I-8017/I-9017 Series
Linux API Reference Manual

V 3.0.0 July 2018

Written by Edward Wu
Edited by Anna Huang
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

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

1. Introduction 4
   - 1.1. Specifications 6
   - 1.2. Pin Assignments 8
   - 1.3. Jumper Settings 14
   - 1.4. Wire Connections 19
   - 1.5. Block Diagram 21
2. Quick Start 23
3. Demo Programs 24
4. API References 25
   - 4.1. I8017_Init 26
   - 4.2. I8017_GetFirmwareVersion 27
   - 4.3. I8017_GetLibVersion 28
   - 4.4. I8017_GetSingleEndJumper 29
   - 4.5. I8017_ReadAI 30
   - 4.6. I8017_ReadAI_AVG 32
   - 4.7. I8017_ReadAIHex 34
   - 4.8. I8017_ReadAIHex_AVG 36
   - 4.9. I8017_ReadGainOffset_Info 38
   - 4.10. I8017_Read_mA_GainOffset 40
   - 4.11. I8017_Select_SingleEnd 42
   - 4.12. I8017_Get_D_Sub_Status 43
5. Troubleshooting 45
   - 5.1. Service Request Requirements 46
   - 5.2. What to do when the data read from the module seems unstable 47

Appendix A. Error Code 48

1. Introduction

The I-8017W/I-9017 I/O modules are high performance analog input module, up to 16-channel single-ended or 8-channel differential inputs. It features 14-bit resolution, 100Ks/s sampling rates. It provides isolation protection of 2500 Vrms.

The I-8017 I/O modules can be used to measure both voltage and current source. The I-8017DW and I-8017HCW modules include a Jumper that can be used to set the discrete input circuits to add a 125 Ω resistor, so it's not necessary to add external resistor for Differential input. For the I-8017HW an optional external 125Ω resistor is required to measure the current source.

Applications

- High speed data acquisition systems
- Process monitoring and control
- Vibration analysis
- Digital pattern generator from the digital I/O port

Applicable Platform table

The following table shows which platform the module applies to.

Platform	OS	Module
XPAC	XP-8000(WES)	I-8017HW/I-8017DW/I-8017HCW
	XP-8000-Atom (WES)	I-8017HW/I-8017DW/I-8017HCW
	XP-8000-WES7 (WES7)	I-8017HW/I-8017DW/I-8017HCW
	XP-8000-CE6 (WinCE 6.0)	I-8017HW/I-8017DW/I-8017HCW
	XP-8000-Atom-CE6 (WinCE 6.0)	I-8017HW/I-8017DW/I-8017HCW
	XP-9000-WES7(WES7)	I-9017/I-9017-15/I-9017C-15
WinPAC	WP-8000 (CE 5.0/7.0)	I-8017HW/I-8017DW/I-8017HCW
	WP-9000-CE7 (CE 7.0)	I-9017/I-9017-15/I-9017C-15
LinPAC	LinPAC-8000(Linux kernel 3.2/4.4)	I-8017HW/I-8017DW/I-8017HCW
	LinPAC-9000(Linux kernel 3.2/4.4)	I-9017/I-9017-15/I-9017C-15
IPAC	iPAC-8000 (MiniOS7)	I-8017HW/I-8017DW/I-8017HCW
	I-8000 (MiniOS7)	I-8017HW/I-8017DW/I-8017HCW

The I-8017DW module is equipped with a D-sub connection, meaning that it can be connected using a 37-pin D-sub Connector, as shown in the image below:

For more detailed information regarding 37-pin D-sub Connectors refer to the models indicated in the table below:

Model	Description
DN-37-A	I/O Connector Block with DIN-Rail Mounting and 37-pin D-sub Connector (Pitch: 5.08 mm)
DN-37-381-A	I/O Connector Block with DIN-Rail Mounting and 37-pin D-sub Connector (Pitch: 3.81 mm)
CA-3705A	Male-Female D-sub Cable 0.5 m
CA-3710A	Male-Female D-sub Cable 1 m
CA-3715A	Male-Female D-sub Cable 1.5 m

1.1. Specifications

I-8017HW/I-8017HCW/I-8017DW

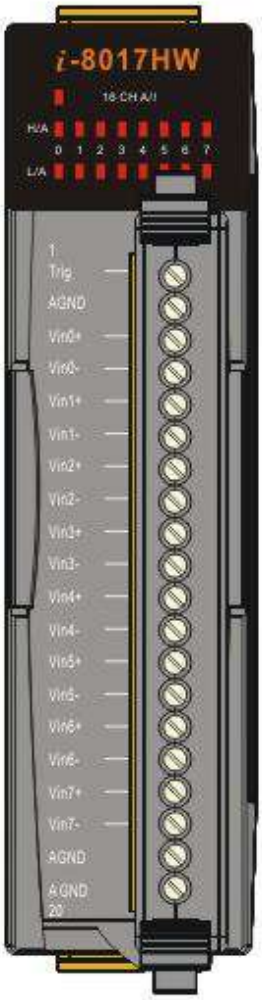
Model	I-8017HW	I-8017DW	I-8017HCW
Analog Output			
Channels	8-ch Differential/16-Single-ended		
Voltage Input Range	±1.25, ±2.5, ±5 V, ±10 V		-
Current Input Range	±20 mA (Requires OptionalExternal 125 Ω Resistor)		±20 mA (Jumper Select)
Resolution	14-bit		
Sample Rate	Single Channel Polling Mode :100K S/s Single Channel Interrupt Mode: 50K S/s 8 channel Scan Mode : 16 K S/s		
Accuracy	0.1% of FSR		
Zero Drift	± 0.1 uV/°C		
Span Drift	± 10 ppm/°C		
ESD Protection	±4 kV Contact for each Terminal		
Input Impedance	20 K, 200 K, 20 M (Jumper Select)		
Input Bandwidth	100 KHz		
Connector	20 Pin Terminal Block		
LED Indicators			
System LED Indicator	1 LED as Power Indicator		
I/O LED Indicator	16 LEDs as User defined Indicators		
Isolation			
Intra-module Isolation, Field-to-Logic	2500 Vrms		
Power			
Power Consumption	2 W Max.		
Mechanical			
Dimension (L x W x H)	102 mm x 30 mm x 115 mm		
Environment			
Operating Temperature	-25 °C ~ +75°C		
Storage Temperature	-30 °C ~ +80°C		
Humidity	10% ~ 90% RH, non-condensing		

I-9017/I-9017-15/I-9017C-15

Model	I-9017	I-9017-15	I-9017C-15
Analog Output			
Channels	8 Differential/ 16 Single-ended	15 Differential/ 30 Single-ended	15 Differential
Voltage Input Range	±1.25, ±2.5, ±5 V, ±10 V		-
Current Input Range	±20 mA (Requires OptionalExternal 125 Ω Resistor)		±20 mA
Resolution	14-bit		
Sample Rate	Single Channel Polling Mode :100K S/s Single Channel Interrupt Mode: 50K S/s 15-channel Scan Mode : 16 K S/s		
Accuracy	0.1% of FSR		
Input Bandwidth	100 KHz		
Zero Drift	± 0.1 uV/°C		
Span Drift	± 10 ppm/°C		
Input Impedance	20 K, 200 K, 20 M (Jumper Select)		-
LED Indicators			
System LED Indicator	1 LED as Power Indicator		
I/O LED Indicator	16 LED as Status Indicator		
EMS Protection			
ESD (IEC 61000-4-2)	±4 kV Contact for each Terminal ±8 kV Air for Random Point		
Isolation			
Intra-module Isolation, Field-to-Logic	2500 Vrms		
Power			
Power Consumption	2 W Max.		
Mechanical			
Dimension (L x W x H)	144 mm x 30.3 mm x 134 mm		
Environment			
Operating Temperature	-25 °C ~ +75°C		
Storage Temperature	-40°C ~ +85°C		
Humidity	10 % ~ 90% RH, non-condensing		

