

I-9048 / I-8048W Module Linux API Reference

(For Linux platform)

V 1.0.2 April 2021



Written by Sean Hsu

Edited by Cindy Huang

Warranty

All products manufactured by ICP DAS are under warranty regarding defective materials for a period of one year, beginning from the date of delivery to the original purchaser.

Warning

ICP DAS assumes no liability for any damage resulting from the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, nor for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Copyright © 2020 by ICP DAS Co., Ltd. All rights are reserved.

Trademarks

Names are used for identification purposes only and may be registered trademarks of their respective companies.

Contact Us

If you have any problems, please feel free to contact us.

You can count on us for a quick response.

Email: service@icpdas.com

Table of Contents

Table of Contents.....	3
Preface.....	5
1. Introduction.....	6
1.1. I/O Module Dimensions	7
1.2. Inserting the I/O Modules	8
1.3. Wire Connections	10
1.4. Demo Programs.....	11
2. I-9048 / I-8048W module features.....	12
2.1. General Introduction	13
2.1.1. Pin Assignment for the I-9048.....	14
2.1.2. Pin Assignment for the I-8048W	15
2.1.3. Block Diagram.....	16
2.1.4. Isolated or TTL Input	18
2.1.5. Digital Input & LED indicators.....	20
2.1.6. Programmable Rising/Falling interrupt.....	21
2.2. Software Introduction	23
2.2.1. Software flow chart	24
2.2.2. Recognize different interrupt service requests	26
3. API References.....	28
3.1. Function List	29
3.2. I8048W_DI_ALL	30
3.3. I8048W_DI_Ch	31
3.4. I8048W_Set_RisingReg.....	32
3.5. I8048W_Set_FallingReg.....	33
3.6. I8048W_Read_RisingReg.....	34
3.7. I8048W_Read_FallingReg.....	35
3.8. I8048W_Read_RisingEvent	36
3.9. I8048W_Read_FallingEvent.....	37
3.10. I8048W_Read_RisingEventCount.....	38
3.11. I8048W_Read_FallingEventCount	39
3.12. I8048W_Clear_RisingEventCount	40
3.13. I8048W_Clear_FallingEventCount.....	41
3.14. I8048W_Init.....	42

3.15. I8048W_InstallISR	43
3.16. I8048W_UnInstallISR.....	45
3.17. I8048W_UnFreezeINT	46
3.18. I8048W_GetLibVersion.....	47
Revision History	48

Preface

The I-9048 / I-8048W is an 8-channel digital input module with hardware interrupt capability for real-time system application. Each channel can be programmed as an interrupt input channel to invoke an interrupt on falling edges, rising edges or both falling and rising edges. Each channel can be either Isolated or Non-isolated TTL Input, selectable by Jumper.

The module includes LED indicators are provided for monitoring DI channel status, together with ± 4 kV ESD protection and 1500 Vrms intra-module isolation.

The information contained in this manual is divided into the following topics:

- [Chapter 1, "Introduction"](#) – This chapter provides information related to the hardware, such as the specifications, the jumper settings details and wiring information.
- [Chapter 2, "I-9048 / I-8048W features"](#) — This chapter introduces the features of I-9048 / I-8048W module.
- [Chapter 3, "API References"](#) – This chapter describes the functions provided in the I-9048 / I-8048W library together with an explanation of the differences in the naming rules used for the different Windows platforms.

1. Introduction

The I-9K/I-8K series module is based on a parallel interface with high communication speed. The differences between the I-9K and I-8k series are listed as follows:

I/O module features comparison

Model	I-9K Series	I-8K series
Communication interface	Parallel bus	Parallel bus
Protocol	-	-
Communication speed	Fast	Fast
DI module with latched function	-	-
DI module with counter input	-	-
Power on value for DO module	Y	-
Safe value for DO module	Y	-
Programmable slew-rate for AO module	-	-

Refer to

<http://www.icpdas.com/en/product/guide+Remote+I+O+Module+and+Unit+%EF%BC%86+Local+I+O+Modules+I-9K+I-97K+Series> for more details regarding of I-9K series module specification, jumper settings details and wiring information.

Refer to

[http://www.icpdas.com/en/product/guide+Remote+I+O+Module+and+Unit+%EF%BC%86+Local+I+O+Modules+I-8K+I-87K+Series+\(High+Profile\)](http://www.icpdas.com/en/product/guide+Remote+I+O+Module+and+Unit+%EF%BC%86+Local+I+O+Modules+I-8K+I-87K+Series+(High+Profile)) for more details regarding of I-8K series module specification, jumper settings details and wiring information.

Those I-9K Modules must work then plugin any slot with the following PAC:

Platform	CPU	Slot Counts
WP-9x2x-CE7 / LP-9x2x	AM335x (ARM)	2,4,8
XP-9x7x-WES7 / XP-9x8x-WES7/ LX-9x71 / LX-9x81	E3827/E3845 (X86)	1,3,7

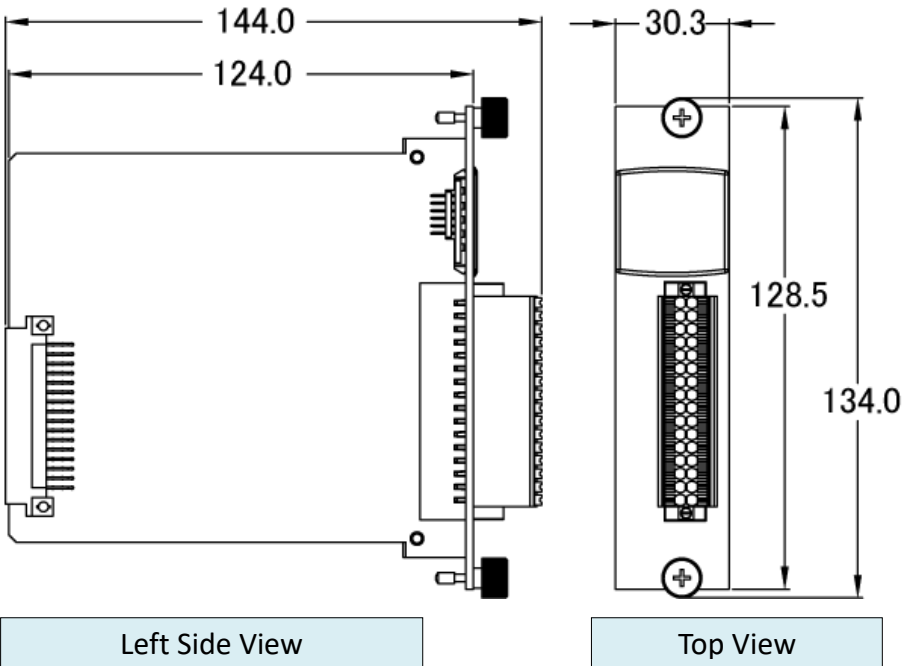
Those I-8K Modules must work then plugin any slot with the following PAC:

Platform	CPU	Slot Counts
WP-8x4x / LP-8x4x	PXA270 (ARM)	1,4,8
WP-8x2x-CE7 / LP-8x2x	AM335x (ARM)	1,4,8
XP-8x3x-CE6 / LX-8x3x	x86 CPU,1 GHZ, dual-core	1,3,7

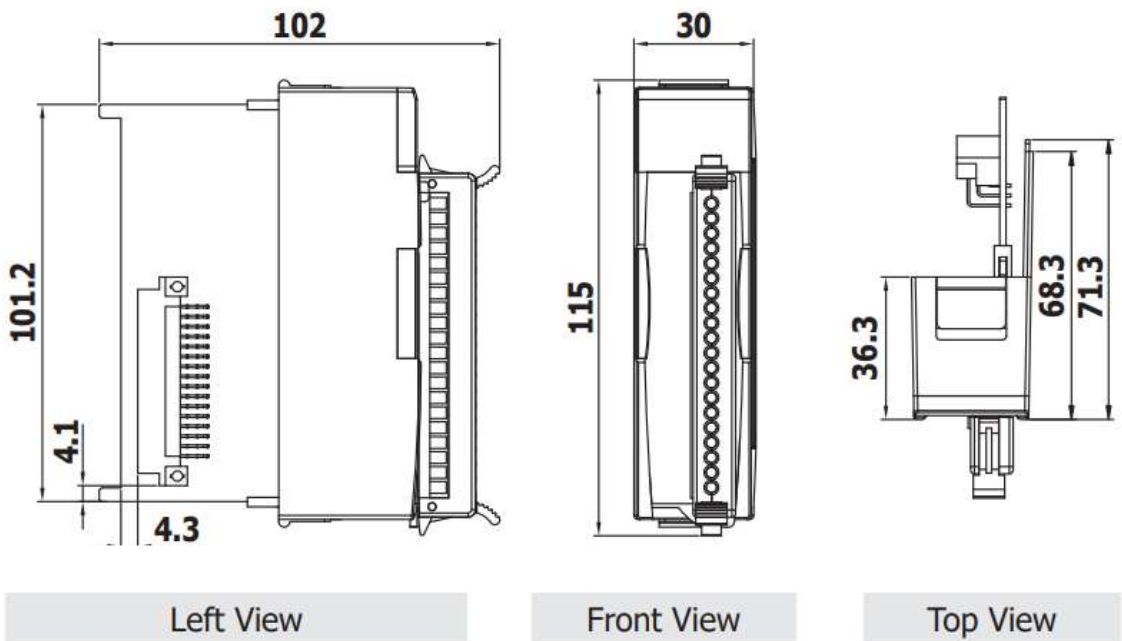
1.1. I/O Module Dimensions

All dimensions are in millimeters.

