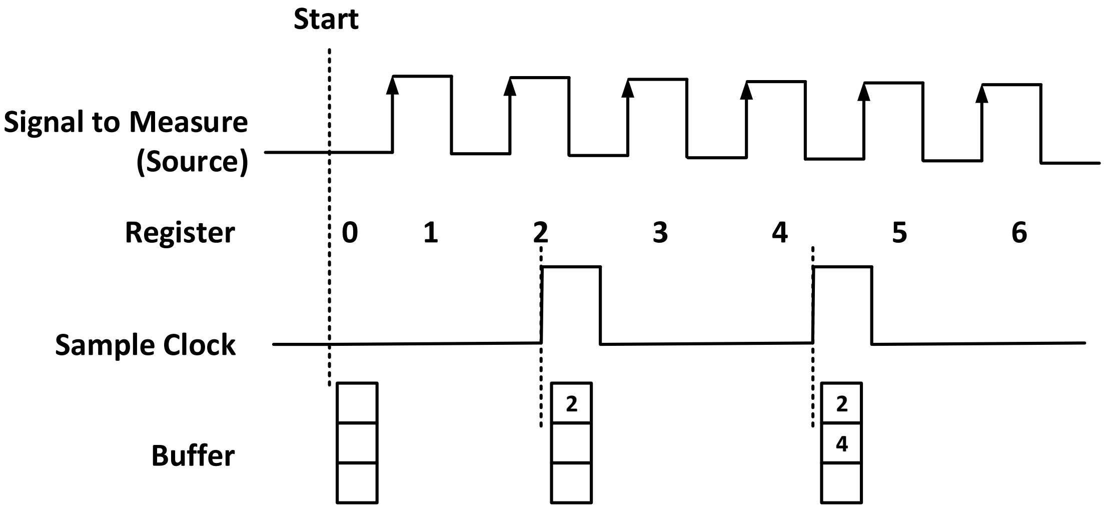


Figure 13 Buffered Edge Counting with Explicit Sample Clock

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Finite or CIMode.Continuous, and set JY7131CITask.SampleClock.Source to CISampleClockSource.Internal or CISampleClockSource.External.

Pause Trigger

Pause trigger is used to pause counting when the input signal is active depending on active polarity configuration as shown in Figure 14. Default, the Pause Trigger signal must be connected to Counter Gate terminal.

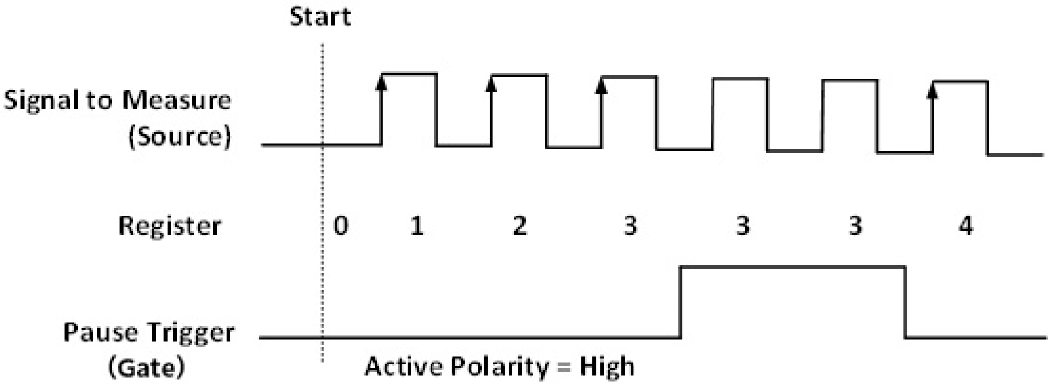


Figure 14 Pause Trigger

To cofigure the pause trigger, use the properties as below:

JY7131CITask.EdgeCounting.Pause.ActivePolarity – To set active level (high or low) to pause counting.

Count Direction

User can control the counting direction through software configuration or by an external input signal. Default, the external control direction signal must be connected to Counter Aux 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 15.

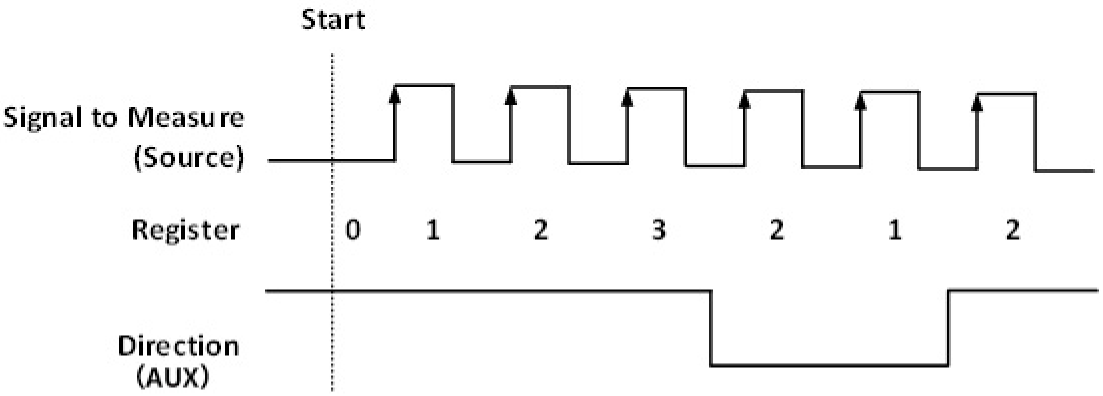


Figure 15 Count Direction

To cofigure the count direction, use the properties as belows:

JY7131CITask.EdgeCounting.Direction – To specify count up, count down, or controled by an external signal.

Exporting Count Event

When the counting value reaches the specified threshold, the counter will generate a pulse. To change the threshold, using following property:

JY7131CITask.EdgeCounting.OutEvent.Threshold

Terminals

To change the terminal of signals instead of using its default value as shown in chapter 3.2, using following properties:

JY7131CITask.EdgeCounting.InputTerminal – Signal-to-measure input terminal.

JY7131CITask.EdgeCounting.Pause.Terminal – Pause signal input terminal.

JY7131CITask.EdgeCounting.DirTerminal – External direction control signal input terminal.

JY7131CITask.EdgeCounting.OutEvent.Terminal – Count event output terminal.