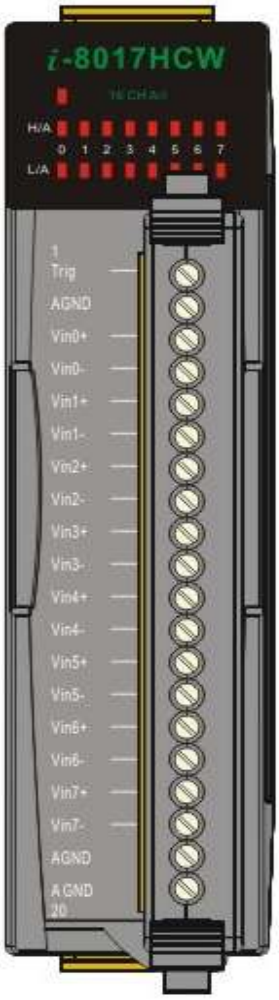
1.2. Pin Assignments

I-8017HW:



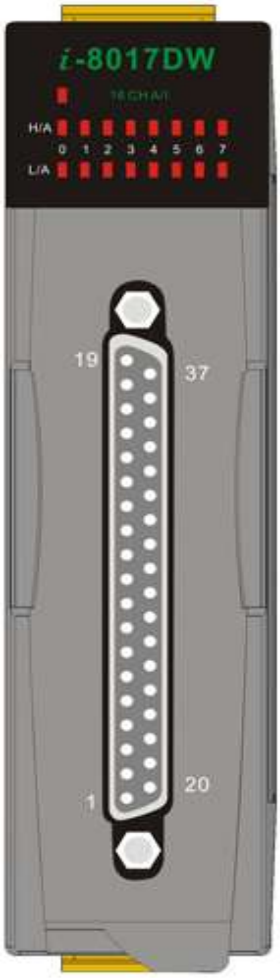
Terminal No.	Pin Assignment Name	
	Differential	Single-ended
01	Trig	Trig
02	AGND	AGND
03	Vin0 +	Vin0
04	Vin0 -	Vin8
05	Vin1 +	Vin1
06	Vin1 -	Vin9
07	Vin2 +	Vin2
08	Vin2 -	Vin10
09	Vin3 +	Vin3
10	Vin3 -	Vin11
11	Vin4 +	Vin4
12	Vin4 -	Vin12
13	Vin5 +	Vin5
14	Vin5 -	Vin13
15	Vin6 +	Vin6
16	Vin6 -	Vin14
17	Vin7 +	Vin7
18	Vin7 -	Vin15
19	AGND	AGND
20	AGND	AGND

I-8017HCW:



Terminal No.		Pin Assignment	
		Differential	Single-ended
	01	Trig	Trig
	02	AGND	AGND
	03	Vin0+	Vin0
	04	Vin0-	Vin8
	05	Vin1+	Vin1
	06	Vin1-	Vin9
	07	Vin2+	Vin2
	08	Vin2-	Vin10
	09	Vin3+	Vin3
	10	Vin3-	Vin11
	11	Vin4+	Vin4
	12	Vin4-	Vin12
	13	Vin5+	Vin5
	14	Vin5-	Vin13
	15	Vin6+	Vin6
	16	Vin6-	Vin14
	17	Vin7+	Vin7
	18	Vin7-	Vin15
	19	AGND	AGND
	20	AGND	AGND

I-8017DW:

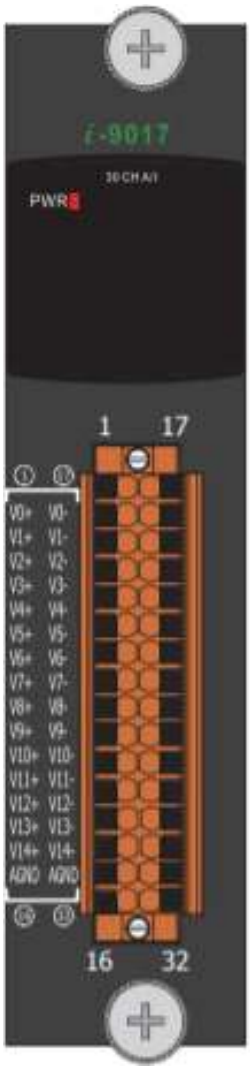


Pin Assignment		Terminal No.		Pin Assignment	
Differential	Single-ended			Differential	Single-ended
AGND	AGND	19	37	BK Sensor	BK Sensor
Trig	Trig	18	36	-	-
AI7-	Ai15	17	35	-	-
AI7+	AI7	16	34	-	-
AI6-	AI14	15	33	-	-
AI6+	AI6	14	32	-	-
AI5-	AI13	13	31	-	-
AI5+	AI5	12	30	-	-
AI4-	AI12	11	29	-	-
AI4+	AI4	10	28	-	-
AI3-	AI11	09	27	-	-
AI3+	AI3	08	26	-	-
AI2-	AI10	07	25	-	-
AI2+	AI2	06	24	-	-
AI1-	AI9	05	23	-	-
AI1+	AI1	04	22	-	-
AI0-	AI8	03	21	AGND	AGND
AI0+	AI0	02	20	AGND	AGND
BK Sensor	BK Sensor	01			

I-9017:

Pin Assignment		Terminal No.		Pin Assignment	
Differential	Single-ended			Differential	Single-ended
Trig+	Trig+	1	11	Trig-	Trig-
V0+(I0+)	Vin0	2	12	V0-(I0-)	Vin8
V1+(I1+)	Vin1	3	13	V1-(I1-)	Vin9
V2+(I2+)	Vin2	4	14	V2-(I2-)	Vin10
V3+(I3+)	Vin3	5	15	V3-(I3-)	Vin11
V4+(I4+)	Vin4	6	16	V4-(I4-)	Vin12
V5+(I5+)	Vin5	7	17	V5-(I5-)	Vin13
V6+(I6+)	Vin6	8	18	V6-(I6-)	Vin14
V7+(I7+)	Vin7	9	19	V7-(I7-)	Vin15
AGND	AGND	10	20	AGND	AGND

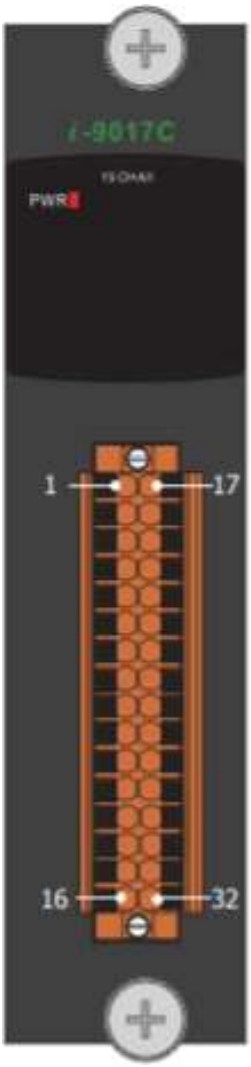
I-9017-15:



Pin Assignment		Terminal No.		Pin Assignment	
V0+	Vin0	01	17	V0 -	Vin15
V1+	Vin1	02	18	V1 -	Vin16
V2+	Vin2	03	19	V2 -	Vin17
V3+	Vin3	04	20	V3 -	Vin18
V4+	Vin4	05	21	V4 -	Vin19
V5+	Vin5	06	22	V5 -	Vin20
V6+	Vin6	07	23	V6 -	Vin21
V7+	Vin7	08	24	V7 -	Vin22
V8+	Vin8	09	25	V8-	Vin23
V9+	Vin9	10	26	V9 -	Vin24
V10+	Vin10	11	27	V10 -	Vin25
V11+	Vin11	12	28	V11 -	Vin26
V12+	Vin12	13	29	V12 -	Vin27
V13+	Vin13	14	30	V13 -	Vin28
V14+	Vin14	15	31	V14 -	Vin29
AGND	AGND	16	32	AGND	AGND

