ICT INTERNATIONAL

AD Node Manual

(For High Resolution Analogue & Digital Sensors)

Contents

1. Introduction

The ICT International AD Node is a LoRaWAN data transmission device, designed for those requiring precision in their analogue and digital measurements. With a 24-bit ADC, the AD-NODE supports two RTDs, a 0–10V with compressing range and a 4–20mA input. Each of the four dry-contact digital inputs is capable of simultaneously sampling at 100 Hz, with periodic reporting. Settings on the device can be altered remotely via LoRaWAN™ or locally via USB.

The device is powered by 3x lithium Energizer batteries (6.7Ah or 13.4Ah) and is charged by external 12-24V DC input – typically a 10W or 20W solar panel.

Available sensor inputs are:

- 4 x dry-contact digital sensors,
- 2 x RTD or Thermistor sensors,
- 1 x 4-20 mA sensor,
- 1 x 0-3 V Analog sensor.

Figure 1. Inside The AD-Node Box - The AD-Node Board

Figure 2. The AD-Node Box - With Antenna

2. General AD-Node Configuration

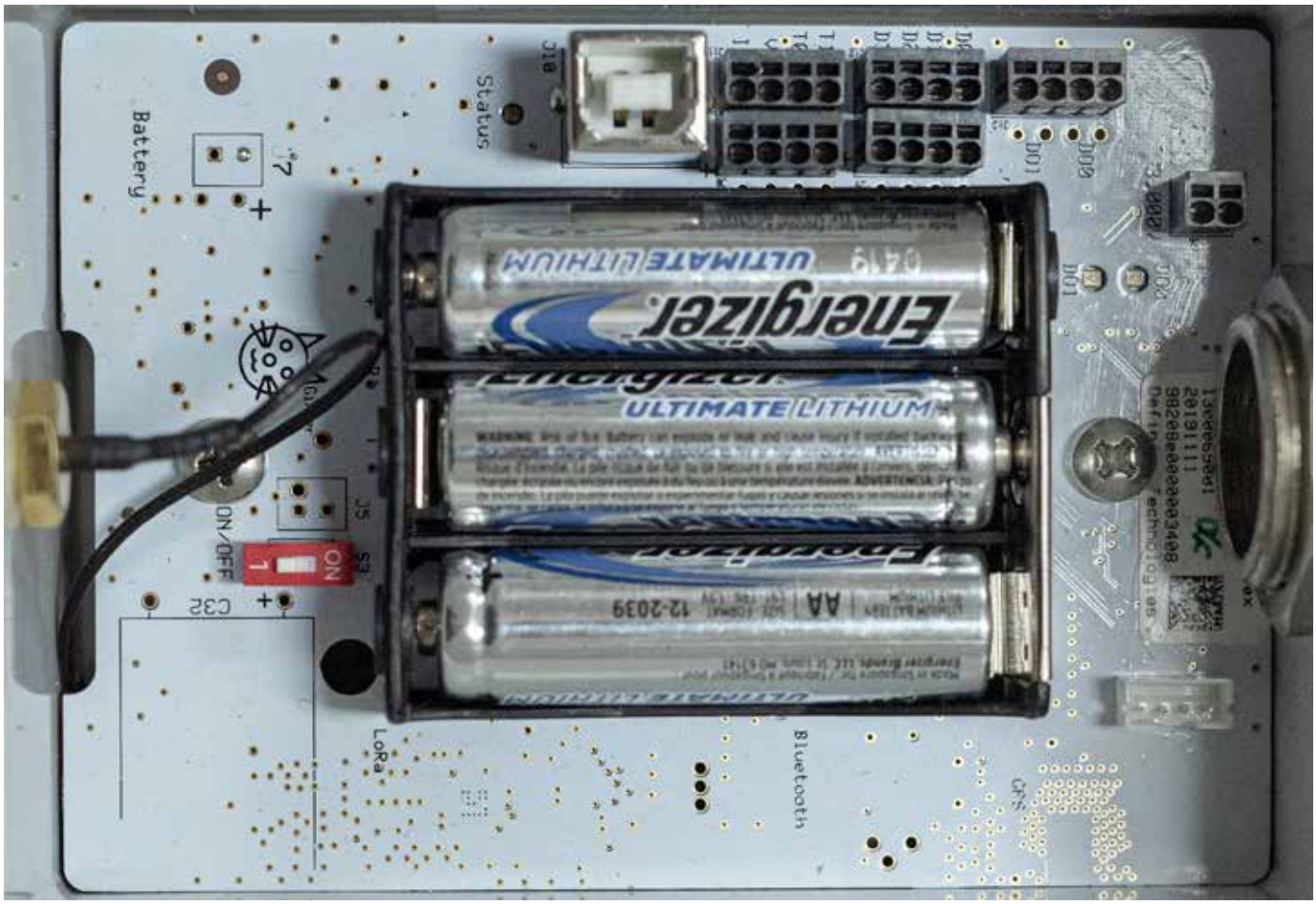
The ICT International AD Node is configured (i.e. identified and authenticated) over USB serial console using a terminal/terminal emulator. It is compatible with Windows 10, Mac OS and Linux.

One terminal emulator we recommend is Putty, which can be downloaded from https://www.putty.org/.

All commands are entered as ASCII text and will return any response as ASCII text.

Connecting an AD Node to a computer via the USB Type B-2.0, recommended serial port settings are as follows:

- Baud Rate: 115200 baud
- Bits: 8
- Parity: None
- Stop Bits: 1
- Parity: None
- Flow Control: Disabled.