I-9K module with spring clamp terminal connector



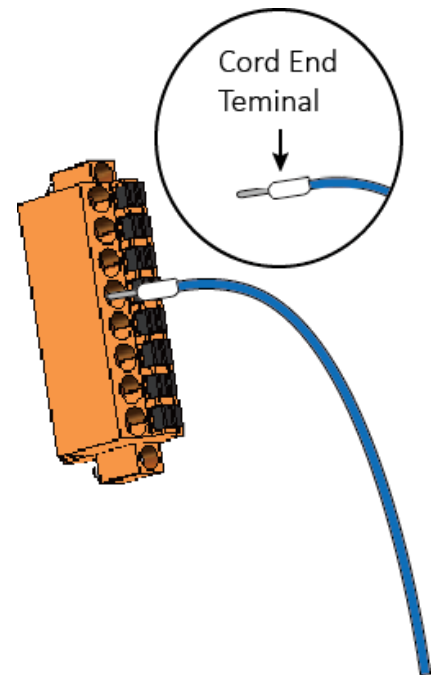
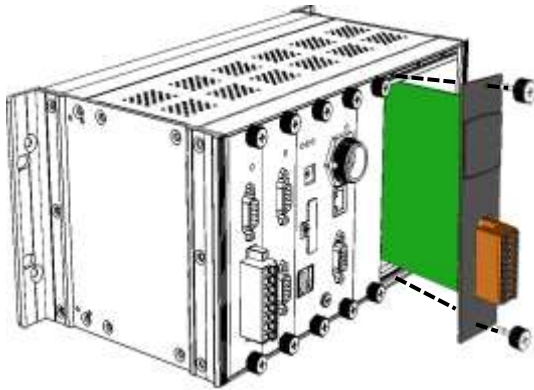
I-8K module with screw terminals connector



1.2. Inserting the I/O Modules

Follow the procedure described below to insert the I-9048 module.

1. Insert the I/O module



2. Wiring connection

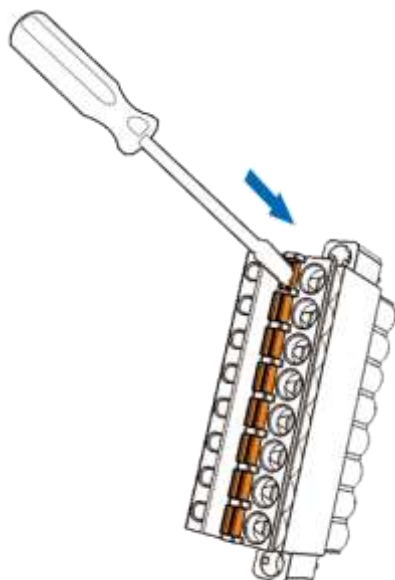
The metal part of the cord end terminal on the wire can be direct wired to the terminal.

Note:

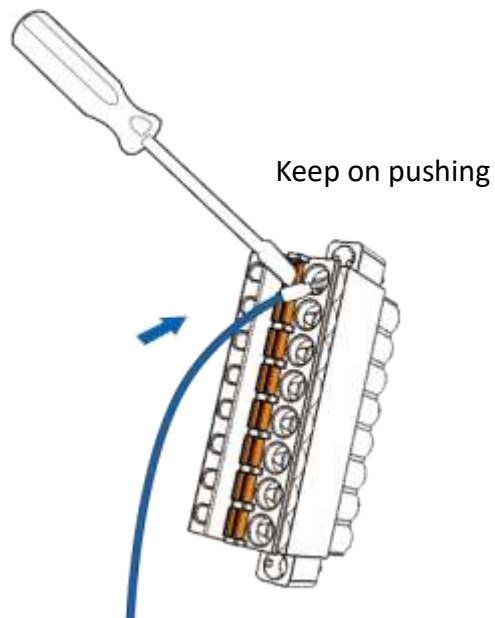
The I-9048 module supports spring clamp terminal connector. The spring clamp terminal connector for the I-9048 module connector offers the advantages (anti-vibration, stable clamping and installation easier) relative to screw terminals.

A tip on how to connect the wiring to the connector

1. Use screwdriver to push the orange clip in.

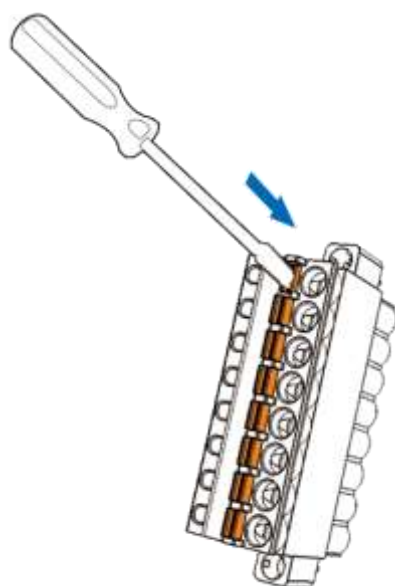


2. Insert the wiring into the terminal block

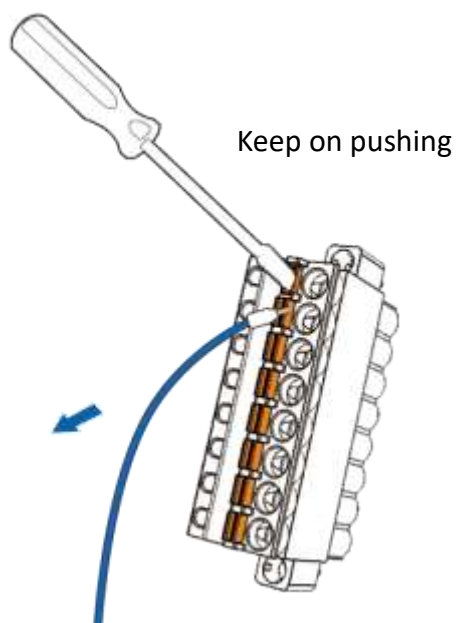


A tip on how to remove the wiring from the connector

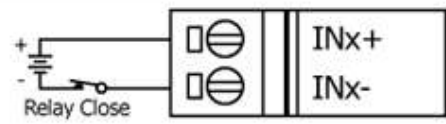
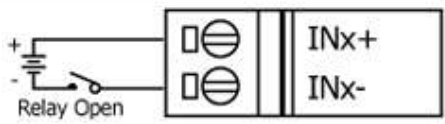
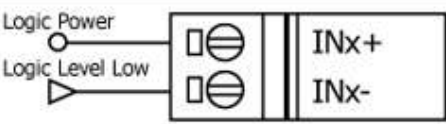
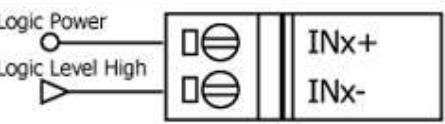
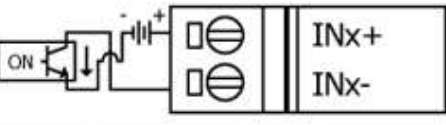
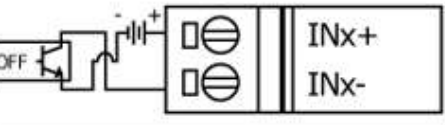
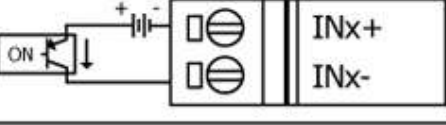
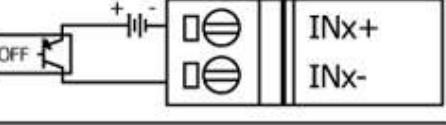
1. Use screwdriver to push the orange clip in.

