LinPAC AM335x Series
User Manual

V2.0.0 Jan 2019

LP-22xx/LP-52xx Series

LP-8x2x Series

LP-9x2x Series

Warranty

All products manufactured by ICP DAS Inc. are warranted against defective materials for a period of one year from the date of delivery to the original purchaser.

Warning

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

Copyright

Trademark

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

Contact US

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

You can count on us for quick response.

Email: service@icpdas.com

Contents

 E-mail: service@icpdas.com

1. Introduction

This chapter introduces the fundamental concepts for the user with the LinPAC AM335x series.

LinPAC AM335x series is the new generation Linux-based PAC (Programmable Automation Controller) from ICP DAS and is equipped with a Cortex-A8 CPU (1.0 GHz) running a Linux kernel 3.x operation system, multiple communication interfaces and slots for high performance parallel I/O modules and serial I/O modules.

Main advantage of the LinPAC AM335x PAC is its high quality control system, including its stably properties, open source and the standard LinPAC SDK for Windows and Linux using the GNU C language, JAVA and GUI software. The main purpose of LinPAC AM335x PAC is to allow the numerous enthusiastic Linux users to control their own embedded system easily within the Linux environment.

ICP DAS also provides a library file, libi8k.a, custom applications can easily be developed for the LinPAC AM335x PAC using either C or Java and .NET applications will also be supported in the future. The various functions contained in the library are divided into sub-group functions for ease of use within the different applications.

1.1. Features

The LinPAC AM335x PAC offers the most comprehensive configuration and remote system upgrade solutions to meet specific application requirements. The following list shows the hardware and software features designed to simplify installation, configuration and application.

- **Powerful CPU Module**
  - AM335x ARM Cortex-A8 (1.0 GHz)
- **Memory Size**
  - SDRAM (512 MB DDR3)
  - Flash (512 MB)
  - MRAM (128 KB) for LP-8x2x/9x2x
  - EEPROM (16 KB for LP-22xx/8x2x/9x2x, 64 KB for LP-52xx)
  - SD Card (support up to 32 GB)
- **Linux OS**
  - Linux kernel 3.2.14
- **Real-Time Capability**
- **64-bit Hardware Serial Number for Software Protection**
- **Rich I/O Expansion Ability**
  - RS-232/RS-485
  - USB
  - FRnet
  - CAN
  - XV-Board for LP-22xx/52xx
- **10/100/1000 Mbit/s Ethernet Port**
- **Redundant Power Input for LP-8x2x/9x2x**
- **GSM/GPS/3G System for LP-5231PM-3GWA**
- **GSM/GPS/3G/4G System for LP-5231PM-4GE/LP-5231M-4GC**
- **Operating Temperature: -25 ~ +75°C**

For full details of specifications which can be found at:

LinPAC-22xx series:

http://ftp.icpdas.com.tw/pub/cd/linpac/napdos/lp-2000/lp-2x41/lp-2241/document/data_sheet/

LinPAC-52xx series:

http://ftp.icpdas.com.tw/pub/cd/linpac/napdos/lp-5000/lp-52xx/lp-5231/document/data_sheet/

LinPAC-8x2x series:

http://ftp.icpdas.com.tw/pub/cd/linpac/napdos/lp-8x2x/document/data_sheet/

LinPAC-9x2x series:

http://ftp.icpdas.com.tw/pub/cd/linpac/napdos/lp-9x2x/document/data_sheet/

Please note:

- **The flash and microSD disk have a finite number of program-erase cycles. Important information should always be backed up on other media or storage device for long-term safekeeping.**
- **The Li-batterie can continually supply power to the 512 KB SRAM to retain the data for 10 years (It is recommended that batteries are changed each 5~7 year.)**