2.1 Configuration Commands

These commands are entered into a terminal or terminal emulator such as Putty, https://www.putty.org/, to action several types of commands to the AD-Node.

2.1.1 Firmware Version – version

Command Input: `version`

Compatible: Device Firmware Versions > 1.2

Command Description: Returns information about the device firmware version and configured frequency.

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`version`	<String>	`Definium Technologies Pty Ltd LoRaWAN Class-A Sensor 235c0e89-dirty Luna Station 4000096001-"AS923"`

2.1.2 Save Current Device Configuration – save

Command Input: `save`

Compatible: All Device Firmware Versions

Command Description: Saves the running configuration to permanent storage.

Command Type	Syntax Used	Response Type	Example Result of Command
Action	`save`	Saved config	`Saved config`

2.1.3 Reload Saved Configuration – load

Command Input: `load`

Compatible: All Device Firmware Versions

Command Description: Saves the running configuration to permanent storage.

Command Type	Syntax Used	Response Type	Example Result of Command
Action	`load`	Loaded config	`Loaded config`

2.1.4 Reset to Factory Defaults – config reset

Command Input: `config reset`

Compatible: Device Firmware Versions > 1.2

Command Description: Resets the running configuration factory defaults.

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`config reset`	Reset app config to defaults	`Reset app config to defaults`

2.1.5 Read RTC Count – rtc

Command Input: `rtc`

Command Description: Command returns the real time clock count in seconds.

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`rtc`	<int>	`RTC=161`

2.1.6 Battery Test – battery test

Command Input: `battery test`

Command Description: Command returns the current battery voltage in mV.

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`battery test`	<int>	`BAT_MV=5179`

2.1.7 Board ID – board_id

Command Input: `board_id`

Command Description: Command returns the board ID.

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`board_id`	<int>	`BOARD=1300069001`

2.1.9 Firmware Update Mode – bootloader

Command Input: `bootloader`

Compatible: All Device Firmware Versions

Command Description: Puts the device into firmware update mode. AD-Node Firmware can be downloaded from: http://ictinternational.com/support/software/

To firmware update the node:

Install Python (make sure to add to path when prompted) - https://www.python.org/downloads/

Then run the following commands in a cmd window:

```
python -m pip install -U pip
pip install pyserial
```

Connect to the node using a terminal emulator (e.g: putty - https://www.putty.org/)

Disable the node by typing in: `enable 0`
Run command: bootloader

The device will stop flashing LED and appear to disconnect via USB.

Open windows cmd, type in: `cd (directory where you saved the firmware)`
Then (in cmd): `python windows_loader.py fw-4000097003-<frequency>.bin`
It will take 1-2 minutes, then once the firmware flash is done the USB will reconnect.

Unplug the node for ~30 seconds, then you can reconnect and reprogram the node.

Command Type	Syntax Used	Response Type
Action	`bootloader`	Node disconnects from serial interface

2.1.10 Command List – help

Command Input: `help`

Compatible: All Device Firmware Versions

Command Description: Lists all available commands with brief descriptions of their functions.

Command Type	Syntax Used	Response Type
Get	`help`	List of commands

3. Device Configuration - LoRaWAN Commands

3.1 General LoRaWAN™ Configuration

These commands are entered into a terminal or terminal emulator such as Putty, https://www.putty.org/, to action several types of commands to the AD-Node.

3.1.1 LoRaWAN EUIs – lora eui

Command Input: `lora eui`

Compatible: All Device Firmware Versions

Command Description: Manage the device's LoRaWAN EUIs (unique identifier), both device and application/ join (dependent on LoRaWAN version).

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command

Get	`lora eui <type>`	`lora eui app`
		`1122334455667788`
			<eui>	`lora eui app`
				`0x11 0x22 0x33 0x44`
				`0x55 0x66 0x77 0x88`

Set	`lora eui <type>`
	`<eui>`
		`lora eui dev`
		`1122334455667788`
			<eui>	`lora eui app`
				`0x11 0x22 0x33 0x44`
				`0x55 0x66 0x77 0x88`
				`0x11 0x22 0x33 0x44`
				`0x55 0x66 0x77 0x88`

Parameter	Type	Description
<type>	String	Type of EUI, options are: `dev`: Device EUI `app`: Application/Join EUI
<eui>	String	16 character Hexadecimal string representing an 8-byte EUI. Example: `0x11 0x22 0x33 0x44 0x55 0x66 0x77 0x88`

3.1.2 LoRaWAN Application/Join Key – lora key app

Command Input: `lora key app`

Compatible: All Device Firmware Versions

Description: Manage the device's LoRaWAN application/join key used for Over-the-Air Activation (OTAA).

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora key app`	`lora eui app 1122334455667788`	<key>	`lora key app 00112233445566778899AABBCCDDEEFF`
Set	`lora key app <key>`	`lora key app`	<key>	`lora eui app 1122334455667788AABBCCDDEEFFGGHH 1122334455667788AABBCCDDEEFFGGHH`

Parameter	Type	Description
<key>	String	32 character Hexadecimal string representing an 16-byte EUI. Example: `0x11 0x22 0x33 0x44 0x55 0x66 0x77 0x88 0xAA 0xBB 0xCC 0xDD 0xEE 0xFF 0xGG 0xHH`

3.1.3 LoRaWAN Network Joined Status – lora joined

Command Input: `lora joined`

Compatible: All Device Firmware Versions

Description: Manage the device's LoRaWAN application/join key used for Over-the-Air Activation (OTAA).