6.3.2 Frequency Measurement

The counter measures the frequency of the signal. Default, the measured signal must be connected to Counter Gate terminal.

Set JY7131CITask.Type to CIType.Frequency to use this function.

Timing

1) Single Mode

Frequency Measurement without sample clock is actually using Pulse Width Measuement internally, refer to chapter 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 fomular and return the signal frequency to the user.

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

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Single.

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

Between every two rising edges of the sample clock, the counter counts the number of full periods ($T1$) of the signal, 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 16.

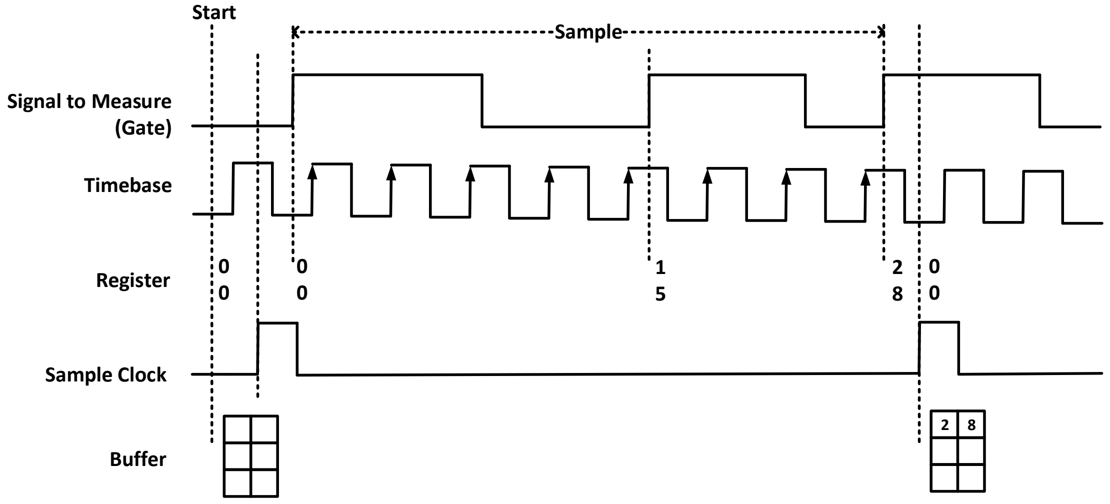


Figure 16 Frequency Measurement with Explicit Sample Clock

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 fomular and return the result to user.

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

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Finite or CIMode.Continuous, and set JY7131CITask.SampleClock.Source to CISampleClockSource.Internal or CISampleClockSource.External.

3) Finite/Continuous Mode with Implicit Sample Clock

Frequency Measurement with implicit sample clock is actually using Pulse Measuement internally. Refer to chapter   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 fomular and return the result to the user.

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

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Finite or CIMode.Continuous,　and set JY7131CITask.SampleClock.Source to CISampleClockSource.Implicit.

Timebase

By default, the counter uses the onboard 200MHz timebase to measure pulses. Use the property JY7131CITask.FrequencyMeas.Timebase to configure the timebase.

Please refer to chapter 6.5.3 for more information about timebase.

Terminals

To change the terminal of signals instead of using its default value shown in chapter 3.2, using following properties:

JY7131CITask.FrequencyMeas.InputTerminal – Signal-to-measure input terminal.

JY7131CITask. FrequencyMeas.Timebase.External.Terminal – External timebase input terminal.

6.3.3 Period Measurement

The counter measures the period of the signal. Default, the signal must be connected to Counter Gate terminal.

Set JY7131CITask.Type to CIType.Period to use this function.

Period Measurements is using Frequency Measurement internally and returns the reciprocal of Frequency Measuremnt. Refer to chapter for more information.

6.3.4 Two-Edge Separation

The counter measures the speration (interval between each rising edges of two signals) between the rising edges of two signals. Default, the first signal must be connected to Counter Gate terminal and the second signal must be connect to Counter Aux terminal.

Set JY7131CITask.Type to CIType.TwoEdgeSeparation to use this function.

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 scecond 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 resgiter on each rising edge of the first signal.

As shown in Figure 17.

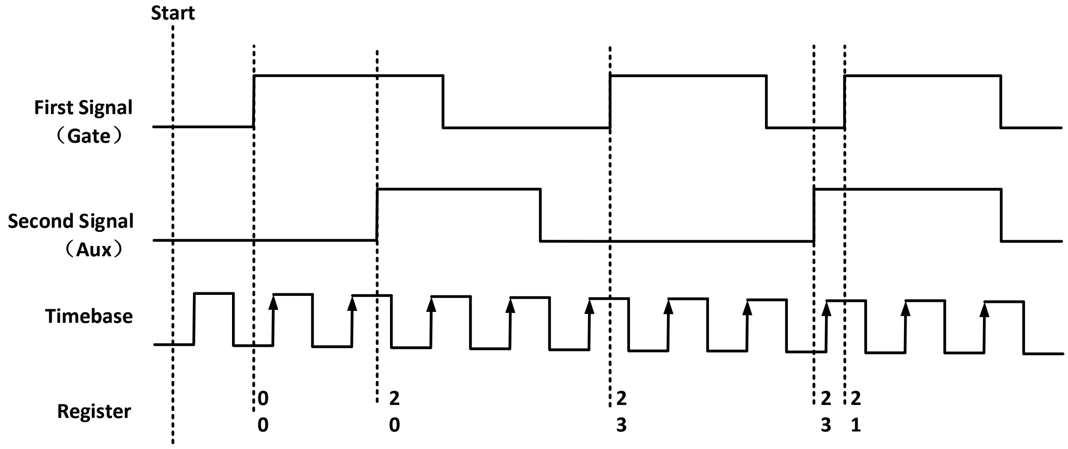


Figure 17 Two-Edge Separation in Single Mode

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Single.

2) Finite/Continuous Mode with Explicit Sample Clock:

The counting 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 18.