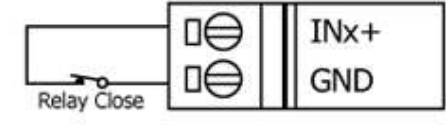
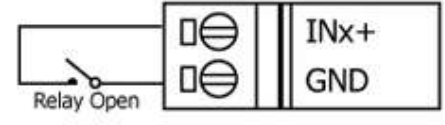
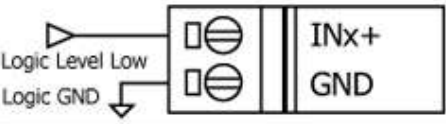
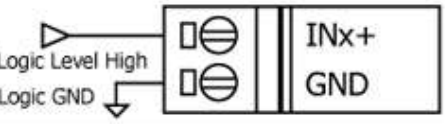
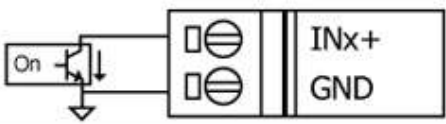
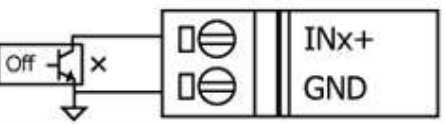


2. Remove the wiring from the terminal block



1.3. Wire Connections

Isolated		
Input Type	ON State LED ON Readback as 0	OFF State LED OFF Readback as 1
Relay Contact	Relay ON	Relay Off
		
TTL/CMOS Logic	Voltage > 4V	Voltage < 1V
		
NPN Output	Open Collector On	Open Collector Off
		
PNP Output	Open Collector On	Open Collector Off
		

Non-Isolated		
Input Type	ON State LED ON Readback as 0	OFF State LED OFF Readback as 1
Relay Contact	Relay ON	Relay Off
		
TTL/CMOS Logic	Voltage < 0.8V	Voltage > 2V
		
Open Collector	Open Collector On	Open Collector Off
		

1.4. Demo Programs

ICP DAS provides a range of demo programs for different platforms that can be used to verify the functions of the I-8048W/9048. The source code contained in these programs can also be reused in your own custom programs if needed.

Both I-8048W/9048 use the same library and demo.



We need to check the following steps before running the program.

1. First, user need to download LinPAC SDK, which includes GNU toolchain, Libraries, header, examples files, etc.
2. Check the power cable, Ethernet cable, VGA monitor, the communication cable between controller and PC has been connected well, and then check the I-8048W/I-9048 has been plugged in the controller.
3. Next, check the communication between controller and PC is fine, and download the demo program files to the controller.
4. The following is a list of the locations where both the demo programs and associated libraries can be found on either the ICP DAS web site, and I-8048W/I-9048 use the same library and demo.

User can find the related files in the below website:

PRRODUCT	CPU	DOWNLOAD LINK
LP-8x4x	PXA270	http://www.icpdas.com/en/download/show.php?num=982&model=LP-8441-EN
LP-8x2x/9x2x	AM335x	http://www.icpdas.com/en/download/show.php?num=915&model=LP-8421
LX-8000/9000	x86/E38xx	http://www.icpdas.com/en/download/show.php?num=904&model=LX-9381

2. I-9048 / I-8048W module features

The I-9048 / I-8048W is an 8-channel digital input module with hardware interrupt capability for real-time system application. Each channel can be programmed as an interrupt input channel to invoke an interrupt on falling edges, rising edges or both falling and rising edges. Each channel can be either Isolated or Non-isolated TTL Input, selectable by Jumper.

The DI module includes LED indicators are provided for monitoring DI channel status, together with ± 4 kV ESD protection and 1500 Vrms intra-module isolation.

2.1. General Introduction

The I-9048 / I-8048W module is an 8-channel digital input module designed for interrupt applications.

The key features of the I-9048 / I-8048W are as following:

Isolation

Intra-module Isolation	1500 Vrms
------------------------	-----------

EMS Protection

ESD (IEC 61000-4-2)	± 4 kV Contact for Each Terminal
---------------------	----------------------------------

LED Indicators

Status	1 x Power and 8 x DI
--------	----------------------

Digital Input

Channels	8
Type	Dry Contact, Wet Contact
Sink/Source (NPN/PNP)	Sink, Source
Wet Contact, ON Voltage Level	Isolated: +4 ~ +30 VDC Non-Isolated: 0 ~ +0.8 VDC
Wet Contact, OFF Voltage Level	Isolated: 0 ~ +1 VDC Non-Isolated: +2 ~ +5 VDC
Input Impedance	2.4 kΩ, 0.5 W

Digital Input/Counter

Interrupts	8 (DI0 ~ DI7) Trigger type: Rising/falling edge programmable for each channel Max. Interrupt frequency: 10 KHz Max.
------------	---

Power





















Consumption	1.75 W Max.
-------------	-------------

Environmental

Operating Temperature	-25 ~ +75 °C
Storage Temperature	-40 ~ +85 °C
Humidity	10 ~ 90% RH, Non-condensing

2.1.1. Pin Assignment for the I-9048





















The pin assignment for the I-9048 is shown as follows:

Pin Assignment		Terminal No.		Pin Assignment	
GND	01			11	GND
IN0+	02			12	IN0-
IN1+	03			13	IN1-
IN2+	04			14	IN2-
IN3+	05			15	IN3-
IN4+	06			16	IN4-
IN5+	07			17	IN5-
IN6+	08			18	IN6-
IN7+	09			19	IN7-
E5V	10			20	EGND

- Pin 1 and 11: TTL GND, ground for non-isolated input signals
- Pins 2 ~ 9, 12~19: 8-channel digital input
- Pins 10 and 20: Isolated power supply, 5V, 200mA max.

2.1.2. Pin Assignment for the I-8048W

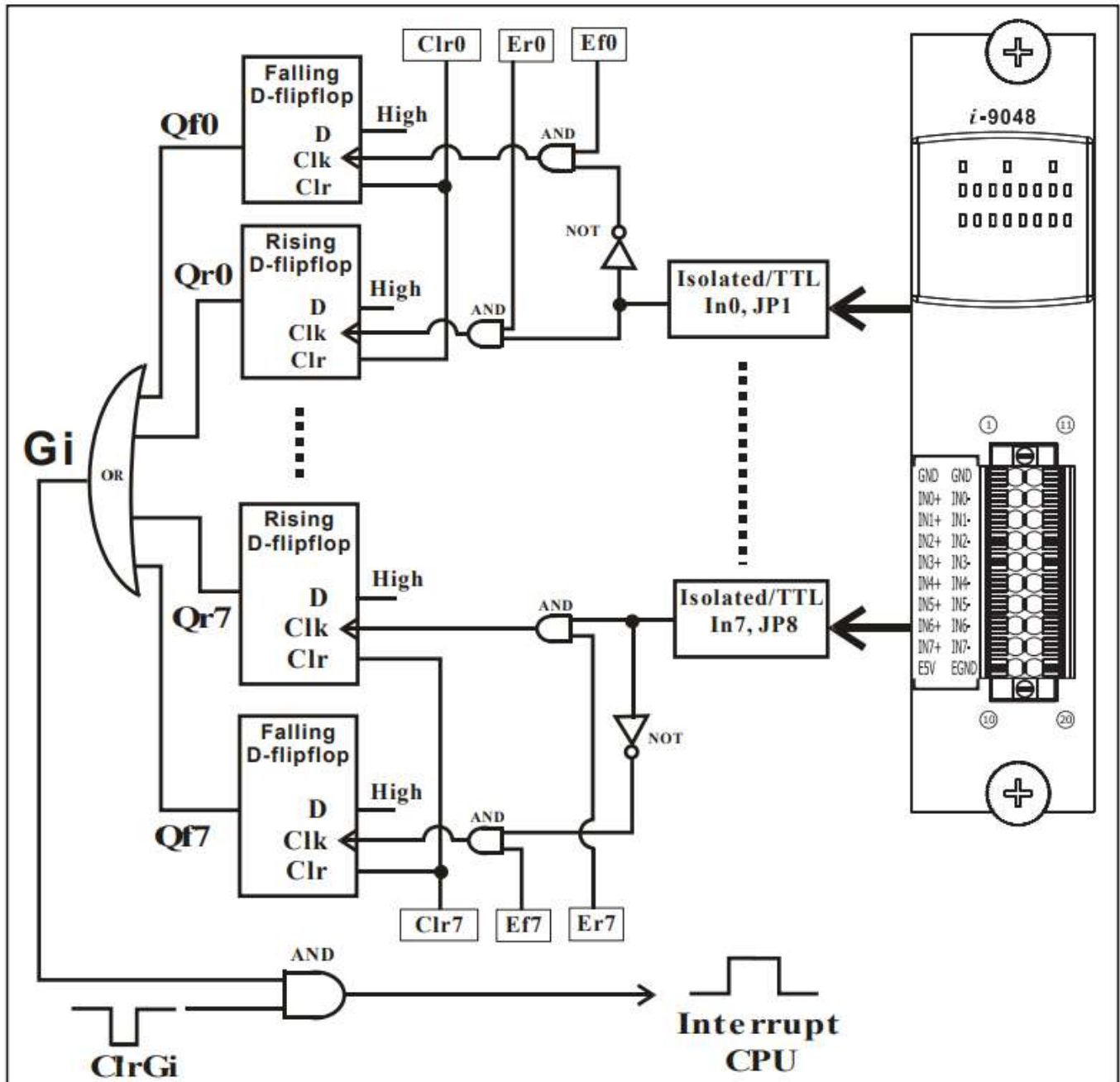
The pin assignment for the I-8048W is shown as follows:

Terminal No.		Pin Assignment
	01	GND
	02	-
	03	IN 0+
	04	IN 0-
	05	IN 1+
	06	IN 1-
	07	IN 2+
	08	IN 2-
	09	IN 3+
	10	IN 3-
	11	IN 4+
	12	IN 4-
	13	IN 5+
	14	IN 5-
	15	IN 6+
	16	IN 6-
	17	IN 7+
	18	IN 7-
	19	E5V
	20	EGND

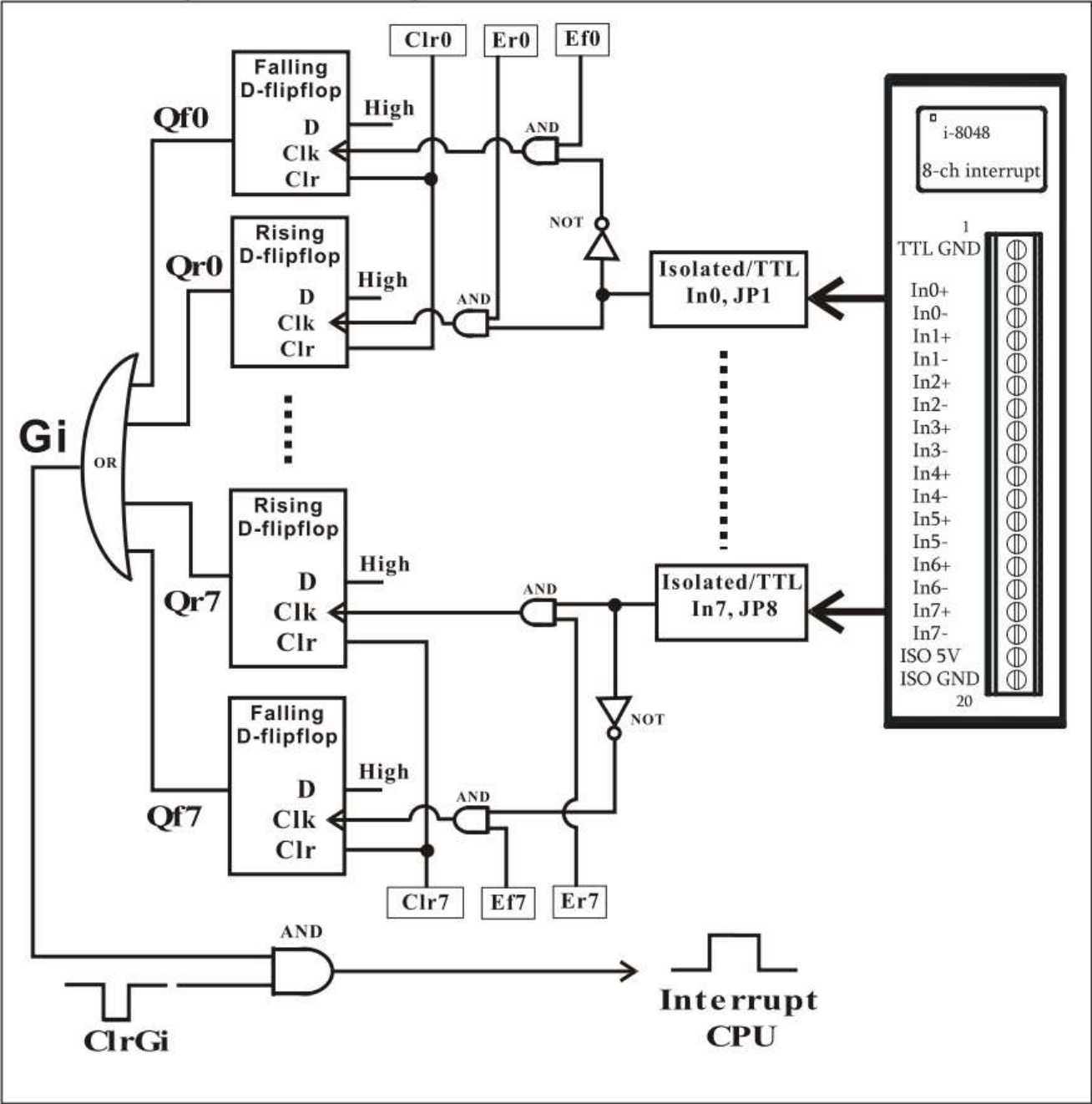
- Pin 1: TTL GND, ground for non-isolated input signals
- Pins 3 ~ 18: 8-channel digital input
- Pins 19 and 20: Isolated power supply, 5V, 200mA max.

2.1.3. Block Diagram

The signal flow block diagram for I-9048 is shown as follows:



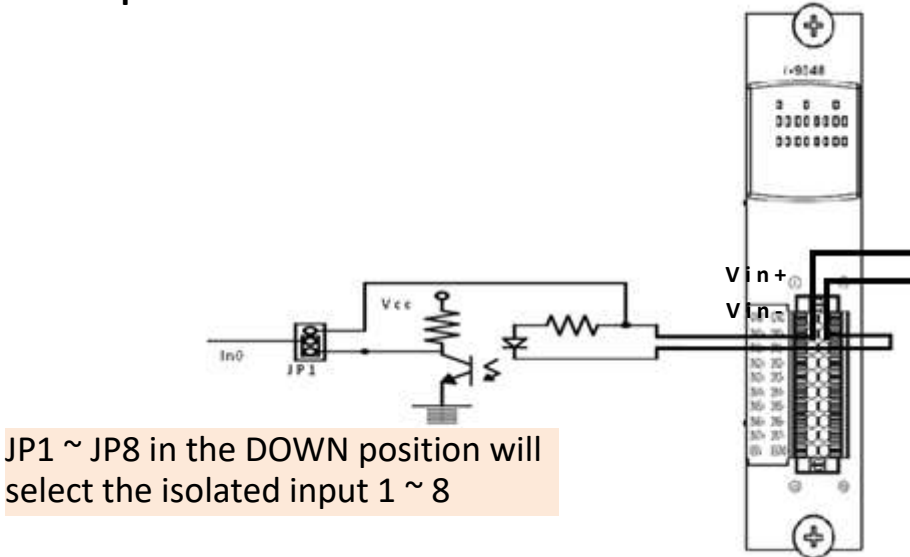
The signal flow block diagram for I-8048W is shown as follows:



2.1.4. Isolated or TTL Input

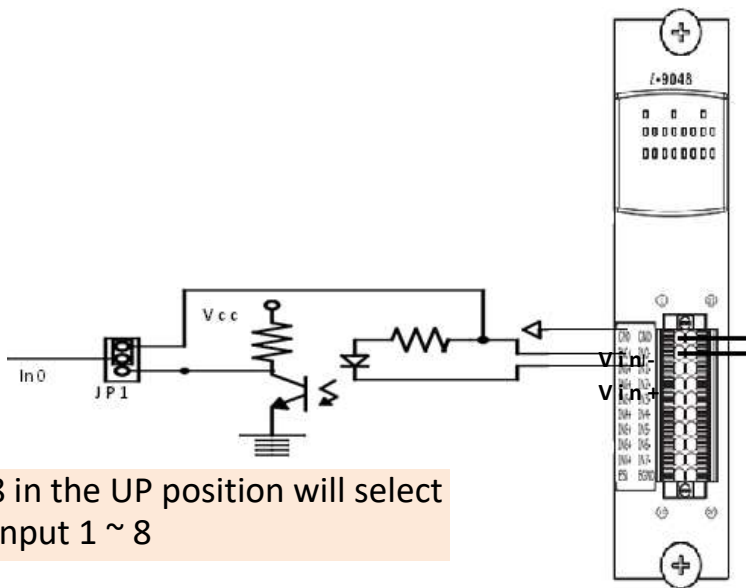
The input signal can be either isolated or TTL input for **I-9048** as follows:

Isolated input



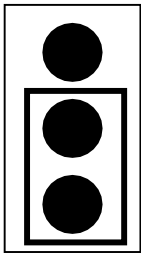
JP1 ~ JP8 in the DOWN position will select the isolated input 1 ~ 8