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora joined`	`lora joined`	<status>	`enabled`
Set	`lora joined <bool>`	`lora joined 0`	<status>	`lora joined 0 disabled`

Parameter	Type	Description
<bool>	Number	Network joined status to be set: `0`: Unjoined `1`: Joined
<status>	String	Network joined status string. Options: `enabled`: Joined `disabled`: Unjoined / Not Joined

3.1.4 LoRaWAN Public Network Mode – lora net public

Command Input: `lora net public`

Compatible: All Device Firmware Versions

Description: Enable/Disable public network mode. Change the sync word used by the LoRa radio between the public and private settings.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora net public`	`lora net public`	<status>	`lora net public enabled`
Set	`lora net public <bool>`	`lora net public 0`	<status>	`lora net public 0 disabled`

Parameter	Type	Description
<bool>	Number	Network joined status to be set: `0`: Disable `1`: Enabled

<status>	String	Public network status string. Options:
		`enabled`
		`disabled`

3.1.5 LoRaWAN Adaptive Data Rate – lora net adr

Command Input: `lora net adr`

Compatible: All Device Firmware Versions

Description: Devices using adaptive data rate will query the server periodically to determine the highest data rate they can transmit at and still be heard. The LoRaWAN specification states that stationary devices should use ADR, however use on mobile devise is discouraged as it is unlikely to work correctly.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora net adr`	`lora net adr`	<status>	`disabled`
Set	`lora net adr <bool>`	`lora net adr 1`	<status>	`lora net adr 1 disabled`

Parameter	Type	Description
<bool>	Number	Network joined status to be set: `0`: Disable `1`: Enabled

<status>	String	Public network status string. Options:
		`enabled`
		`disabled`

3.1.6 ADR Link Check Period– linkcheck period

Command Input: `linkcheck period`

Compatible: Device Firmware Versions > 1.2

Description: Determines the number of unconfirmed packets between link checks when using ADR. Typically, this is set automatically and should not need to be changed.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`linkcheck period`	`linkcheck period`	<period>	`64`
Set	`linkcheck period <period>`	`linkcheck period 0`	<period>	`linkcheck period 200 200`

Parameter	Type	Description
<period>	Number	Number of unconfirmed packets between link checks.

3.1.7 LoRaWAN Default Data Rate – lora data rate

Command Input: `lora data rate`

Compatible: All Device Firmware Versions

Description: Manage the device's default data rate. Used when ADR is not enabled.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora data rate`	`lora data rate`	<data rate>	`2`
Set	`lora data rate <data rate>`	`lora data rate 0`	<data rate>	`lora data rate 1 1`

Parameter	Type	Description
<data rate>	Number	LoRaWAN Data Rate, where 0 is minimum (LoRa spreading factor 12). Maximum value is region specific.

3.1.8 LoRaWAN Maximum TX Power – lora tx power

Command Input: `lora tx power`

Compatible: All Device Firmware Versions

Description: Manage the device's default maximum transmit power. This is relative to the maximum EIRP, which can be changed by the network server after the device is joined to the network.

The actual maximum transmit power will be the maximum EIRP minus 2 times this setting, in dBm, unless that value is above 20 dBm, as that is the maximum power of the device's radio.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora tx power`	`lora tx power`	<power>	`0`
Set	`lora tx power <power>`	`lora tx power 2`	<power>	`lora tx power 2 2`

Parameter	Type	Description
<power>	Number	Maximum transmit power, power below MAX_EIRP. TX_PWR = MAX_EIRP – 2 x power (dBm)

3.1.9 Confirmed Messaging – lora confirmed

Command Input: `lora confirmed`

Compatible: All Device Firmware Versions

Description: Enables/Disables LoRaWAN confirmed messaging.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora confirmed`	`lora confirmed`	<string>	`disabled`
Set	`lora confirmed <bool>`	`lora confirmed 0`	<status>	`lora net adr 0 disabled`

Parameter	Type	Description
<bool>	Number	Indicates whether to enable or disable confirmed messaging: `0`: Disable `1`: Enabled

3.1.10 LoRa Band – lora band

Command Input: `lora band`

Compatible: Device Firmware Versions

Description: Sets the LoRaWAN sub-band and default channel mask. Note that band should be set to 2 for AU915 and US915. Band selection is not applicable to AS923.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora band`	`lora band`	<string>	`lora band` `Band set to: 1`
Set	`lora band <band>`	`lora band 2`	<string>	`lora band 2` `Band set to: 2`

Parameter	Type	Description
<band>	Number	LoRaWAN Sub-band selection. Frequency plan dependant.

3.2 LoRaWAN Activation-by-Personalisation Mode

These commands are entered into a terminal or terminal emulator such as Putty, https://www.putty.org/, to action several types of commands to the AD-Node. In Activation-by-Personalisation mode the device is pre-supplied with all keys and identifiers such that it does not need to undergo an over-the-air join process.

All settings in this section are required to be manually set when using Activation-by-Personalisation (ABP) and will be retrieved during the join process for OTAA devices. Do not change these if using OTAA on the device.

3.2.1 LoRaWAN Activation-by-personalisation – lora net abp

Command Input: `lora net abp`

Compatible: All Device Firmware Versions

Command Description: Enable/Disable Activation-by-Personalisation Mode.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora net abp`	`lora net abp`	<status>	`disabled`
Set	`lora net abp`	`lora net abp 1`	<status>	`lora net abp 1 enabled`

Parameter	Type	Description
<bool>	Number	Indicates whether to enable or disable abp mode: `0`: Disable / `1`: Enabled
<status>	String	ABP mode status string. Options: `enabled` / `disabled`

3.2.2 LoRaWAN Session Keys – lora key session app

Command Input: `lora key session app`

Compatible: All Device Firmware Versions

Command Description: Manage the device's application and network session keys.