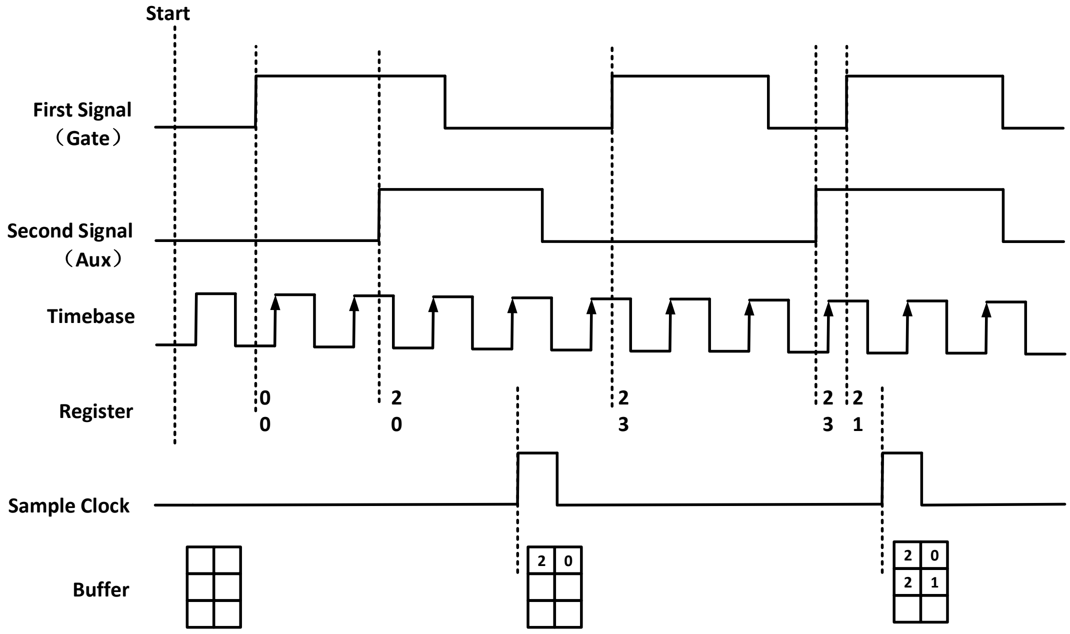


Figure 18 Two-Edge Seperation with Explicit Sample Clock

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Finite or CIMode.Continuous, and set JY7131CITask.SampleClock.Source to CISampleClockSource.Internal or CISampleClockSource.External.

3) Finite/Continuous Mode with Implicit Sample Clock

In implicit mode, the signal active edge as the implicit sample clock edge. The counting 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 19.

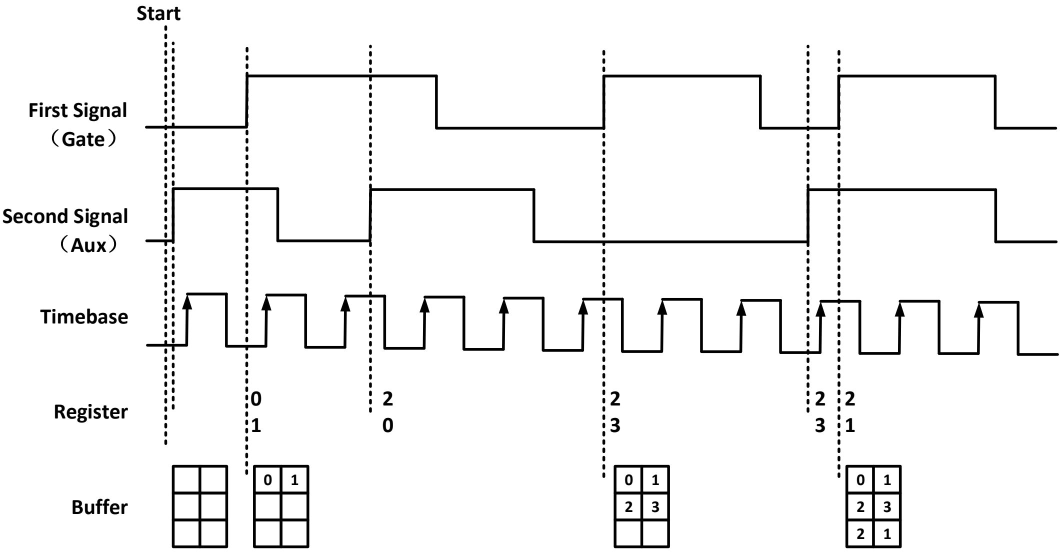


Figure 19 Two-Edge Seperation with Implicit Sample Clock

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Finite or CIMode.Continuous， and set JY7131CITask.SampleClock.Source to CISampleClockSource.Implicit.

Timebase

By default, the counter uses the onboard 200MHz timebase to measure pulses. Use the property JY7131CITask.TwoEdgeSeparation.Timebase to configure the timebase.

Please refer to chapter 6.5.3 for more information about timebase.

Terminals

To change the terminal of signals instead of using its default value shown in chapter 3.2, using following properties:

JY7131CITask.TwoEdgeSeparation.FirstInputTerminal – First signal-to-measure input terminal.

JY7131CITask.TwoEdgeSeparation.SecondInputTerminal – First signal-to-measure input terminal.

JY7131CITask.TwoEdgeSeparation.Timebase.External.Terminal – External timebase input terminal.

6.3.5 Quadrature Encoder

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

Set JY7131CITask.Type to CIType. QuadEncoder to use this function, and use property JY7131CITask.QuadEncoder.EncodingType to change the type of encoding. Default, the A signal must be connected to Counter A terminal, the B signal mest be connected to Counter B terminal and the Z signal mest be connected to Counter Z terminal. For terminal Z, you can connect Z signal to it, or you can disable Z with property "ZReloadEnabled".

Encoding Type

1) x1 Encoding

When A signal leads B signal, the counter increases the count value on the rising edge of A signal; when B signal leads A signal, the counter decreases the count value on the falling edge of A signal as shown in Figure 20.

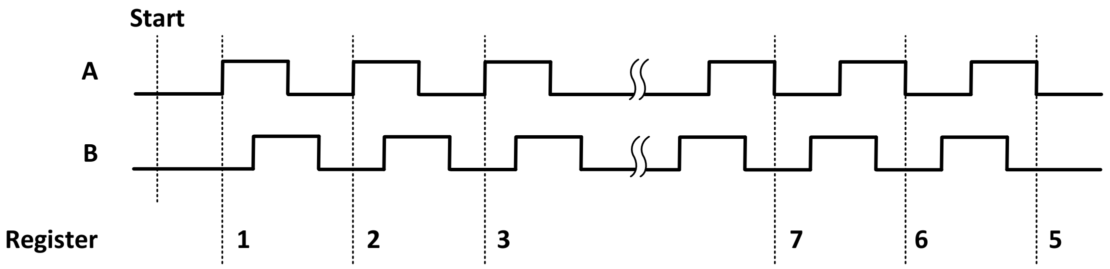


Figure 20 Quadrature Endcoder x1 Mode

2) x2 Encoding

When A signal leads B signal, the counter increases the count on the rising edge and the falling edge of A signal; when B signal leads A signal, the counter decreases the count value the rising edge and falling edge of A signal as shown in Figure 21.