32-pin Connector

I-9017C-15:

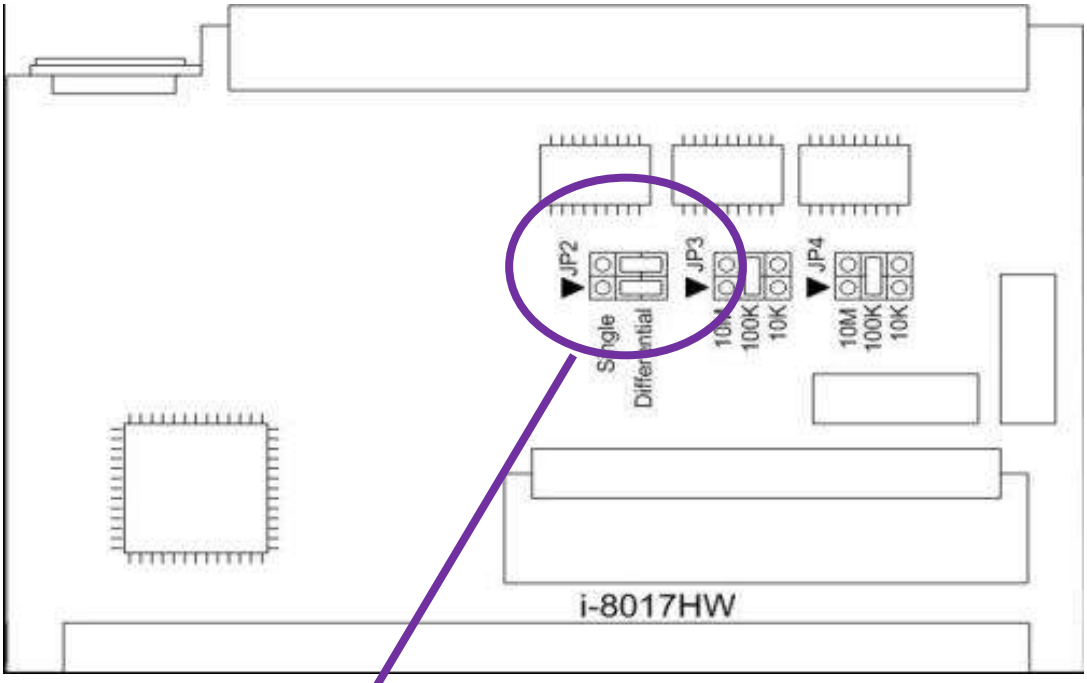


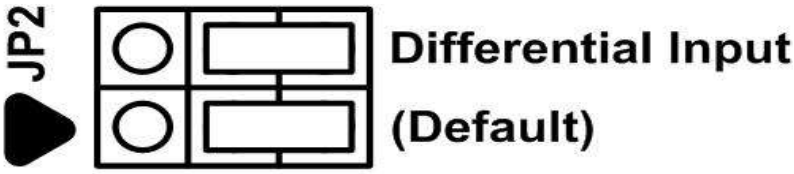
Pin Assignment	Terminal No.		Pin Assignment
I0+	01	17	I0 -
I1+	02	18	I1 -
I2+	03	19	I2 -
I3+	04	20	I3 -
I4+	05	21	I4 -
I5+	06	22	I5 -
I6+	07	23	I6 -
I7+	08	24	I7 -
I8+	09	25	I8-
I9+	10	26	I9 -
I10+	11	27	I10 -
I11+	12	28	I11 -
I12+	13	29	I12 -
I13+	14	30	I13 -
I14+	15	31	I14 -
AGND	16	32	AGND
			32-pin Connector

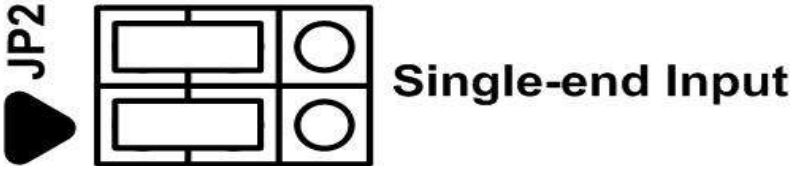
1.3. Jumper Settings

I-8017HW

Single-ended and Differential Jumper:

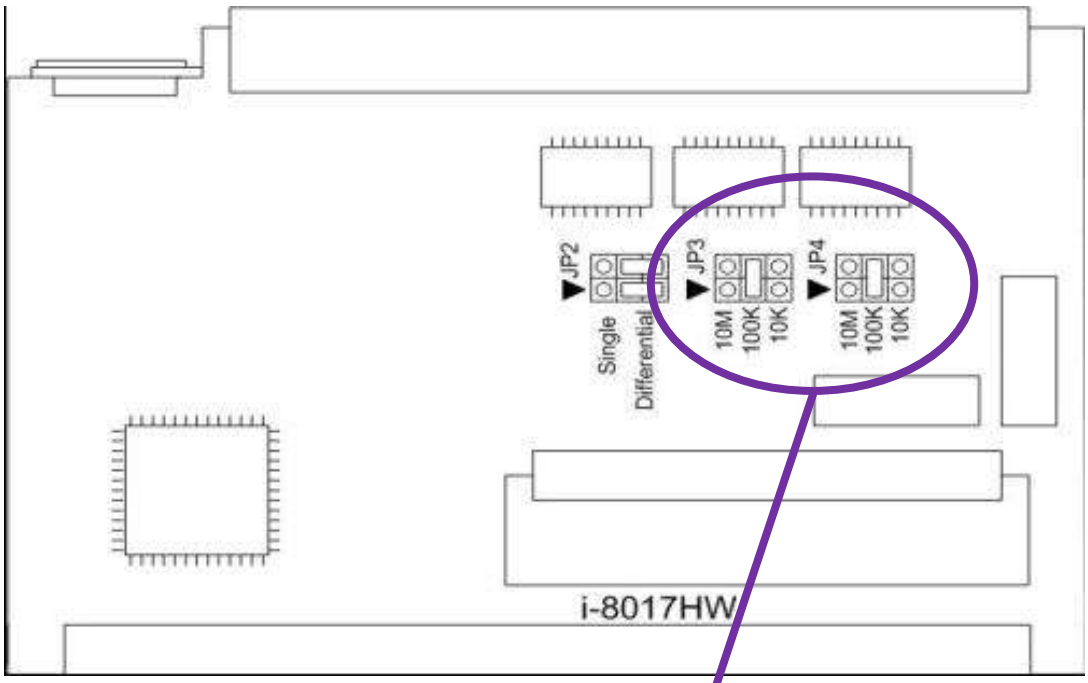


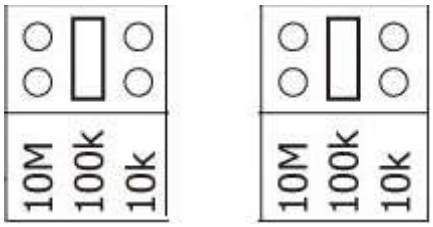




This jumper is used to set the discrete input circuits as either “Single-ended” or “Differential” inputs.