Type	Syntax Used	Response Type	Example Result of Command
Get	`lora key session app <type>`	<key>	`1122334455667788AABBCCDDEEFFGGHH`
Set	`lora key session app <type> <key>`	<key>	`lora key session app 1122334455667788AABBCCDDEEFFGGHH` `1122334455667788AABBCCDDEEFFGGHH`

Parameter	Type	Description
<key>	String	32 character Hexadecimal string representing an 16-byte EUI. Example: `0x11 0x22 0x33 0x44 0x55 0x66 0x77 0x88 0xAA 0xBB 0xCC 0xDD 0xEE 0xFF 0xGG 0xHH`
<type>	String	Key Types include: `app` (Application Session Key) `nwk` (Network Session Key)

3.2.3 LoRaWAN Device Address – lora net dev addr

Command Input: `lora net dev addr`

Compatible: Device Firmware Versions

Command Description: Get/Set the device's LoRaWAN network address.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora net dev addr`	`lora net dev addr`	<eui>	`11223344`
Set	`lora net dev addr <eui>`	`lora net dev addr`	<eui>	`lora net dev addr 11223344 11223344`

Parameter	Type	Description
<eui>	String	8 character Hexadecimal string representing an 4-byte EUI. Examples: `0x11 0x22 0x33 0x44` `11223344`

3.2.4 LoRaWAN Network Indentifier – lora net id

Command Input: `lora net id`

Compatible: Device Firmware Versions

Command Description: Get/Set the network identifier of the connected LoRaWAN network.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`lora net id`	`lora net dev addr`	<eui>	`11223344`
Set	`lora net id <eui>`	`lora net dev addr`	<eui>	`lora net id 11223344 11223344`

Parameter	Type	Description
<eui>	String	8 character Hexadecimal string representing an 4-byte EUI. Examples: `0x11 0x22 0x33 0x44` `11223344`

3.3 Timing

Commands for managing the device's state transition timings.

3.3.1 Maximum retransmit time after failed communication – backoff max

Command Input: `backoff max`

Compatible: Device Firmware Versions > 1.2

Command Description: Maximum wait time on failed communications before retrying.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`backoff max`	`backoff initial`	<string>	`Maximum Backoff Time: 86400 sec`
Set	`backoff max <time>`	`backoff initial 90`	<string>	`backoff initial 90000 Initial Backoff Time: 90000 sec`

Parameter	Type	Description
<time>	Number	Wait time in seconds.

3.3.2 Period between reports – report period

Command Input: `report period`

Compatible: Device Firmware Versions > 1.2

Command Description: Initial wait time on failed communications before retrying. Doubles each failure until it reaches backoff max.

Command Type	Syntax Used	Example Command	Response Type	Example Result of Command
Get	`report period`	`report period`	<string>	`Report Period: 900 sec Current: 12 Last: 0 Next:0`
Set	`report period <period>`	`report period 600`	<string>	`report period 600 Report Period: 600 sec Current: 26 Last: 0 Next:0`

Parameter	Type	Description
<period>	Number	Time in seconds between reports.
<time>	Number	Current device timestamp
<last>	Number	Timestamp that state last triggered at
<next>	Number	Timestamp of next state trigger

4. Device Configuration - Digital Command

4.1 Digital Input Command

4.1.1 Test Digital Inputs – counter

Command Input: `counter`
Command Description: Displays current values of the digital inputs

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`counter`	<int>	`COUNTER=1,2,3,4`

5. Device Configuration - Analog Commands

5.1 RTD/Thermistor Temperature

5.1.1 NTC thermistor Beta value – ntc beta

Command Input: `ntc beta`
Command Description: Configuring the steinhart-hart equation to use the inserted beta value (e.g 3990 for a THERM-SS or THERM-EP).

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`ntc beta`	<int>	`NTC Beta: 3976`
Set	`ntc beta 3990`	<int>	`NTC Beta: 3990`

5.1.2 NTC Thermistor Reference Temperature – ntc t0

Command Input: `ntc t0`
Command Description: Temperature in Kelvin of the thermistor at the measured reference temperature (e.g. the R25°C value from the thermistor's specification sheet).

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`ntc t0`	<int>	`NTC T0: 298.15`
Set	`ntc t0 298.15`	<int>	`NTC T0: 298.15`

5.1.3 NTC thermistor reference resistance – ntc r0

Command Input: `ntc r0`
Command Description: Resistance in Ohms of the thermistor at the measured reference temperature (e.g. the R25°C value from the thermistor's specification sheet).

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`ntc r0`	<int>	`NTC R0: 10000`
Set	`ntc r0 10000`	<int>	`NTC R0: 10000`

5.1.4 Test temperature – ntc test

Command Input: `ntc test`
Command Description: Returns current thermistor temperature from inputs T0 and T1 (Example result below T0=21.353°C and T1=8.728°C).

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`ntc test`	<int>	`NTC=21535,8728`

5.2 4-20 mA Input Commands

5.2.1 4-20mA Reading – adc ma

Command Input: `adc ma`
Command Description: Returns signal on the 4 - 20mA input in microamps (mA).

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`adc ma`	<int>	`uA:4950`

5.2.2 4 -20mA 12v Settle Delay – settle delay

Command Input: `settle delay`
Command Description: Delay in milliseconds for 12v rail to warm-up settle delay.