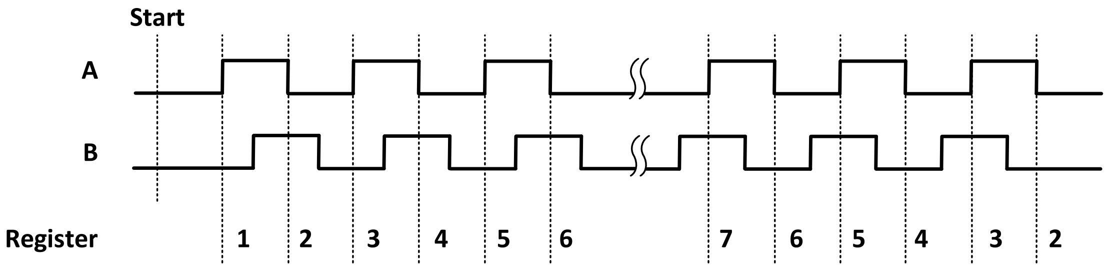


Figure 21 Quadrature Encoder x2 Mode

3) x4 Encoding

When A signal leads B signal, the counter increases the count value on the rising and falling edges of A signal and B signal. When B signal leads A signal, the counter decreases the count value on the rising and falling edges of A signal and B signal. As shown in Figure 22.

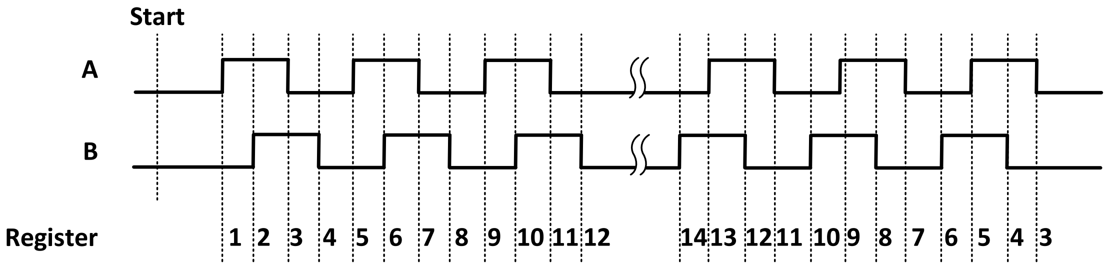


Figure 22 Quadrature Encoder x4 mode

Channel Z Behavior

The reload phase is when Z signal is high and A signal and B signal 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 A signal, as shown in Figure 20.

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Single.

2) Finite/Continuous Mode with Explicit Sample Clock

The count value is stored into the buffer on each rising edge of the sample clock, as shown in Figure 23.

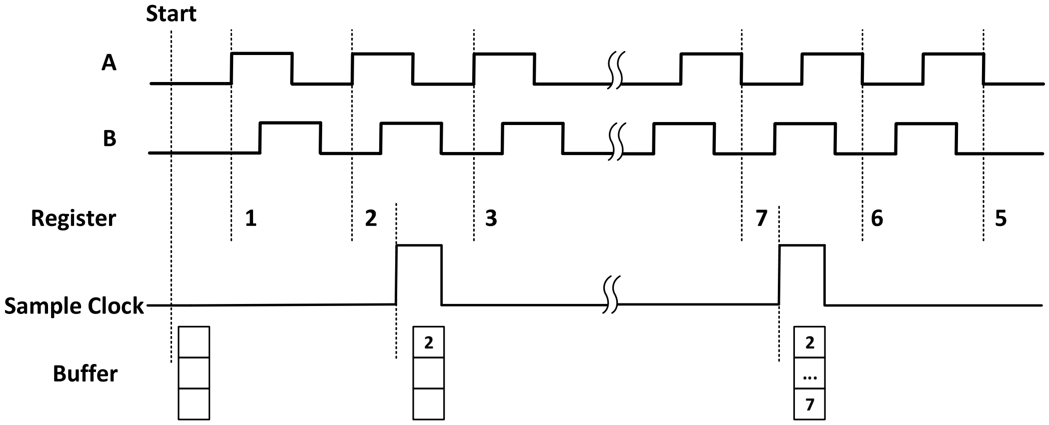


Figure 23 Quadrature Encoder x4 with Explicit Sample Clock

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Finite or CIMode.Continuous， and set JY7131CITask.SampleClock.Source to CISampleClockSource.Internal or CISampleClockSource.External.

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Finite or CIMode.Continuous， and set JY7131CITask.SampleClock.Source to CISampleClockSource.Implicit.

Terminals

To change the terminal of signals instead of using its default value as shown in chapter 3.2, use following properties:

JY7131CITask.QuadEncoder.AInputTerminal – Signal A input terminal.

JY7131CITask.QuadEncoder.ZInputTerminal – Signal Z input terminal.

JY7131CITask.QuadEncoder.BInputTerminal – Signal B input terminal.

6.3.6 Two-Pulse Encoder

The count value increases on the rising edge of A signal and decreases on the rising edge of B signal. Default, the A signal must be connected to Counter A terminal, the B signal mest be connected to Counter B terminal.

Set JY7131CITask.Type to CIType.TwoPulseEncoder to use this function

Timing

1) Single Mode

The count value is written to the register on each rising edge of the A signal, and B signel, as shown in Figure 24.