TTL input

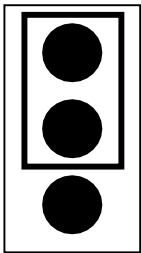


JP1 ~ JP8 in the UP position will select the TTL input 1 ~ 8

JP1	for ch0
JP2	for ch1
JP3	for ch2
JP4	for ch3
JP5	for ch4
JP6	for ch5
JP7	for ch6
JP8	for ch7



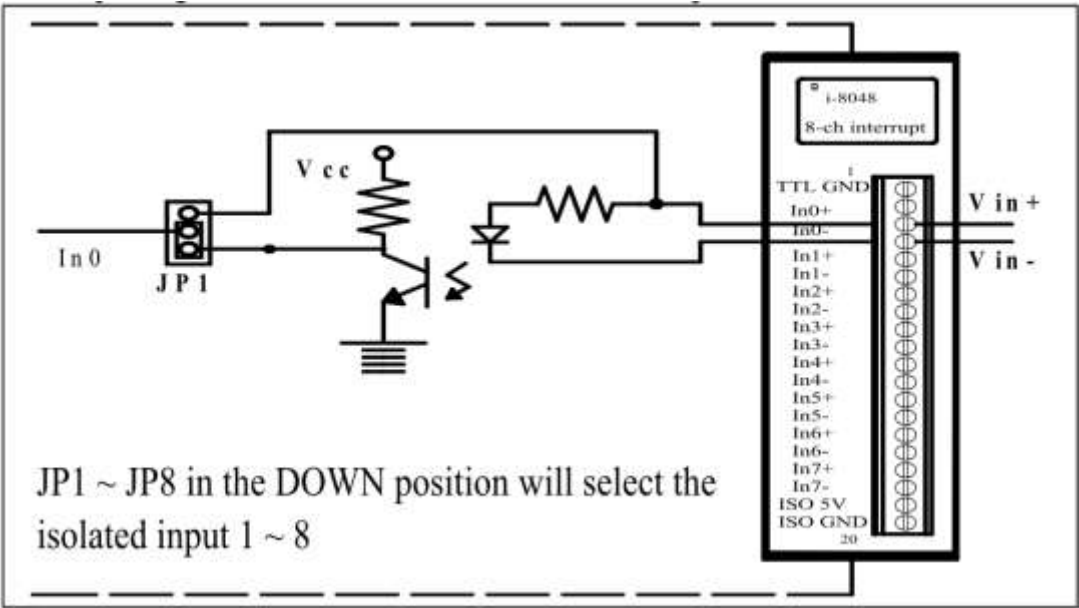
Isolated input (default)



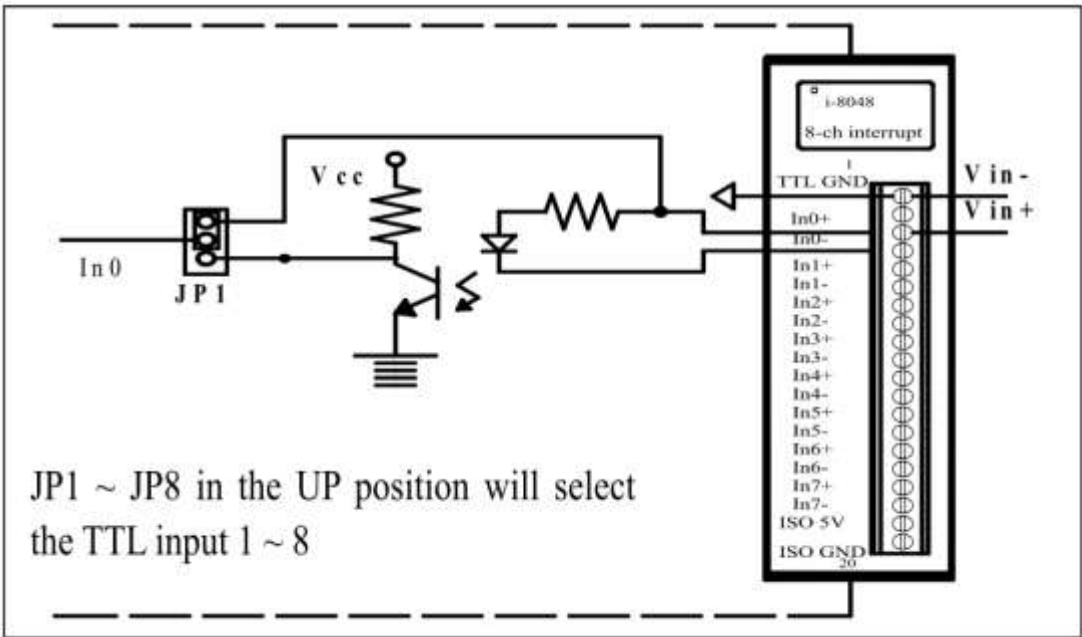
TTL input

The input signal can be either isolated or TTL input for **I-8048W** as follows:

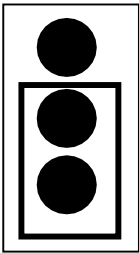
Isolated input



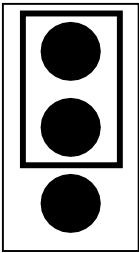
TTL input



JP1	for ch0
JP2	for ch1
JP3	for ch2
JP4	for ch3
JP5	for ch4
JP6	for ch5
JP7	for ch6
JP8	for ch7



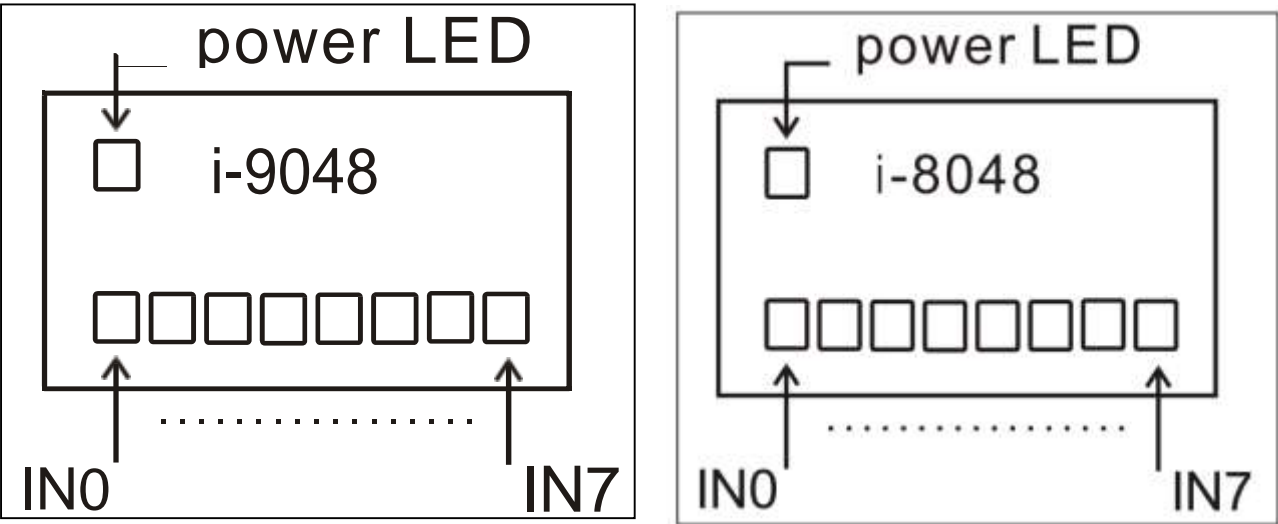
Isolated input (default)



TTL input

2.1.5. Digital Input & LED indicators

The LED status and the digital input relation are listed as follows:



Isolated:

Input status	Digital Input(Logic level)	Electric signal	LED
OPEN	1	Low	OFF
0 ~ +1V	1	Low	OFF
+4 ~ +30V	0	High	ON

TTL:

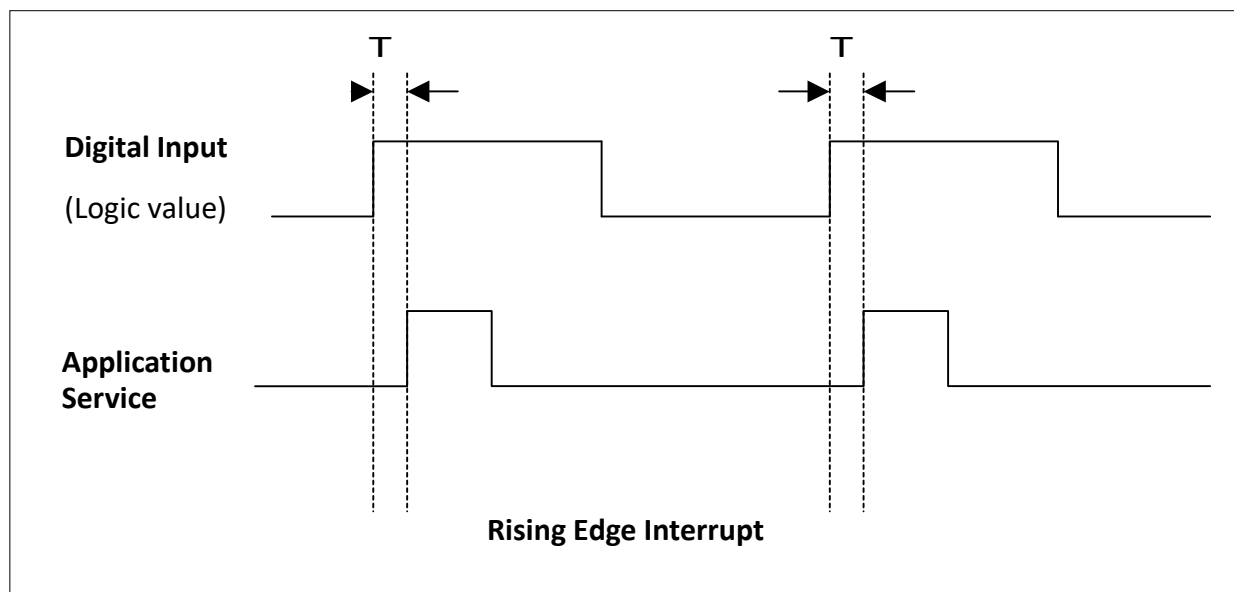
Input status	Digital Input(Logic level)	Electric signal	LED
OPEN	1	Low	OFF
0 ~ +0.8V	0	Low	ON
+2 ~+ 5V	1	High	OFF

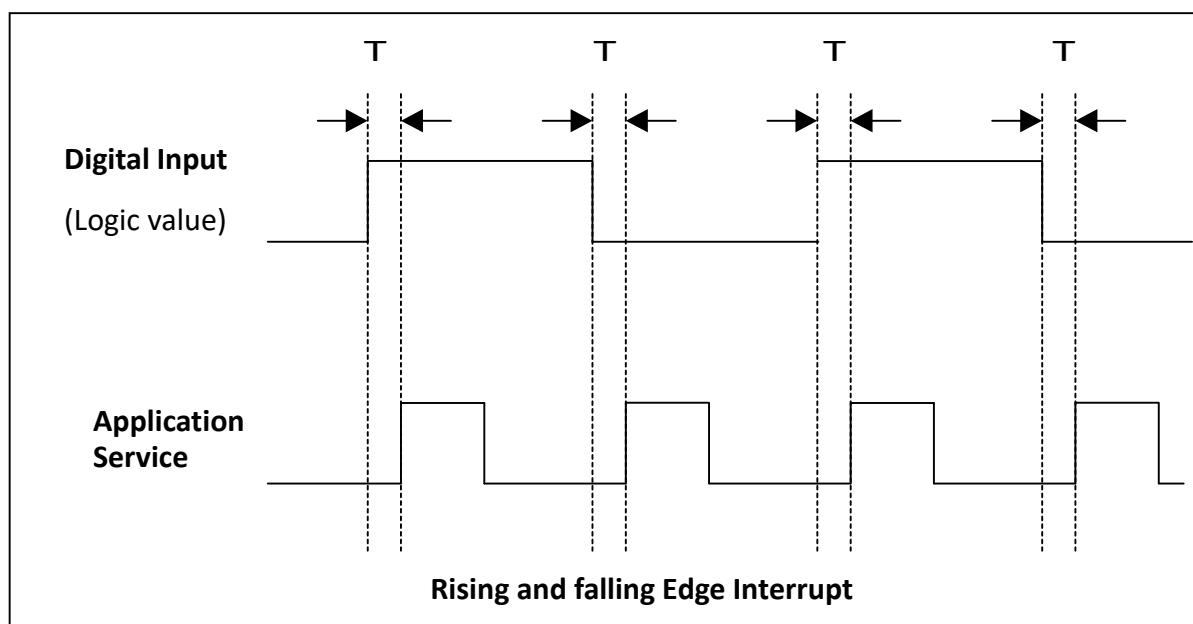
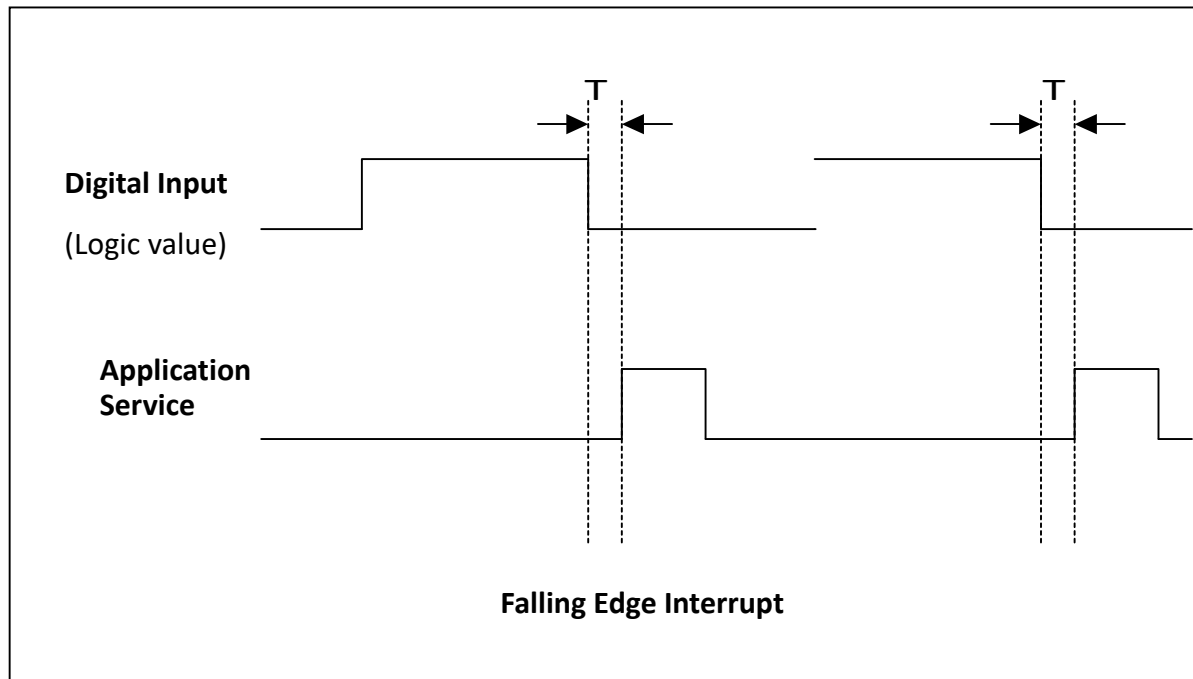
2.1.6. Programmable Rising/Falling interrupt

Each channel of the I-9048 can be programmed as one of the following types individually.

- Rising edge interrupt input
- Falling edge interrupt input
- Rising edge and Falling edge interrupt input

The Interrupt trigger types are as follows:





T is the leading time between I-9048 receiving the input signal to the application receiving the interrupt service.