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`settle delay`	<int>	`500`
Set	`settle delay 1000`	<int>	`1000`

5.3 0-1.5 Volt Input Commands

5.3.1 0-1.5 Volt Reading – adc v

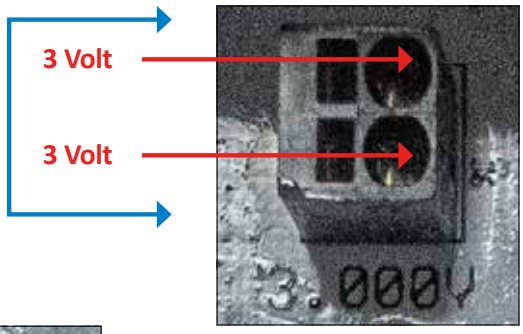
Command Input: `adc v`
Command Description: Returns current reading on the ADC in microvolts (µV).

Command Type	Syntax Used	Response Type	Example Result of Command
Get	`adc v`	<int>	`ADC COUNTS:11775`

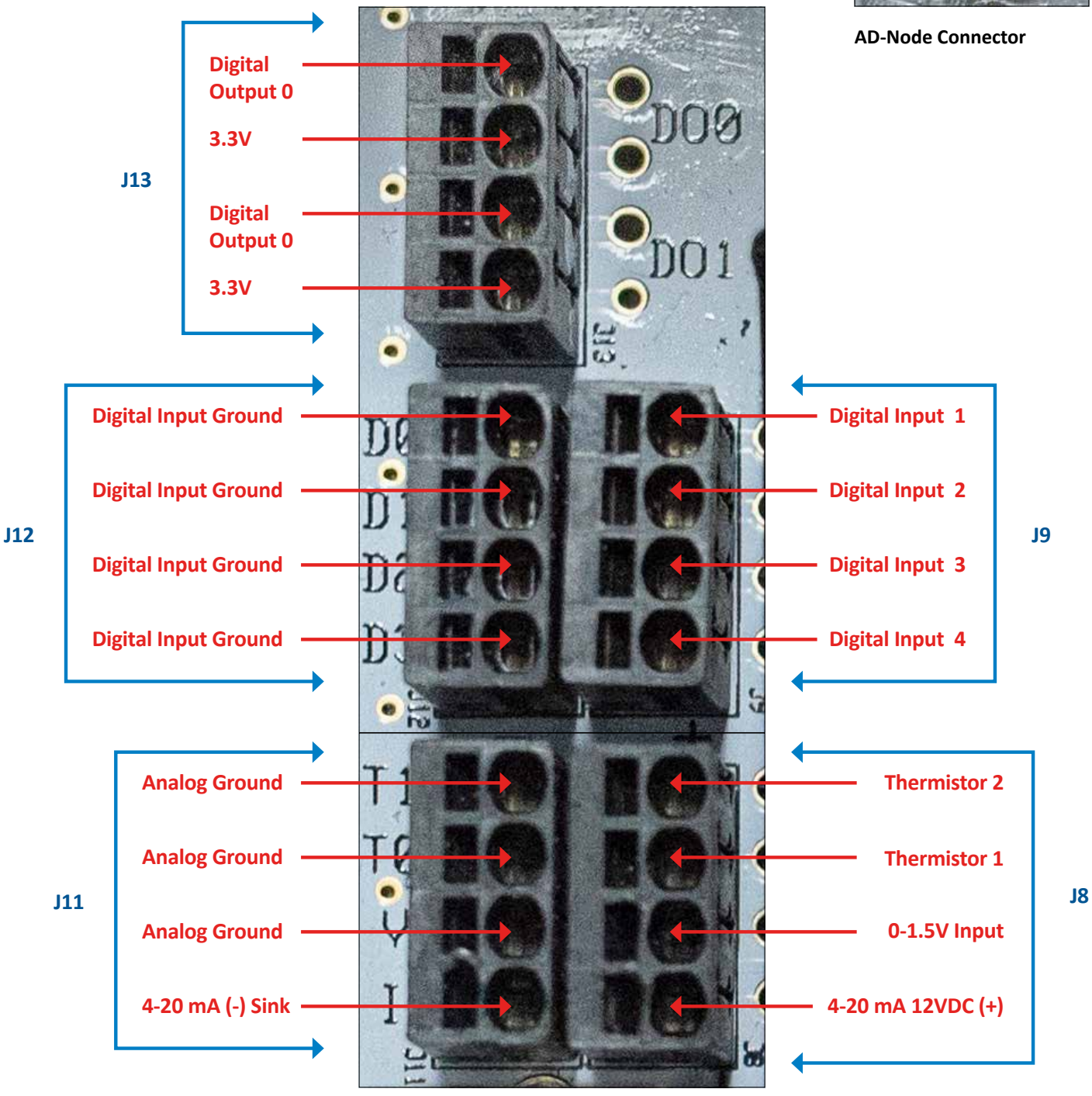
6. Connecting Sensors and Power

6.1 Sensors & Power Connector Locations

The AD-Node has 4 Digital Inputs (D0, D1, D2, D3), two thermistor inputs (T1, T0), a voltage input and a current (4-20mA) input.