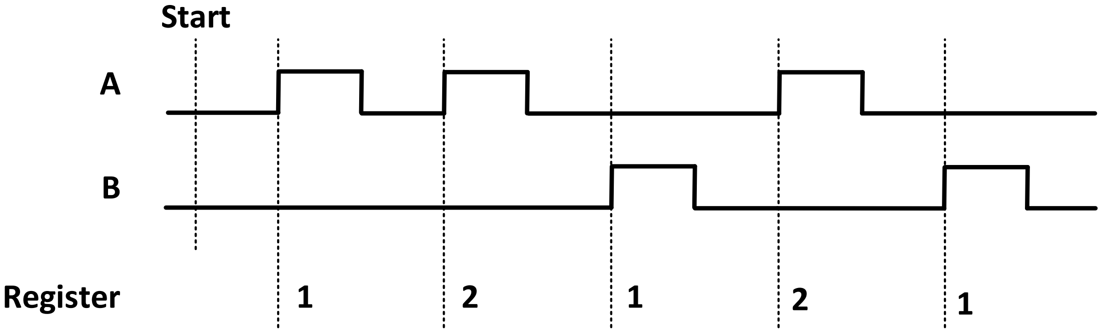


Figure 24 Two-Pulse Encoder in Single Mode

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Single.

2) Finite/Continuous Mode with Explicit Sample Clock

The count value is stored into the buffer on each rising edge of the sample clock, as shown in Figure 25.

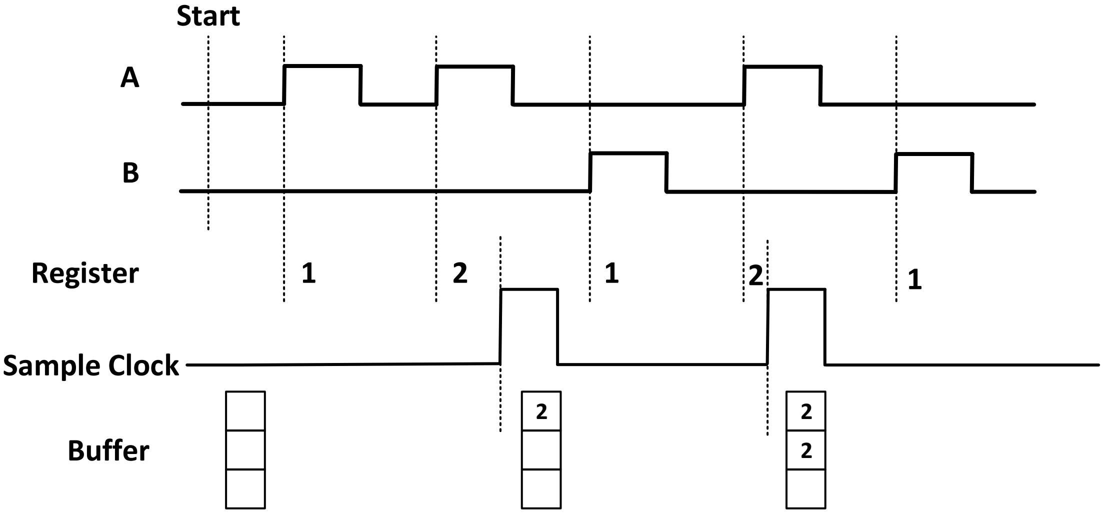


Figure 25 Two-Pulse Encoder with Explicit Sample Clock

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Finite or CIMode.Continuous,　and set JY7131CITask.SampleClock.Source to CISampleClockSource.Internal or CISampleClockSource.External.

To configure the counter to work in this mode, set JY7131CITask.Mode to CIMode.Finite or CIMode.Continuous,　and set JY7131CITask.SampleClock.Source to CISampleClockSource.Implicit.

Terminals

To change the terminal of signals instead of using its default value as shown in chapter 3.2, use following properties:

JY7131CITask.TwoPulseEncoder.AInputTerminal – Signal A input terminal.

JY7131CITask. TwoPulseEncoder.BInputTerminal – Signal B input terminal.

6.4 Counter Generation Operations

The JY-7131 can generate multiple forms of output signals according to different timing modes, including:

- Pulse generation with dynamic update
- Buffered pulse sequence generation

Timing

1) Single Mode

7131 can output a single pulse with a specified pulse configuration. The timing diagram of the pulse output is shown in Figure 26.

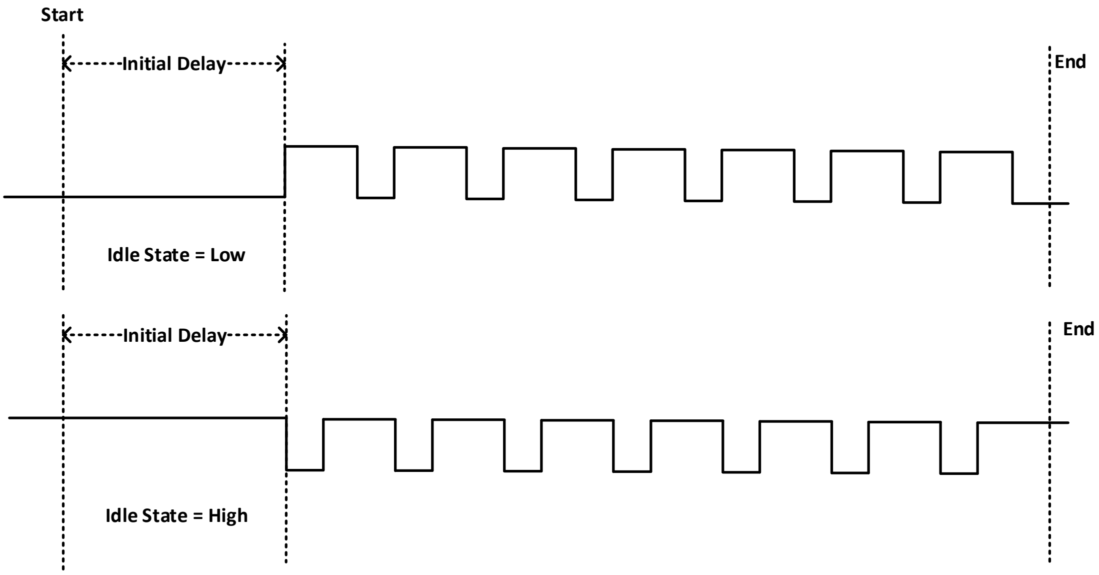


Figure 26 Pulse Output in Single Mode

Dynamic Update

If the number of pulses is set to -1, the pulses will be output continuously until requesting to stop. In this case, it is allowed to change the frequency and duty cycle on the fly. The timing diagram is shown as Figure 27.