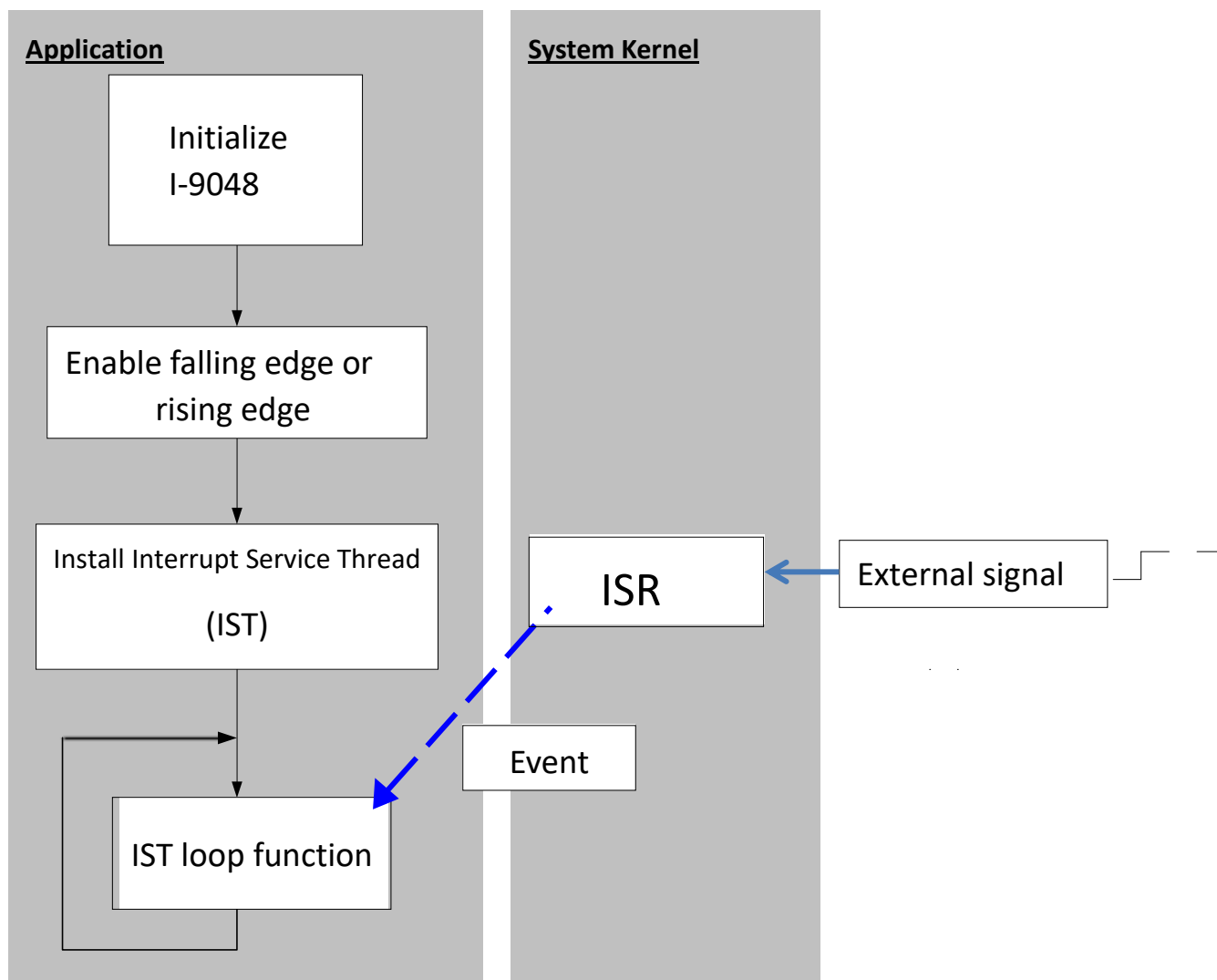
Note: If the interrupt signal is too short, the new status may be as same as old status. In that condition, the interrupt service thread cannot identify which interrupt source is active. So the interrupt signal must be hold long enough until the interrupt service thread is executed. This holding time is different for different hardware & O.S.

2.2. Software Introduction

Users can follow the sections to understand the usage of I-9048 / I-8048W Library software.

We offer multi-functions in the I-9048 / I-8048W library, and users can use these functions to develop I-9048 / I-8048W program.

2.2.1. Software flow chart



◆ Initialize I-9048:

Gives all internal variables an initial value. Any interrupt signal clear to low, then all interrupt will not be blocked and CPU will be able to receive any further interrupts.

The function declaration:

```
I8048W_Init (int slot)
```


◆ Enable falling edge or rising edge:

1. Each channel can set to:

- Rising edge interrupt
- Falling edge interrupt
- Rising edge and Falling edge interrupt

2. User can enable one channel, several channels or all channels as interrupt signal input.

The function declaration:

- `I8048W_Set_RisingReg (int slot, int channel, int Enable)`
- `I8048W_Set_FallingReg (int slot, int Channel, int Enable)`

◆ Install IST

The interrupt service thread (IST) is a thread that does most of the interrupt processing.

After installing an IST, when “External signal” enters the I-9048, the kernel signals the event on behalf of the ISR, and then the IST performs necessary operations collect the data and process them. When the interrupt processing is completed, the IST informs the kernel to re-enable the hardware interrupt.

The function declaration:

- `I8048W_InstallISR(int slot, void *ISR_Function (int slot));`

ISR is a piece of code that is built-in or loaded into the kernel, and the user's application cannot handle this function. A process or driver can create threads (called ISTs) specifically registered to handle interrupts in user space. After a hardware interrupt occurs, the ISR processes the interrupt and immediately informs the system to execute the corresponding IST. ISR and IST are usually used in pairs. Although the API function name is ISR, it is actually used to install IST, users can design their own IST functions.

The IST function declaration:

`Void ISR_Function (int slot);`

2.2.2. Recognize different interrupt service requests

I-9048 / I-8048W is an 8 channels rising/falling edge programmable interrupt module. It can plug into slot of the PAC controllers. The section introduces how to recognize interrupt requests from different slots and channels.

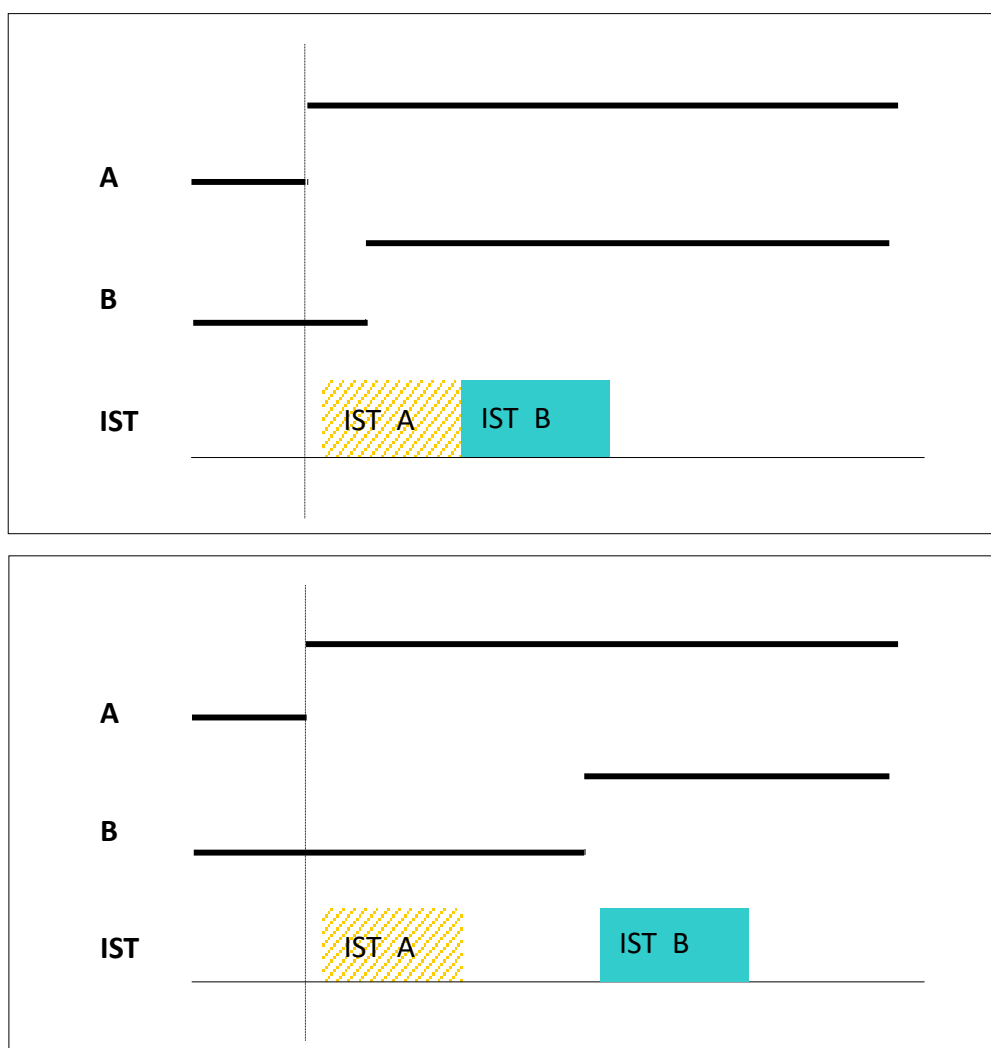
Requests from different slots

Each slot is assigned an interrupt pin of the CPU shown in section 1.3. So their ISTs are different. CPU executes different ISR when the request is from different slot.

Considering when the interrupts occur, the situations can be divided to two kinds.

Interrupted at different time

CPU executes ISTs one by one in the order of when they occurred.



3. API References

ICPDAS supplies a range of C API functions for the I-8048W/I-9048 module. When developing a custom program, refer to either the i8048.h header file, or the API functions described in the following sections for more detailed information.

The following is an overview of the functions provided in the LinPAC library - **libi8k.a**. Detailed information related to individual functions can be found in the following sections.

More details of where to find the relevant libraries and files, and refer to Chapter 1.3. Location of the Demo and Library Programs.

3.1. Function List

The common API functions of I-8048W / I-9048 Module list as below table. Detailed information related to individual functions can be found in the following sections.