Adjusting the Input impedance





Select Input Impedance: 200 kΩ (Default)

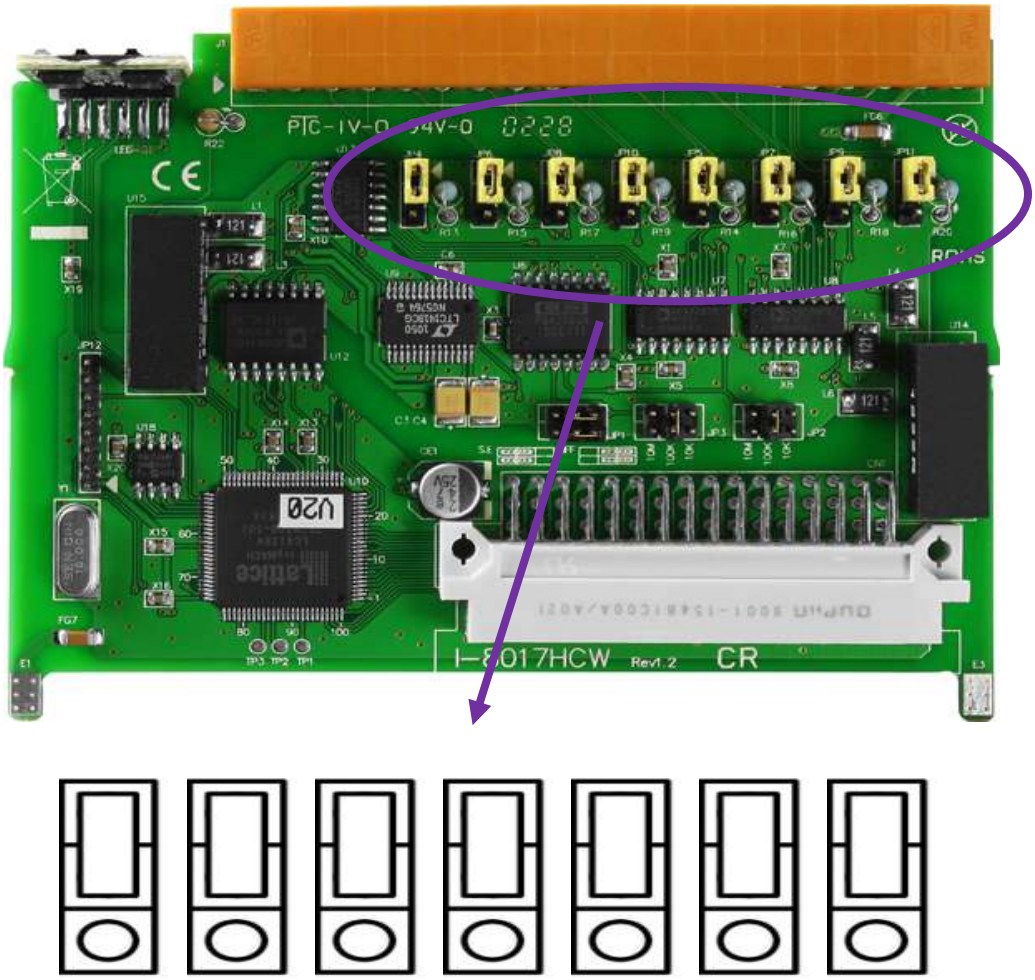
Note: 1. The Jumpers should set on the same value
2. Input Impedance = 2 x setting value

The I-8017 series modules allows three input impedance options including as 20 kΩ, 200kΩ (the default setting) and 20MΩ to meet system requirements.

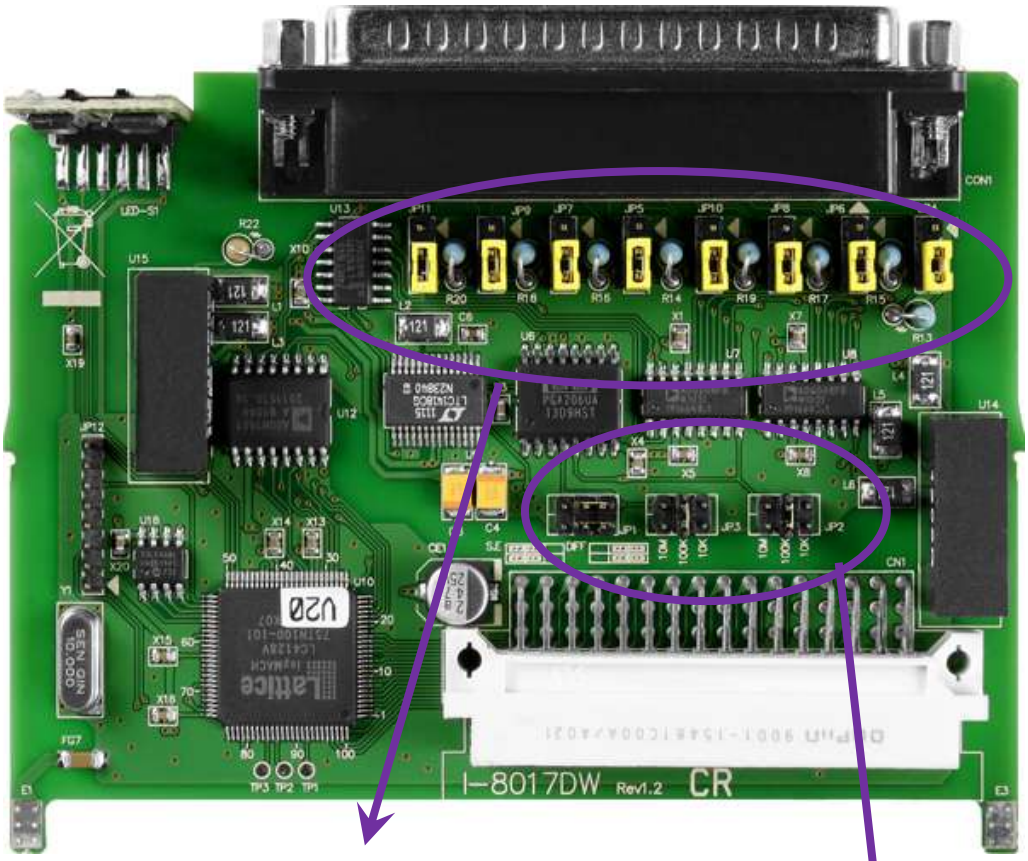
In most cases, 200kΩ is sufficient. Note that each time the input impedance is adjusted on a calibrated module, the module must be recalibrated. For more details, refer to the relevant Calibration information, which can be found in Section 5.1 if you are using either an I-8000 or iPAC-8000 (MiniOS7 platform) controller, or in Section 5.2 if you are using a WinCE or WES platform unit.

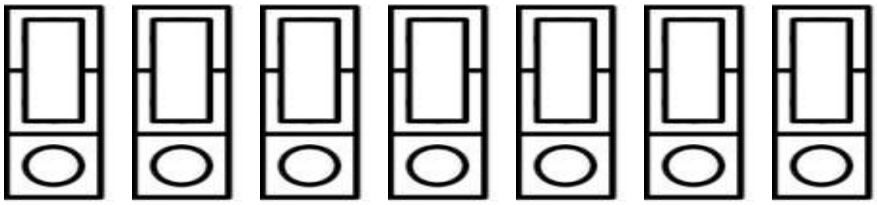
125Ω Resistor Jumper

By default, the I-8017HCW module is configured for current source measurement, and the I-8017DW is configured for voltage measurement, as illustrated below:



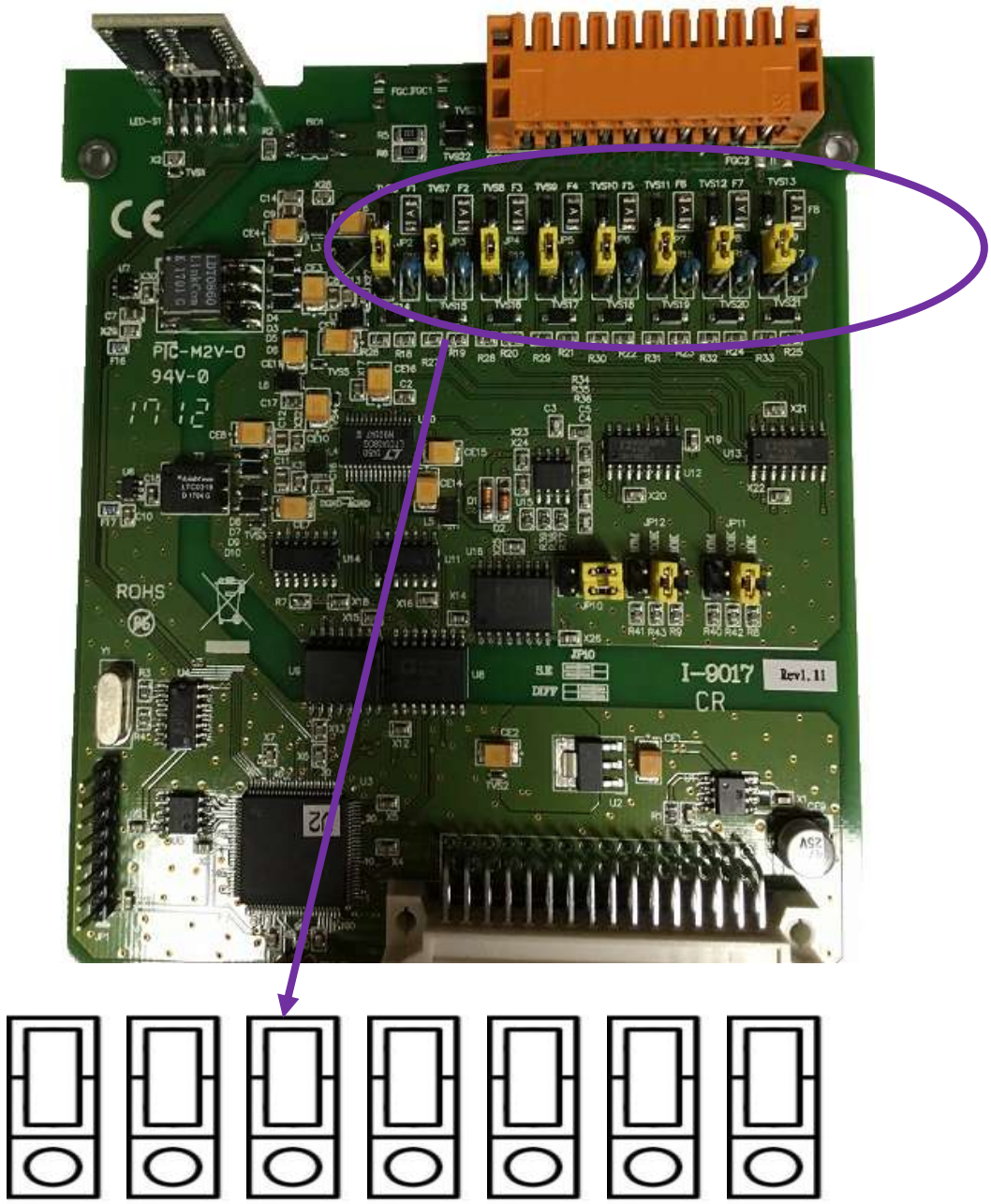
The default jumper position for current measurement on the I-8017HCW





The default jumper position for voltage measurement on the I-8017DW

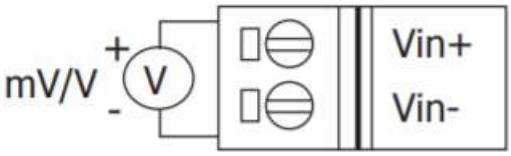
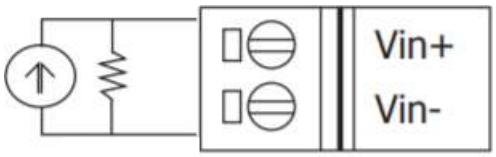
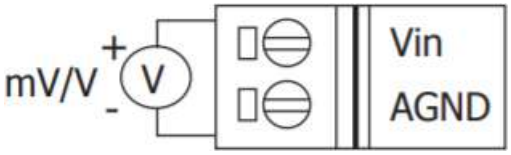
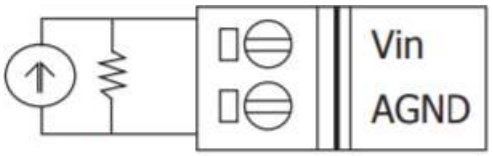
The jumper settings used to adjust both the input impedance and the single-ended and differential input on the I-8017HW are the same as those for the I-8017DW,I-8017HCW and the I-9017.



The default jumper position for voltage measurement on the I-9017

1.4. Wire Connections

I-9017

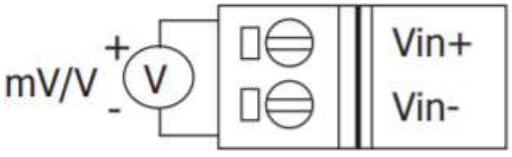
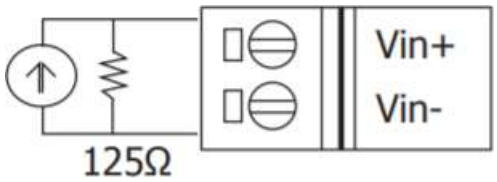
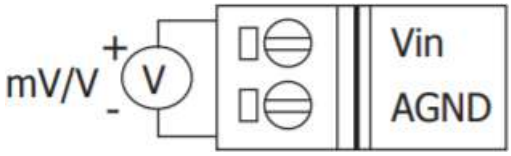
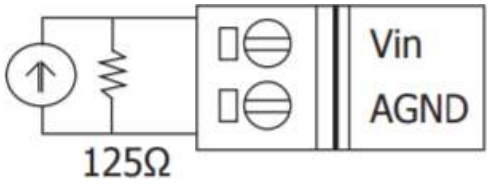
	Voltage Input Wiring	Current Input Wiring
Differential		
Single-ended		

Note:

Differential Input Type: Current Input Wiring need to jumper at current input.

Single-ended Input Type: Current Input Wiring need to jumper at voltage input, an options external 125 Ω resistor is required.

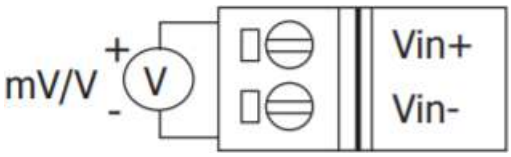
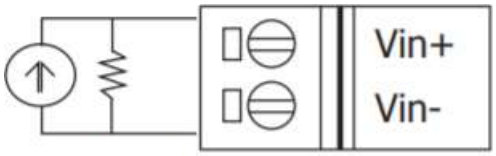
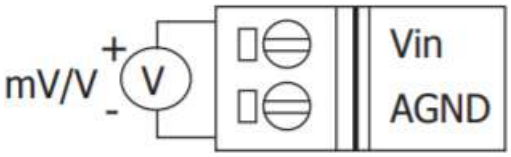
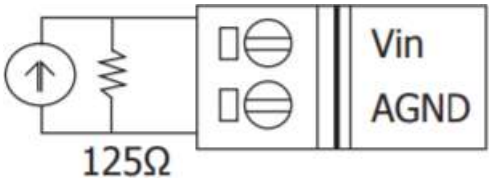
I-8017HW/I-9017-15:

	Voltage Input Wiring	Current Input Wiring
Differential		
Single-ended		

Note:

When connecting to a current source, an optional external 125 Ω resistor is required.

I-8017DW/I-8017HCW

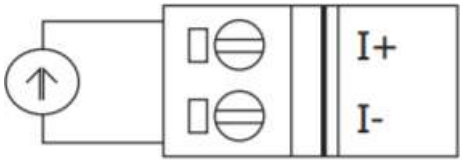
	Voltage Input Wiring	Current Input Wiring
Differential		
Single-ended		

Note:

Differential Input Type: Current Input Wiring need to jumper at current input.