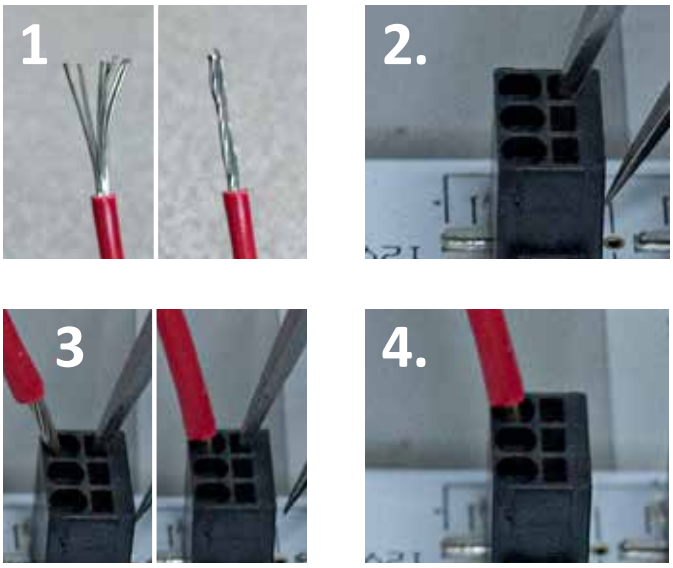
AD-Node Connector



AD-Node Sensor Connectors

6.2 Prepare, Insert & Re-Insert Power/Sensor Wires

1. Prepare the wire: Strip the outer insulation to expose the wire. Lightly twist if stranded. Ensure exposed wire has maximum length of 7mm.
2. Use a pair of tweezers or suitable flathead screwdriver to push into the square hole beside the desired socket to temporarily release the spring clamp.
3. Hold the tweezers down and insert the prepared wire. Push the wire in, down all the way and remove the tweezers.
4. The wire should be clamped in. After removing tweezers, ensure wire is secured correctly with a simple tug test. The wire can be removed or repositioned by a simple reversal of these steps.

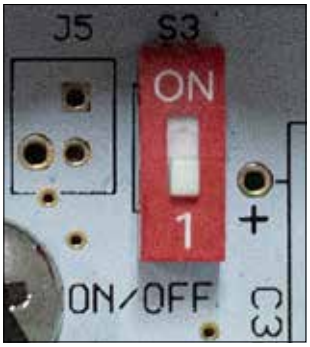


6.3 Powering The AD-NODE

The node runs off 3x 1.5V AA batteries – ICT International recommends Energizer Ultimate Lithium or similar high-end batteries.

With the batteries correctly inserted, move the switch to the ON position to begin transmitting.

AD-Node Power - Included



AD-Node Power - ON

6.4 LED Position

The LED is used to indicate the following:

- **LIGHT BLUE**: Joining Network
- **DARK BLUE**: Network Joined / Taking measurement
- **ORANGE**: Transmitting sensor data
- **PURPLE**: Measurement Complete
- **GREEN**: USB Idle
- **RED**: Failed to Join Network

AD-Node LED Light Position

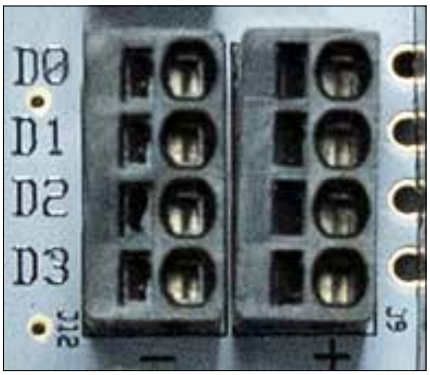
6.3 Connecting Digital (Pulse) Sensor Inputs

Digital inputs are connected between the left and right connectors labelled D0 – D3. As digital sensors are not polarised, either wire can be inserted into either connector.

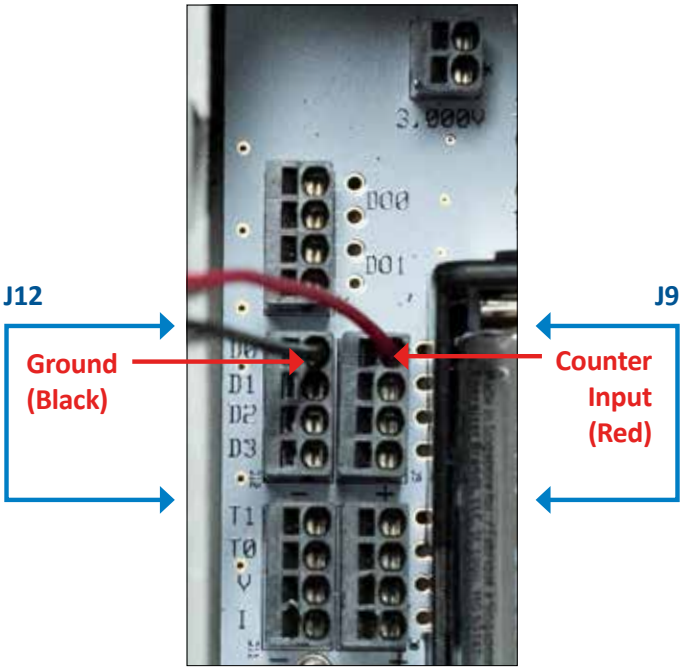
Voltage and Current sensors are connected with the excitation wire on the right hand bank (labelled with a +), sensor output/signal on the left hand bus (labelled with a -) and sensor ground connected to any digital input on the left side – this acts as ground. Wire signal to a counter input on J9, wire ground to a GND on J12.

Wire the sensor according to the sensor manual. Most Digital Pulse sensors (rain gauges, anemometers, etc.) are passive and should be wired between an input (D0 to D3) and ground.

If excitation is required, use the 3V pin.