Systematics	Function	Description
DI Read	I8048W_DI_ALL	This function reads the full channel DI value of the I-9048 module.
	I8048W_DI_Ch	This function read the single-channel DI status of the I-9048 module.
RisingReg / FallingReg	I8048W_Set_RisingReg	This function Enable/Disable the rising interrupt of an I-9048 channel.
	I8048W_Set_FallingReg	This function Enable/Disable the falling interrupt of an I-9048 channel.
	I8048W_Read_RisingReg	This function Read the rising interrupt setting status of an I-9048 channel.
	I8048W_Read_FallingReg	This function Read the falling interrupt setting status of an I-9048 channel.
Event	I8048W_Read_RisingEvent	This function read rising interrupt status from the I-9048 modules.
	I8048W_Read_FallingEvent	This function read falling interrupt status from the I-9048 modules.
	I8048W_Read_RisingEventCount	This function read total count values of the rising interrupt occurred on an I-9048 channel.
	I8048W_Read_FallingEventCount	This function read total count values of the rising interrupt occurred on an I-9048 channel.
	I8048W_Clear_RisingEventCount	This function clear total count values of the rising interrupt occurred on an I-9048 channel
	I8048W_Clear_FallingEventCount	This function clear total count values of the falling interrupt occurred on an I-9048 channel
Init	I8048W_Init	This function sets the rising or falling trigger type of the I-9048 module.
Interrupt service routine	I8048W_InstallISR	This function is used to install a slot interrupt service route for I-9048.
	I8048W_UnInstallISR	This function is used to uninstall a slot interrupt service route and disable a hardware interrupt.
	I8048W_UnFreezeINT	This function clears interrupt status of the I-9048 channel.
Library	I8048W_GetLibVersion	This function to get the library version of I-9048.

3.2. I8048W_DI_ALL

This function reads the full channel DI value of the I-8048W / I-9048 module.

Syntax

【c】

```
int I8048W_DI_ALL (  
    int iSlot  
);
```

Parameters

iSlot

[in] Specifies the slot number (1 - 8).

Return Value

Return the full channel DI value.

Example

[C]

```
int byteData, slotNumber = 0;  
byteData = I8048W_DI_ALL(slotNumber);
```

Remarks

Isolated			TTL		
JPn selected isolated input	Digital Input	LED	JPn selected TTL input	Digital Input	LED
Input is OPEN	High	OFF	Input is OPEN	High	OFF
Input is 3.5 ~ 30V	Low	ON	Input is 0 ~ 0.8V	Low	ON
Input is 0 ~ 1V	High	OFF	Input is 2 ~ 5V	High	OFF

3.3. I8048W_DI_Ch

This function read the single-channel DI status of the I-8048W / I-9048 module.

Syntax

[c]

```
int I8048W_DI_Ch(  
    int slot,  
    int DI_Channel,  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

DI_Channel

[in] The single channel of the I-9048 module to be read.

Return Value

1: DI ON

0: DI OFF

Example

[C]

```
int DI=0;  
int channel = 0;  
int slot = 0;  
DI = I8048W_DI_Ch(slot, channel);
```

3.4. I8048W_Set_RisingReg

This function is used to Enable/Disable the Rising interrupt of an I-8048W / I-9048 channel.

Syntax

[c]

```
void I8048W_Set_RisingReg(  
    int slot,  
    int channel,  
    int enable  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Enable

[in] Select Enable or Disable, 1=Enable, 0=Disable

Return Value

None

3.5. I8048W_Set_FallingReg

This function is used to Enable/Disable the Falling interrupt of an I-8048W / I-9048 channel.

Syntax

【c】

```
void I8048W_Set_FallingReg(  
    int slot,  
    int channel,  
    int enable  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Enable

[in] Select Enable or Disable, 1=Enable, 0=Disable

Return Value

None

3.6. I8048W_Read_RisingReg

This function Read the rising interrupt setting status of an I-8048W / I-9048 channel.

Syntax

[c]

```
int I8048W_Read_RisingReg (  
    int slot,  
    int channel  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Return Value

Rising status:

1: Enable

0: Disable

3.7. I8048W_Read_FallingReg

This function Read the falling interrupt setting status of an I-8048W / I-9048 channel.

Syntax

[c]

```
int I8048W_Read_FallingReg (  
    int slot,  
    int channel  
);
```

Parameters

slot

[in] specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Return Value

1: Enable

0: Disable

3.8. I8048W_Read_RisingEvent

This function read rising interrupt status from the I-8048W / I-9048 modules.

Syntax

[c]

```
int I8048W_Read_RisingEvent(  
    int slot,  
    int channel  
);
```

Parameters

slot

[in] specifies the slot number (1 - 8).

channel

[in] Specifies the channel number(0 to 7).

Return Value

1: Rising interrupt occur

0: No rising interrupt occur

3.9. I8048W_Read_FallingEvent

This function read falling interrupt status from the I-8048W / I-9048 modules.

Syntax

[c]

```
int _I8048W_Read_FallingEvent (  
    int slot,  
    int channel  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Return Value

1: falling interrupt occur

0: No falling interrupt occur

3.10. I8048W_Read_RisingEventCount

This function read total count values of the rising interrupt occurred on an I-8048W / I-9048 channel.

Syntax

[C]

```
DWORD I8048W_Read_RisingEventCount(  
    int slot,  
    int channel  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Return Value

Rising interrupt count value

3.11. I8048W_Read_FallingEventCount

This function read total count values of the rising interrupt occurred on an I-8048W / I-9048 channel.

Syntax

[c]

```
DWORD I8048W_Read_FallingEventCount(  
    int slot,  
    int channel  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Return Value

falling interrupt count value

3.12. I8048W_Clear_RisingEventCount

This function clear total count values of the rising interrupt occurred on an I-8048W / I-9048 channel.

Syntax

【c】

```
void I8048W_Clear_RisingEventCount (  
    int slot,  
    int channel  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Return Values None

Return Value

None.

3.13. I8048W_Clear_FallingEventCount

This function clear total count values of the falling interrupt occurred on an I-8048W / I-9048 channel.

Syntax

[c]

```
void I8048W_Clear_FallingEventCount (  
    int slot,  
    int channel  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Return Value

None

3.14. I8048W_Init

This function is used to initialize the driver and confirm the hardware ID of the I-8048W / I-9048 module.

Syntax

[C]

```
int I8048W_Init(  
    int iSlot  
);
```

Parameters

iSlot

[in] Specifies the slot number (1 - 8).

Return Value

Return 0 if success, otherwise false.

Example

[C]

```
int ret = 0;  
int slotNumber = 0;  
ret = I8048W_Init(slotNumber);
```

3.15. I8048W_InstallISR

This function is used to install a slot interrupt service thread for I-9048.

Syntax

【c】

```
void I8048W_InstallISR (  
    int iSlot,  
    CALLBACK_FUNC *IST_Function,  
);
```

Parameters

iSlot

[in] Specifies the slot number (1 - 8).

**ISR_Function*

[in] After installing an IST, the CPU will execute the ISR_Function when “External signal” enters the I-9048.

Return Value

None

Example

[C]

```
int slotNumber = 1;
int channelNumber =1;
int ret=0;

ret = I8048W_Init(slotNumber);
if (!ret){
    I8048W_Set_RisingReg(slotNumber,channelNumber,1);
    I8048W_Set_FallingReg(slotNumber,channelNumber,0);
    I8048W_InstallISR(slotNumber, ISRFUN);
}
...
Void ISRFUN(){
    int temp = 0;
    ....
    I8048W_UnFreezeINT(slotNumber, channelNumber);
    return 1;
}
```

3.16. I8048W_UnInstallISR

This function is used to uninstall a slot interrupt service thread and disable a hardware interrupt.

Syntax

[C]

```
void I8048W_UnInstallISR (  
    int iSlot,  
);
```

Parameters

iSlot

[in] Specifies the slot number (1 - 8).

Return Value

None

Example

[C]

```
int slotNumber = 1;  
  
I8048W_UnInstallISR(slotNumber);
```

3.17. I8048W_UnFreezeINT

This function clear interrupt status of the I-8048W / I-9048 channel.

Syntax

[C]

```
void I8048W_UnFreezeINT(  
    int slot,  
    int channel  
);
```

Parameters

slot

[in] Specifies the slot number (1 - 8).

channel

[in] Specifies the channel number (0 to 7).

Return Value

None

Examples

[C]

```
int channel = 0;  
int slotNumber = 0;  
I8048W_UnFreezeINT(slotNumber,channel);
```

3.18. I8048W_GetLibVersion

This function to get the library version of I-8048W / I-9048W.

Syntax

【c】

```
short I8048W_GetLibVersion()
```

Return Value

Version number

For Example : 0x106 = Rev 1.0.6

Revision History

This chapter provides revision history information to this document.

The table below shows the revision history.

Revision	Date	Description
1.0.2	October 2020	Initial issue