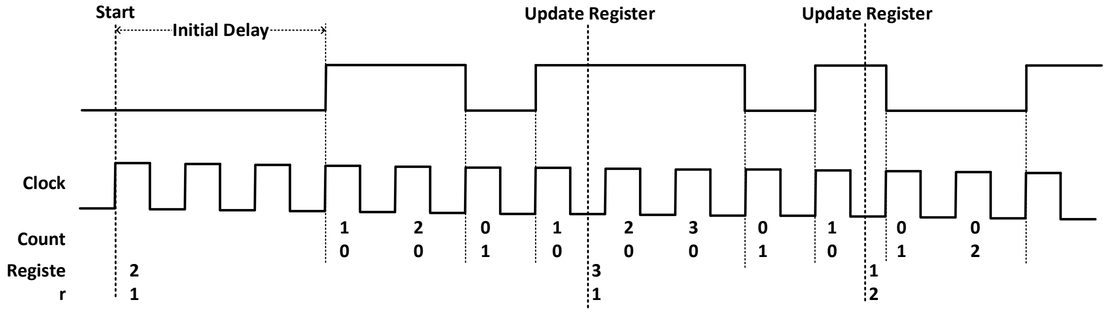


Figure 27 Pulse Output in Single Mode with Dynamic Update

2) Finite/Continuous Mode

This mode allows the user to write all configurations of pulses to be output to the buffer in advance. After send the currently configured pulses, the counter will automatically read the configuration of the next set of pulses to be sent from the buffer and start output. The timing diagram is shown in Figure 28.

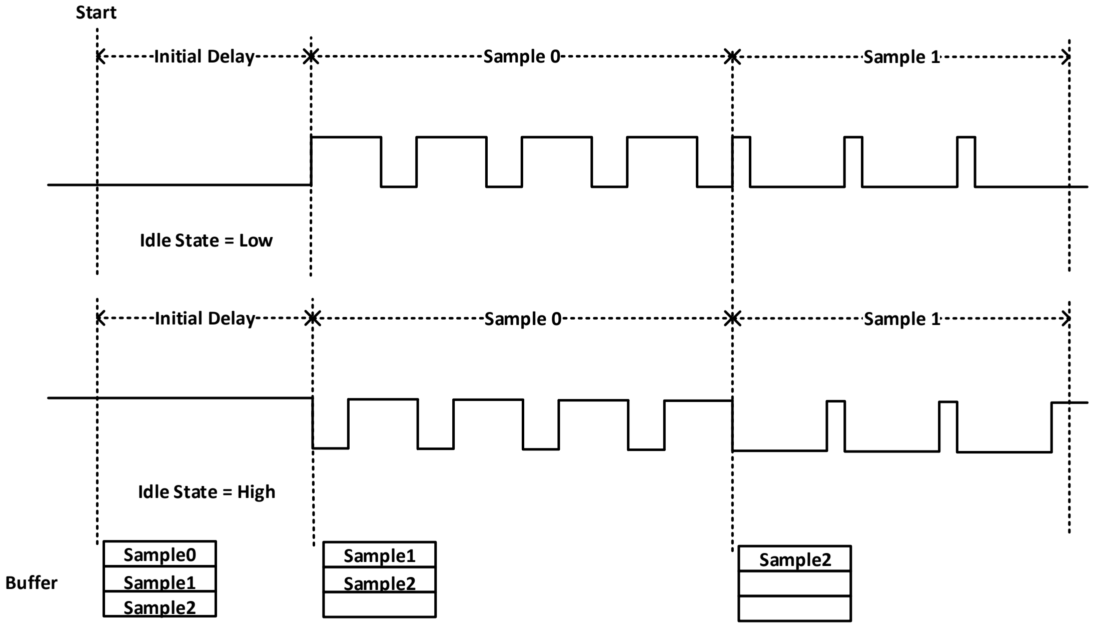


Figure 28 Buffered Pulse Sequence Generation

Each set of pulses can be configured in different ways:

- Frequency, Duty cycle and Number of pulses
- High time, Low time and Number of pulses
- High ticks, Low ticks and Number of pulses

Timebase

By default, the counter uses the onboard 200MHz timebase to generate pulses. Use the property JY7131COTask.Timebase to configure the timebase.

Please refer to chapter 6.5.3 for more information about timebase.

Terminals

To change the terminal of signals instead of using its default value as shown in chapter 3.2, using following properties:

- JY7131COTask.OutputTerminal – Signal output terminal.

6.5 Clocks

Figure 29 shows the structure of counter clock system.

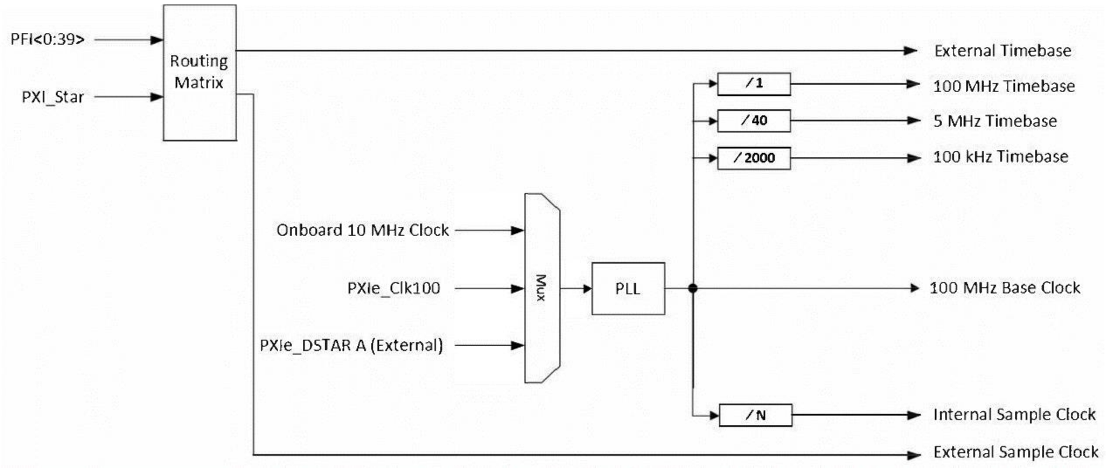


Figure 29 Clocks Diagram

6.5.1 PLL

PLL (Phase Locked Loop) is a phase-locked clock generator that can generate a clock signal of a specified frequency according to the selected reference clock source.