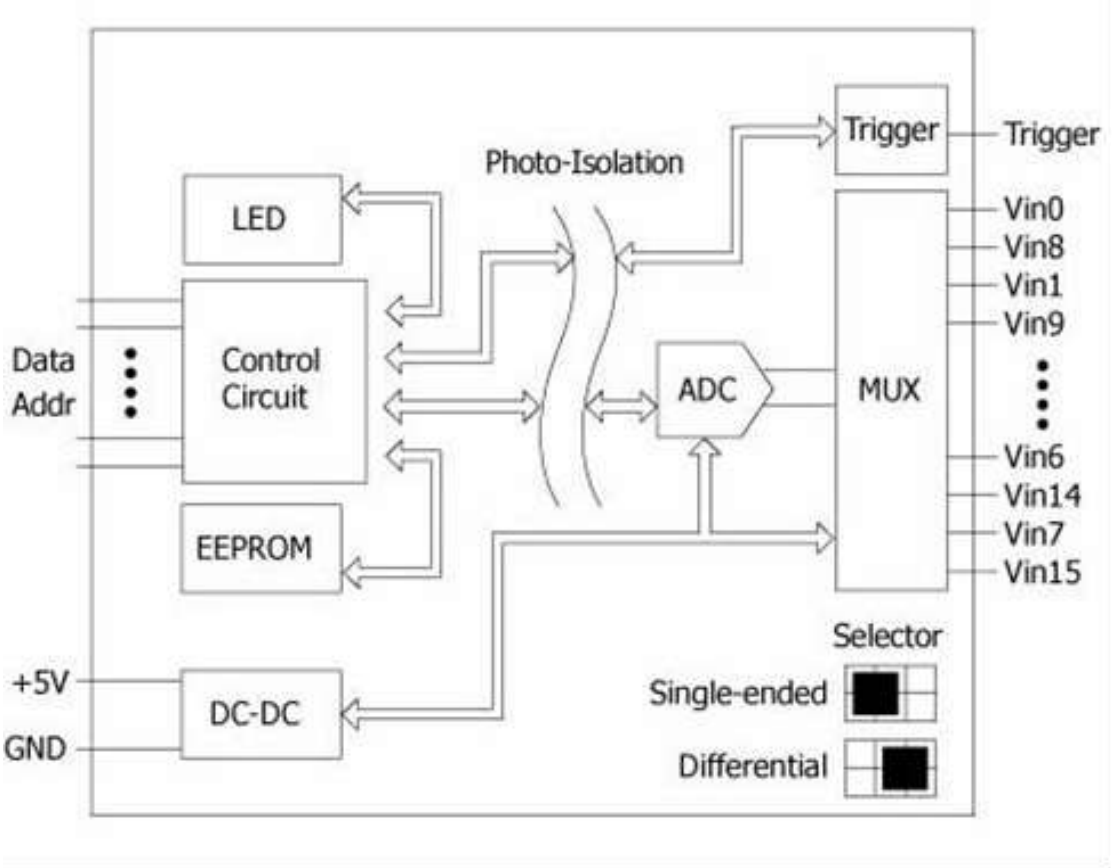
Single-ended Input Type: Current Input Wiring need to jumper at voltage input, an options external 125 Ω resistor is required.