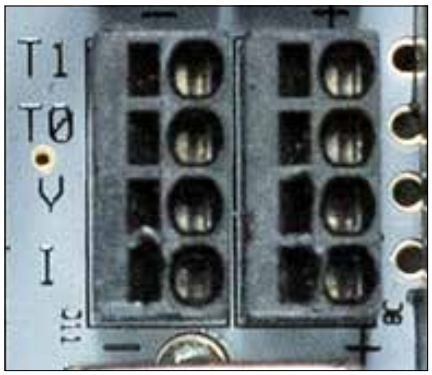


Digital Input Connectors in J12 and J9; Rows D0, D1, D2, D3

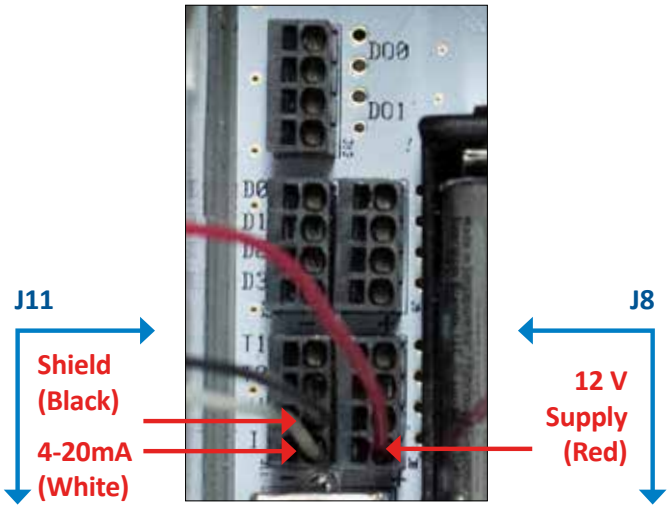


Wiring Example of:
Tipping Bucket Rain Gauge

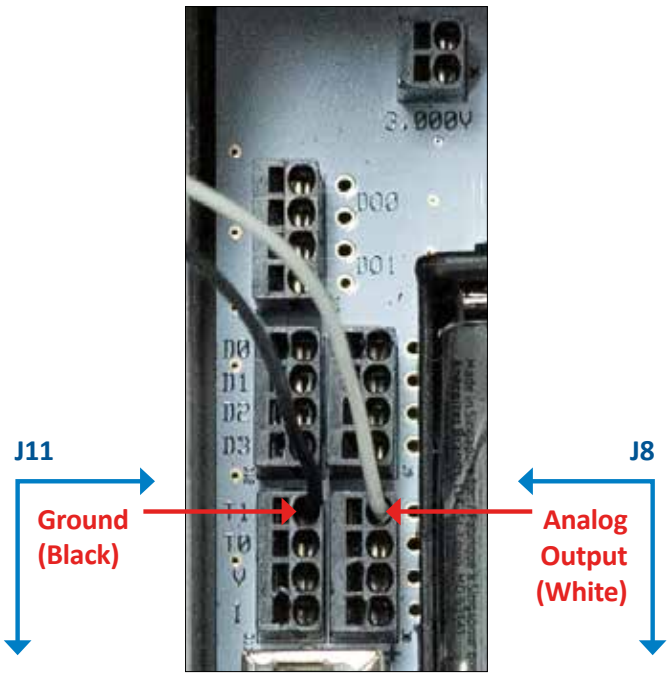
6.4 Connecting Analog Sensor Inputs



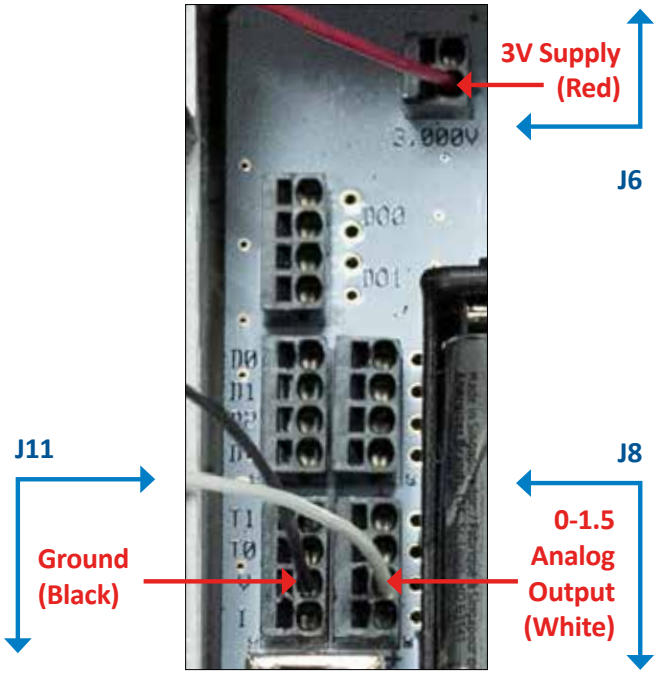
Analog Input Connectors in J11 and J8; Rows T1, T0, V, I