JY-7131 Series boards support the following reference clock source:

1) Onboard 10MHz Clock

Using the on-board 10MHz (TXCO) as the PLL input source can help improve the PLL output clock performance, including improving clock accuracy, temperature stability, and phase noise.

2) PXIe_CLK100

The PXIe_CLK00 siganal is a 100MHz clock provided by the PXIe backplane for every peripheral slot. When using this clock, PXIe-7131 can provide multi-card synchronization.

3) External Reference Clock

An external reference clock is a clock provided by user through terminal PXIe-DSTARA. To use an external reference clock, the user needs to specify its frequency.

By defaut, onboard 10MHz Clock is selected as the reference clock source. To change the reference clock source, configure the device as follows:

Set property Task.Device.ReferenceClock.Source to target reference clock source. Task is user defined JY7131CITask or JY7131COTask.

If the target clock source is External:

- Set Task.Device.ReferenceClock.External.Terminal to dersired terminal.
- Set Task.Device.ReferenceClock.External.Frequency to the frequency of this clock source.

Call method Device.Commit() to activate the configuration.

Note:

1. Clock configuration are not allowed to change while any counter tasks are running.

Clock configuration is applied to all tasks (including DI, DO, CI, CO).

The JY-7131 module must be powered off and restarted if the user submits the incorrect clock frequency by using external clock source (PXIe_DSTARA).

6.5.2 Sample Clock

For all counter measurement applications with buffered measurement, JY-7131 provides 3 sample clock options as follows:

Internal

The internal sample clock is generated by dividing down the 100 MHz base clock, and can be set independently for each counter.

To use the internal sample clock, configure as follows:

1. Set JY7131CITask.SampleClock.Source to CISampleClockSource.Internal
2. Set JY7131CITask.SampleClock.Internal.Rate to specify sample rate

External

External sample clock refers to an external signal input from a terminal as the sample clock

To use the external sample clock, configure as follows:

1. Set JY7131CITask.SampleClock.Source to CISampleClockSource.External
2. Set JY7131CITask.SampleClock.External.ExpectedRate to the rate of the external sample rate.

This property's value helps the driver determine a more suitable DDR writing frequency.

Can be set to an approximate value (sometimes the external sample clock may not have a fixed frequency). But cannot be less than the actual sampling rate, otherwise it will cause DDR write exception.

The default value of this property is -1, which means that the driver will be set with the safest value.

Implicit

Using an implicit sampling clock means the counter will send data to the buffer whenever there is a new measurement or count value.

To use the implicit sample clock, configure as follows:

1. Set JY7131CITask.SampleClock.Source to CISampleClockSource.Implicit.
2. Set JY7131CITask.SampleClock.Implicit.ExpectedRate to desired sample rate. This property has the same effect as JY7131CITask.SampleClock.External.ExpectedRate.

6.5.3 Timebase

JY7131 provides four options for timebase source as follows:

Internal 200MHz: - Same signal as the 100 MHz base clock generated by PLL.

Internal 5MHz – Generated by dividing down the 100 MHz timebase.

Internal 100kHz – Generated by dividing down the 100 MHz timebase.

External - Use a signal on a terminal as the timebase

6.6 Start Trigger

For all counter measurement and generation applications, the task starts running when the start trigger happens.

Start trigger has the following types:

Immediately

The task will start immediately after JY7131xxTask.Start() is called.

Software

After calling JY7131xxTask.Start () on the software, the task will not start until a software trigger is received.

Digital

An external digital trigger is generated when the external trigger source terminal detects a rising edge as shown in Figure 30.

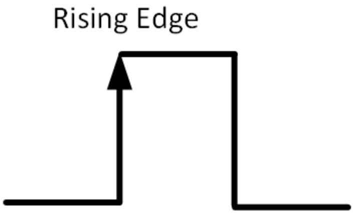


Figure 30 Rising Edge Digital Trigger

6.7 Multi-Card Synchronization

JY-7131 Series modules support master-slave synchronization mechanism to achieve multi-card synchronous acquisition.

Master-Slave Synchronization

It will use 3 signals, Reference clock, Sync Pulse and Triger to achieve data acqusistion simultaneously with multiple modules. First, the master module will notice all of slave modules by routing the trigger signal through PXI trigger bus, PXI_TRIG<0..7> when master module receives trigger. Second, we also need to make sure every module to start the acquisition task in the same time, therefore we could take advange of PXI system which can provide a synchronization pulse, PXIe_SYNC100 to coordinate with the acquisition task of multiple modules.

Third, every module must use the same reference clock to keep pace with each others and user can use PXIe_CLK100 which provides by PXI system as reference clock.

The timing diagram is shown as Figure 31.

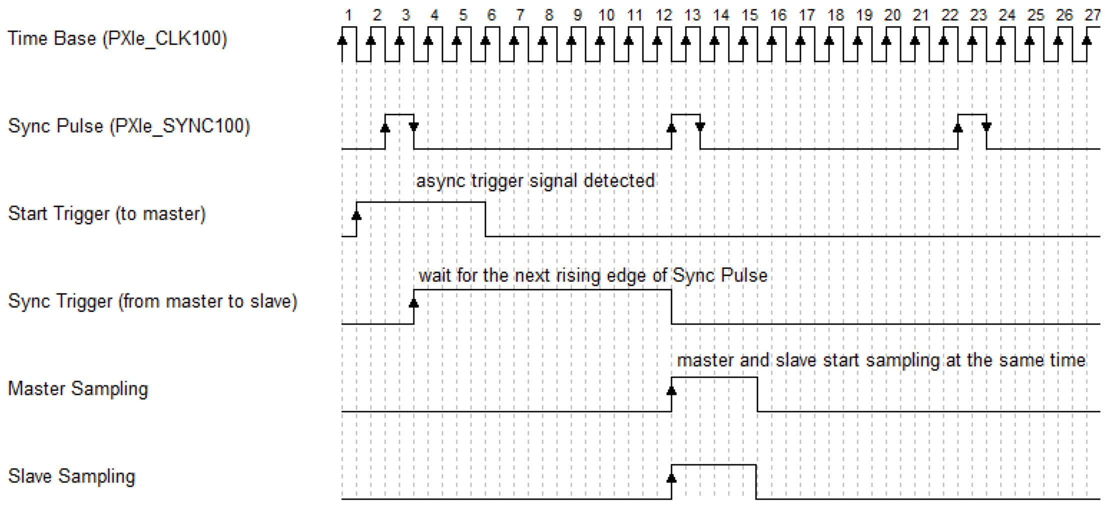

Time Base (PXIe_CLK100)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27
Sync Pulse (PXIe_SYNC100)
async trigger signal detected
Start Trigger (to master)
wait for the next rising edge of Sync Pulse
Sync Trigger (from master to slave)
master and slave start sampling at the same time
Master Sampling
Slave Sampling