I-9017C-15

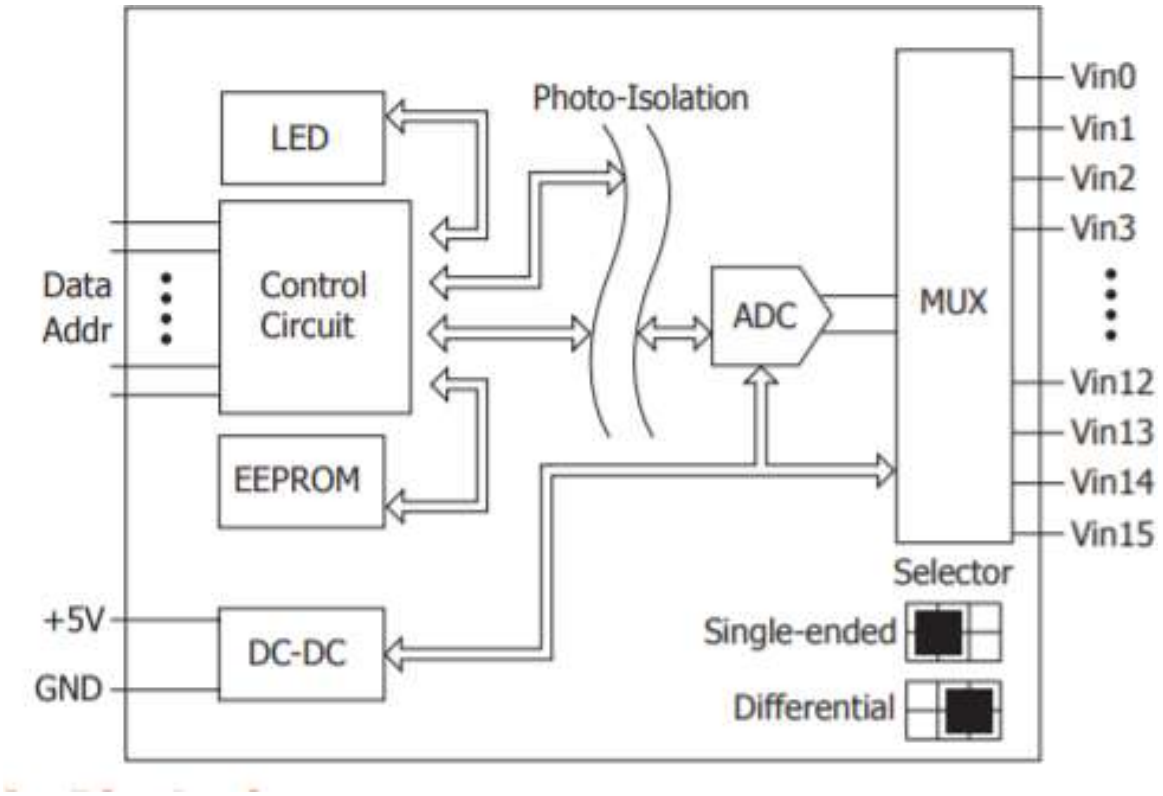
	Current Input Wiring
Differential	

1.5. Block Diagram

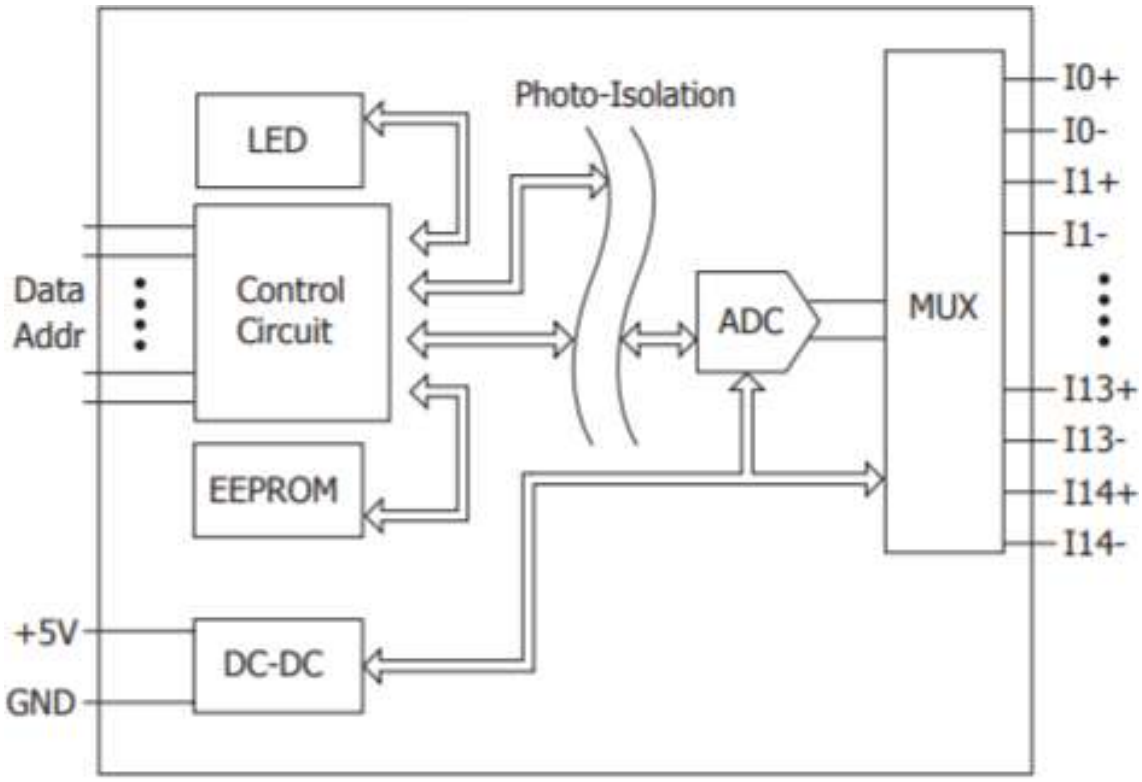
I-8017HW/I-8017DW/I-8017HCW:



I-9017/I-9017-15



I-9017C-15

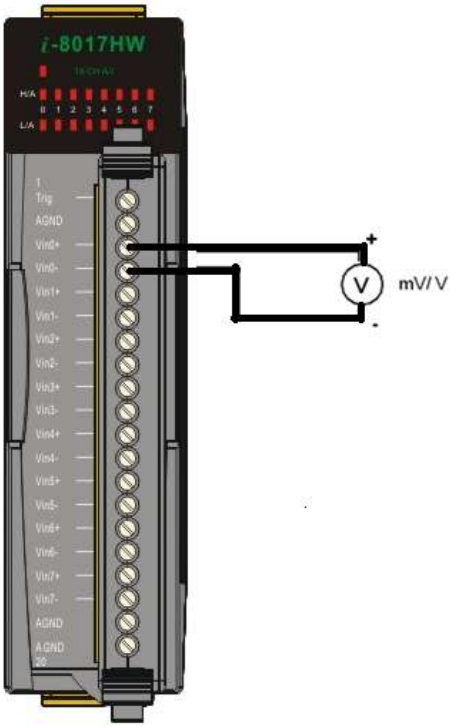


2. Quick Start

ICP DAS provides a range of demo programs for different platforms that can be used to verify the functions of the I-8017W/9017. The source code contained in these programs can also be reused in your own custom programs if needed. Both I-8017W/9017 use the same library and demo. The executable file, can be used to retrieve the basic configuration information related to the module and to verify the AI read functions. The basic configuration information includes:

- The version number and the FPGA version
- The Differential/Single-ended jumper settings
- The Gain and Offset values for each input range
- The data read from each channel

1. First, user need to download LinPAC SDK, which is 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-8017W/9017 has been plugged in the controller.
3. Refer to the Jumper Settings section. Ensure that the Differential/Single-ended selection jumper is in the Differential position.
4. Connect a stable signal source to the module (e.g., a battery output) using the differential wiring method, as illustrated below.



5. Next, check the communication between controller and PC is fine, and download the demo program files to the controller.

3. Demo Programs

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

User can find the related files in the product CD or below website:

http://www.icpdas.com/root/product/solutions/pac/linpac/linpac-8000_download.html

or

http://newweb.icpdas.com/en/download/index.php?kind=8

4. API References

ICPDAS supplies a range of C API functions for the I-8017W/9017 module. 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.

Platform	Product included	API prefix characters
Linux	I-8017/8017H/8017HW series	"I8017W_" + function name
	I-9017	"I9017_" + function name

API for I-8017HW series and I-9017 series

Function	Description
I8017_Init	Used to initialize the module
I8017_GetFirmwareVersion	Used to read the firmware (FPGA) version information
I8017_GetLibVersion	Used to read the version and build information for the currently installed Library
I8017_GetSingleEndJumper	Used to read the status of the input jumper (Differential or Single-ended mode)
I8017_ReadAI	Used to read the Analog Input value from a specific channel in float format
I8017_ReadAI_AVG	Used to read the average Analog input value from a specific channel in float format
I8017_ReadAIHex	Used to read the Analog Input value from a specific channel in 16-bit hexadecimal format
I8017_ReadAIHex_AVG	Used to read the average Analog input value from a specific channel in hexadecimal format
I8017_ReadGainOffset_Info	Used to read the calibrated voltage Gain and Offset values
I8017_Read_mA_GainOffset	Used to read the calibrated currents Gain and Offset values

API for I-9017-15 only

Function	Description
I8017_Select_SingleEnd	Used to set the Single-ended/ differential mode of I-9017-15.

API for I-8017DW

Function	Description
I8017_Get_D_Sub_Status	Used to get connector status between D sub and 8017DW.

4.1. I8017_Init

This function is used to initialize the module and must be called at least once before using any other function.

Syntax

```
short   I8017_Init(
    int slot
);
```

Parameters

slot:

specifies the slot number (1 ~ 8).

Return Values

Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
int slot;
Open_Slot(slot);

I8017_Init(slot);
```

4.2. I8017_GetFirmwareVersion

This function is used to read the firmware (FPGA) version information for the module.

Syntax

```
short   I8017_GetFirmwareVersion(
    int slot,
    short* firmware
);
```

Parameters

slot

specifies the slot number (1 ~ 8).

**firmware*

[Output]The firmware version information for the I-8017 module.

Return Values

Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
int slot;
short firmware;
Open_Slot(slot);

I8017_GetFirmwareVersion(slot, &firmware);
```

4.3. I8017_GetLibVersion

This function is used to read the version and build information for the Library.

Syntax

```
short    I8017_GetLibVersion(void);
```

Parameters

None

Return Values

The version number and build information for the Library used by the module.

Others: Refer to Appendix A: “Error Code Definitions” for more details.

Examples

[C]

```
short version;
Open_Slot(slot);

version = I8017_GetLibVersion();
```

4.4. I8017_GetSingleEndJumper

This function is used to read whether the jumper is set to either Differential or Single-ended mode.

Syntax

```
short    I8017_GetSingleEndJumper(
    int iSlot,
    short* selectJumper
);
```

Parameters

iSlot

specifies the slot number (1 ~ 8).

** selectJumper*

[Output] The status of module.

0: Differential Mode

1: Single-ended Mode

Return Values

Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
int slot, jumper;
Open_Slot(slot);

I8017_GetSingleEndJumper(slot, & jumper);
```

Note: The old version LinPAC SDK will show I8017_GetSingleEndJumper(slot) function in demo, if you want to use new I8017_GetSingleEndJumper(slot, & jumper) function, it is necessary to install the latest SDK and recompile your examples.

4.5. I8017_ReadAI

This function is used to read the Analog Input value in float format from a specific channel of the module.

Syntax

```
short I8017HW_ReadAI(
    int iSlot,
    int iChannel,
    int iGain,
    float* fValue
);
```

Parameters

iSlot

specifies the slot number (1 ~ 8).

iChannel

Specifies the channel number

iGain

Specifies the input range

0: +/- 10.0V

1: +/- 5.0V

2: +/- 2.5V

3: +/- 1.25V

4: +/- 20mA

** fValue*

[Output] the analog input value in float format.

Return Values

Others: Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
int slot, ch, gain;
float fValue;
Open_Slot(slot);

I8017_ReadAI(slot, ch, gain,& fValue);
```

4.6. I8017_ReadAI_AVG

This function is used to read the average Analog Input value in float format from the module.