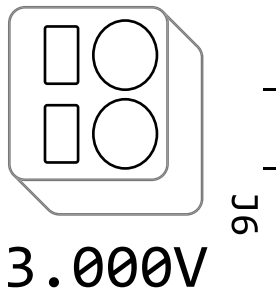
Wiring Example of a 4-20mA Sensor

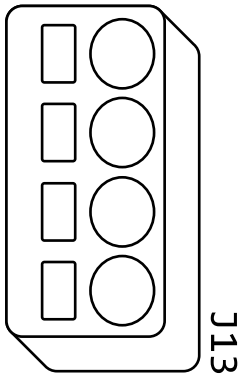


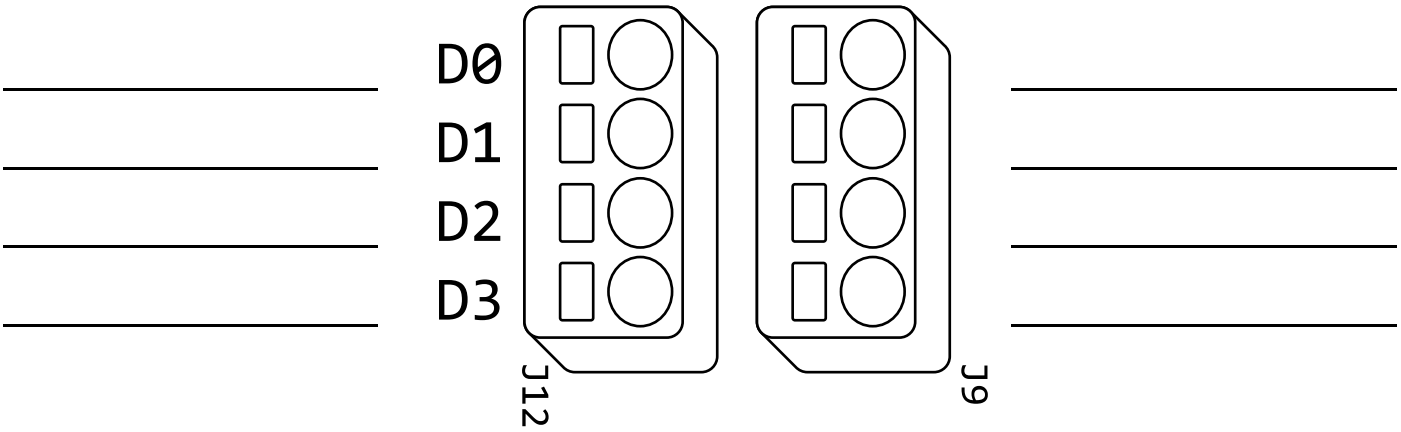
Wiring Example of a Thermistor

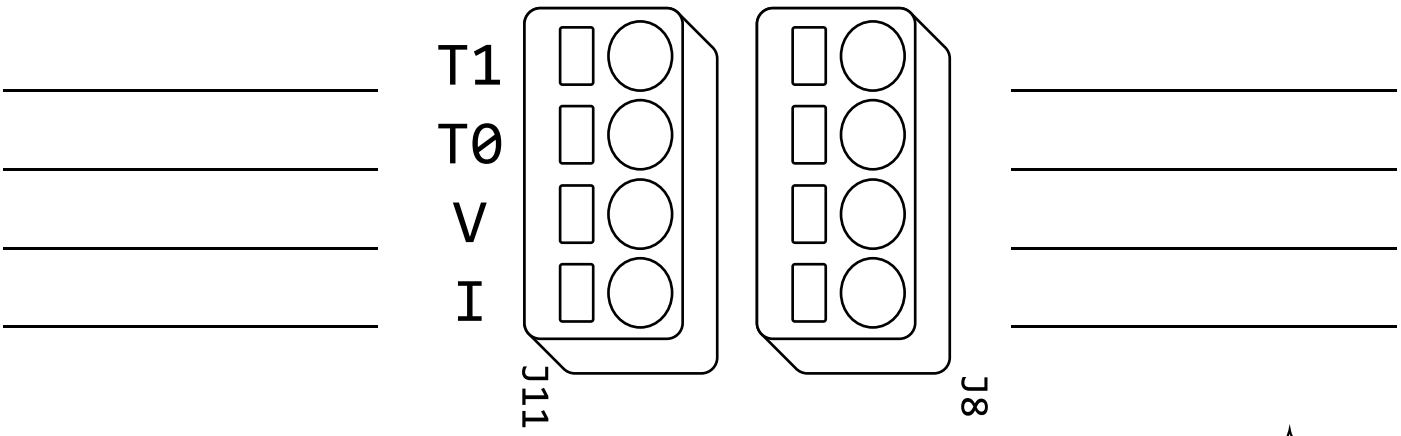


Wiring Example of a Voltage Sensor


J6
3.000V


DO0
DO1
J13


D0
D1
D2
D3
J12
J9


T1
T0
V
I
J11
J8

7. LoRaWAN Packet Structure

LoRaWAN packets sent by the AD-Node use the following format.

- fPort contains the LoRaWAN fPort number
- if (port == 0 || port == 1) Will return all variables such as RTC, Battery mV, Inputs: uA, uV, Temperature 1 and 2, and C1 to C4 inputs.
- (fPort == 10) Will return Device Type, Product ID, Batch Number and Software Version.
- (fPort == 100) Is the Downlink Response

- Device Uptime in Seconds – 4 Bytes
- Device Battery Level in millivolts – 2 Bytes
- 4 – 20 mA Reading – 2 Bytes
- 0 – 16.5 V Reading – 4 Bytes
- Thermistor 1 – 4 Bytes
- Thermistor 2 – 4 Bytes
- Counter Input 1 – 4 Bytes
- Counter Input 2 – 4 Bytes
- Counter Input 3 – 4 Bytes
- Counter Input 4 – 4 Bytes

An example payload decoder is available at the end of the manual, section 8.

8. Decoder Notes

Please contact ICT International for the AD-NODE decoder applicable to your order and suitable for TTN (https://www.thethingsnetwork.org/):

Enabling better global research outcomes in soil, plant & environmental monitoring.

www.ictinternational.com sales@ictinternational.com.au
Ph: +61 2 6772 6770 | 211 Mann St., Armidale, NSW 2350, Australia