Figure 31 Master-Slave Synchronization

To enable the multi-card synchronization, configure the board as follows:

1. Set each task on different card as Master or Slave. Only one master is allowed.
2. Set the reference clock source of master card and slave card to PXIe_CLK100. Refer to 6.5.1 for more information.
3. Route trigger signals of all tasks to the same signal terminal, trigger signal will be sent from the master card to all slave cards through this terminal.
4. Route Sync Pulse signals of all tasks to the same signal terminal, Sync Pulse signals will be sent from the master card to all slave cards through this terminal.
5. Start all slave tasks.
6. Start the master task (if digital trigger or software trigger is enabled, you need to wait for the trigger signal to arrive), all the tasks will start to work synchronously on a rising edge of the PXIe_SYNC100 signal.

6.8 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.

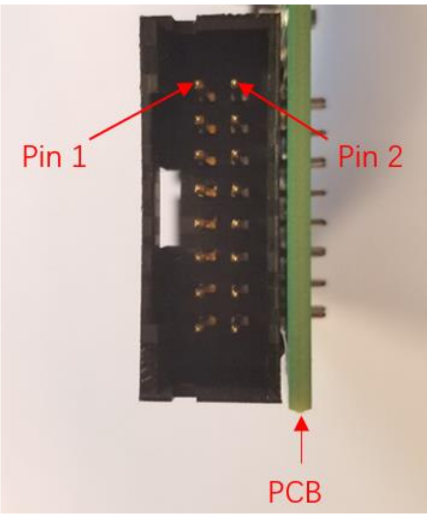


Figure 32 SSI Connector in PCIe-7131

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 15 SSI Connector Pin Assignment for PCIe-7131

6.9 DIP Switch in PCIe-7131

PCIe-7131 series modules have a DIP switch. The card number can be adjusted manually by turning the DIP switch, which is used to identify the module with different slot positions.

For example, if you want to set the card number to 3, you could turn the position 2 and 1 of the DIP switch to the ON position and the orthers to OFF. Find the detail below.

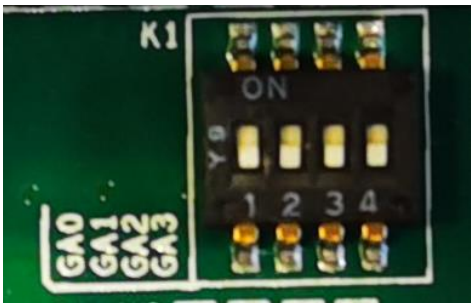


Figure 33 DIP Switch in PCIe-7131

	Position 4 (GA3)	Position 3 (GA2)	Position 2 (GA1)	Position 1 (GA0)
Slot 0	0	0	0	0
Slot 1	0	0	0	1
Slot 2	0	0	1	0
Slot 3	0	0	1	1
Slot 4	0	1	0	0
Slot 5	0	1	0	1
Slot 6	0	1	1	0
Slot 7	0	1	1	1
Slot 8	1	0	0	0
Slot 9	1	0	0	1
Slot 10	1	0	1	0
Slot 11	1	0	1	1
Slot 12	1	1	0	0
Slot 13	1	1	0	1
Slot 14	1	1	1	0
Slot 15	1	1	1	1
Note: OFF=0/ ON=1				

Table 16 Relationship between switch position and slot number

7. Using JY-7131 in Other Software

While JYTEK's defalt application platform is Visual Studio, the programming language is C#, we recognize there are other platforms that are either becoming very popular or have been widely used in the data acquisition applications. Among them are C++ etc. This chapter explains how you can use JY-7131 DAQ card using one of this software.

7.1 C++

JYTEK internaly uses our C++drivers to design the C# drivers. We recommend our customers to use C# drivers because C# platform deliver much better efficiency and performance in most situations. We also make our C++ drivers available. However, due to the limit of our resources, we do not actively support C++ drivers. If you want to be our partner to support C++ drivers, please contact us.

8. About JYTEK

8.1 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 is a joint venture between Adlink Technologies and a group of experienced professionals form the industry. JYTEK independenly develop the software and hardware products and is entirely focused on the Chinese market. Our Shanghai headquarters and production service center have regular stocks to ensure timely supply; we have R&D centers in Xi'an and Chongqing to develop new products; we also have highly trained direct technical sales representatives in Shanghai, Beijing, Tianjin, Xi'an, Chengdu, Nanjing, Wuhan, Haerbin, and Changchun. We also have many patners who provide system level support in various cities.

8.2 JYTEK Hardware Products

According to JYTEK's agreement with our equity partner Adlink Technologies, JYTEK's hardware is manufactured by the state-of-art manufacturing facility located in Shanghai Zhangjiang Hi-Tech Park. Adlink has over 20 years of the world-class low-volumn and high-mix manufacturing expertise with ISO9001-2008, China 3C, UL, ROHS, TL9000, ISO-14001, ISO-13485 certifications. Its 30,000 square meters facilities and three high-speed Panasonic SMT production lines can produce 60,000 pieces boards/month; it also has full supply chain management - planning, sweeping, purchasing, warehousing and distribution. Adlink's manufacturing excellence ensures JYTEK's hardware has word-class manufacturing quality.

One core technical advantage is JYTEK's pursue for the basic and fundamental technology excellence. JYTEK China has developed a unique PCIe, PXIe, USB hardware driver architecture, FirmDrive, upon which many our future hardware will be based.

In addition to our own developed hardware, JYTEK also rebrands Adlink's PXI product lines. In addition, JYTEK has other rebranding agreements to increase our hardware coverage. It is our goal to provide the complete product coverage in PXI and PCI modular instrumentation and data acquisition.

8.3 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.

8.4 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.

9. 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 explaination for JYTEK JY-7131 Series family of multi-function data acquisition boards. 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