Syntax

```
short   I8017_ReadAI_AVG(
    int slot,
    int iChannel,
    int iGain,
    unsigned short averageCnt,
    float* fValue
);
```

Parameters

Slot

specifies the slot number (1 ~ 8).

iChannel

Specifies the channel number

iGain

Specifies the input range

0: +/- 10.0V

1: +/- 5.0V

2: +/- 2.5V

3: +/- 1.25V

4: +/- 20mA

averageCnt

the average count for each sampling routine.

** fValue*

[Output] the analog input value in float format.

Return Values

Others: Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
int slot, ch, gain;
unsigned short cnt;
float fValue;
Open_Slot(slot);

I8017_ReadAI_AVG(slot, ch, gain, cnt,& fValue);
```

4.7. I8017_ReadAIHex

This function is used to read the Analog Input value in 16-bit hexadecimal format.

Syntax

```
short    I8017_ReadAIHex(
    int iSlot,
    int iChannel,
    int iGain,
    short* iValue
);
```

Parameters

iSlot

specifies the slot number (1 ~ 8).

iChannel

Specifies the channel number

iGain

Specifies the input range

0: +/- 10.0V

1: +/- 5.0V

2: +/- 2.5V

3: +/- 1.25V

4: +/- 20mA

** iValue*

[Output] the analog input value in hexadecimal format.

Return Values

Others: Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
int slot, ch, gain;
short hval;
Open_Slot(slot);

I8017_ReadAIHex(slot, ch, gain,& hval);
```

Note

the I-8017HW/I-8017DW/I-8017HCW/I-9017/I-9017-15/I-9017C-15 modules use a 14-bit AD chip, when user needs to scale the hexadecimal data ,it is convenient to use the I8017_ReadHex function to return a 16-bit data.

4.8. I8017_ReadAIHex_AVG

This function is used to read the average Analog Input value in 16-bit hexadecimal format.

Syntax

```
short   I8017HW_ReadAIHex_AVG(
    int slot,
    int iChannel,
    int iGain,
    unsigned short averageCnt,
    short* iValue
);
```

Parameters

slot

specifies the slot number (1 ~ 8).

iChannel

Specifies the channel number

iGain

Specifies the input range

0: +/- 10.0V

1: +/- 5.0V

2: +/- 2.5V

3: +/- 1.25V

4: +/- 20mA

averageCnt

the average count for each sampling routine.

** iValue*

[Output] the analog input value in hexadecimal format.

Return Values

Others: Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
int slot, ch, gain;
unsigned short cnt;
short hval;
Open_Slot(slot);

I8017_ReadAIHex_AVG(slot, ch, gain, cnt,& hval);
```

4.9. I8017_ReadGainOffset_Info

This function is used to read the calibrated Gain and Offset values for the I-8017 module inserted in a specific slot

Syntax

```
short   I8017_ReadGainOffset_Info(
    int iSlot,
    int iGain,
    unsigned short* iGainValue,
    short* iOffsetValue
);
```

Parameters

iSlot

specifies the slot number (1 ~ 8).

iGain

Specifies the input range

0: +/- 10.0V

1: +/- 5.0V

2: +/- 2.5V

3: +/- 1.25V

4: +/- 20mA

**iGainValue*

[Output]Specifies the calibrated Gain value

**iOffsetValue*

[Output]Specifies the calibrated Offset value

Return Values

Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
Int slot,Gain;
unsigned short GainValue;
short OffsetValue;
Open_Slot(slot);

I8017_ReadGainOffset_Info(slot, Gain,& GainValue,& OffsetValue);
```

4.10. I8017_Read_mA_GainOffset

This function is used to read the calibrated Gain and Offset values for the I-8017HCW/I-9017/I-9017C-15 module inserted in a specific slot.

Syntax

```
short   I8017_Read_mA_GainOffset(
    int slot,
    short ch,
    unsigned short* GainValue,
    short* offsetValue
);
```

Parameters

slot

specifies the slot number (1 ~ 8).

ch

Specifies the channel

Valid range :

I-8017HCW/I-9017 = 0 to 7

I-9017C-15 = 0 to 14

** GainValue*

Specifies the calibrated Gain value

** offsetValue*

Specifies the calibrated Offset value

Return Values

Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
int slot;
short ch;
unsigned short GainValue;
short OffsetValue;
Open_Slot(slot);

I8017_Read_mA_GainOffset(slot, ch,& GainValue, & OffsetValue);
```

4.11. I8017_Select_SingleEnd

This function is used to set the Single-ended/ differential mode of I-9017-15.

Syntax

```
short   I8017_Select_SingleEnd (
    int slot,
    short selection
);
```

Parameters

slot

specifies the slot number (1 ~ 8).

selection

The status of module.

0: Differential Mode

1: Single-ended Mode

Return Values

Refer to Appendix A: “Error Code” for more details.

Examples

[C]

```
int slot;
short status;
Open_Slot(slot);

I8017_Select_SingleEnd(slot, status);
```

4.12. I8017_Get_D_Sub_Status

This function is used to get connector status between D sub and 8017DW.

Syntax

```
short   I8017_Get_D_Sub_Status(
    int iSlot,
    short* D_Sub_Status
);
```

Parameters

slot

specifies the slot number (1 ~ 8).

**D_Sub_Status*

The status of D_Sub.

1 : Open

0 : Close

Return Values

Refer to Appendix A: “Error Code” for more details.

Note

function for I-8017DW module only, in the others 8017 series module,

the value of D Sub Status will always be 1.

Examples

[C]

```
int slot;
short status;
Open_Slot(slot);

I8017_Get_D_Sub_Status (slot, status);
```

5. Troubleshooting

This chapter discusses how to solve some common problems you may encounter.

This chapter contains:

- What to do when the data read from the module seems unstable

5.1. Service Request Requirements

If you are using a stable signal source to output a signal to the module, such as a battery, and are receiving incorrect or unstable data, prepare the following three items and e-mail them to service@icpdas.com

- An image of the physical wiring
- The file saved from the Basic Information tab
- The file saved from the AI Test tab

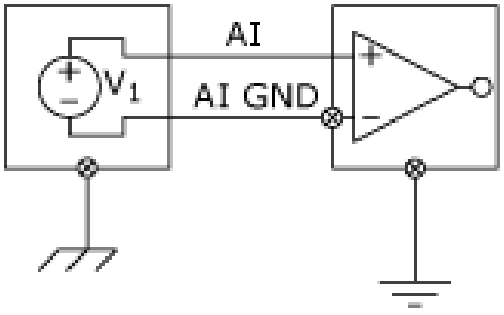
5.2. What to do when the data read from the module seems unstable

If the voltage can be measured correctly when testing using a battery, but not when using the real signal source, the error may be caused by any or all of the following factors:

- A noise-corrupted signal source
- Instability in the signal source
- A floating signal source that is not referenced to a system ground(earth or building ground)

Because of the nature of the high-speed data acquisition function on the module, any noise coupled to a signal, or any change in voltage on an unstable source, is also captured. In this situation, signal filtering or isolation should be considered in order to enhance the quality of the signal.

It is recommended that the V- pin is connected to the AGND (system ground) pin when measuring differential signals, as shown in the figure below.



Appendix A. Error Code

Error Code	Definition	Description
0	NoError	This indicates that there have been no errors
-1	ID_ERROR	There is a problem with the module ID
-2	SLOT_ERROR	There is a Slot index error (1 ~ 8)
-3	CHANNEL_ERROR	There is a Channel index error (0 ~ 15)
-4	GAIN_ERROR	There is a Gain error (0 ~ 4)
-6	NOT_SUPPORT_ERROR	The function is not support the Firmware
-7	NOT_Calibration	The module is not calibrated