2. LinPAC AM335x Getting Started

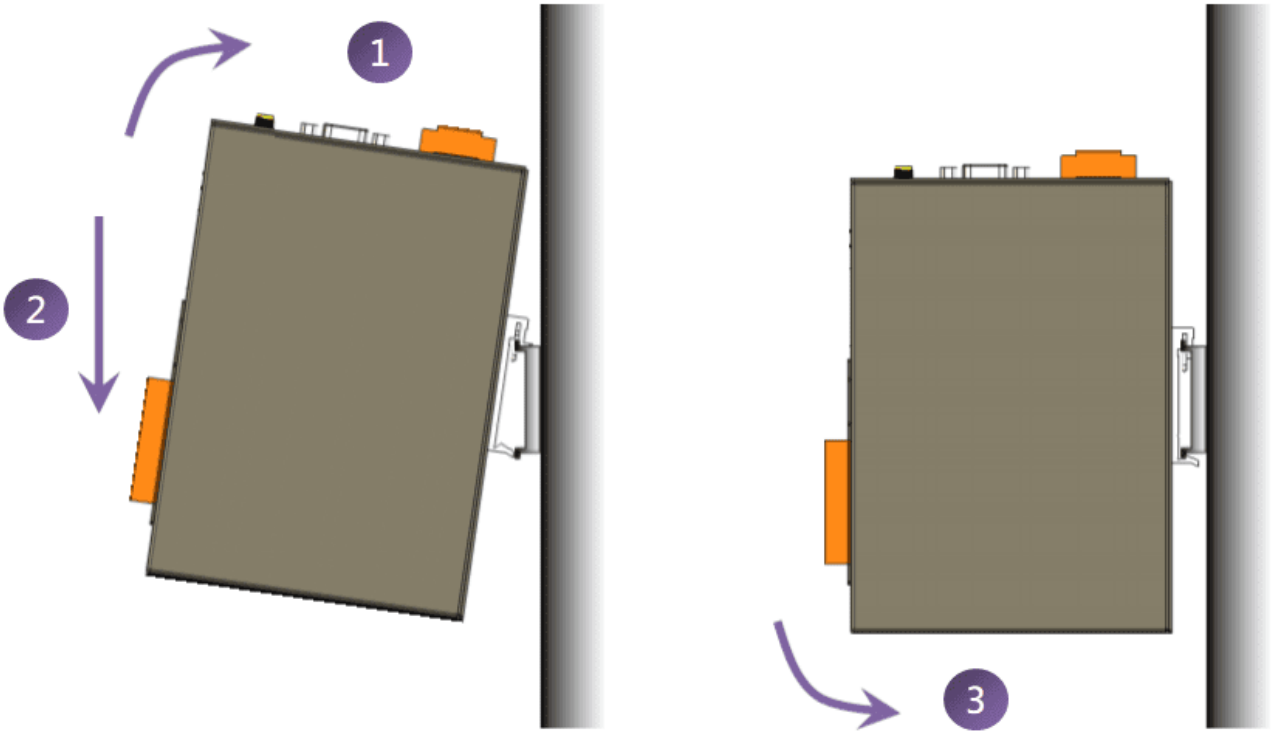
This chapter provides a guided tour of the LinPAC AM335x series PAC installation and configuration that describes the steps needed to download, install, configure, and run the basic procedures for the user working with the in LinPAC AM335x PAC for the first time.

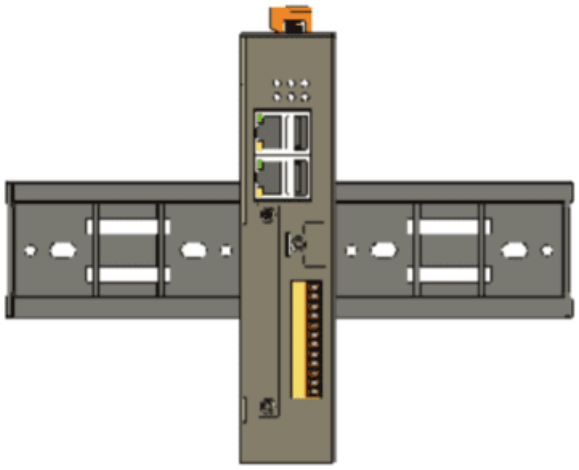
2.1. Mounting the Hardware

2.1.1. Mounting the LP-22xx

DIN-Rail mounting

The LP-2241 has simple rail clips for mounting reliably on a standard 35 mm DIN-Rail.

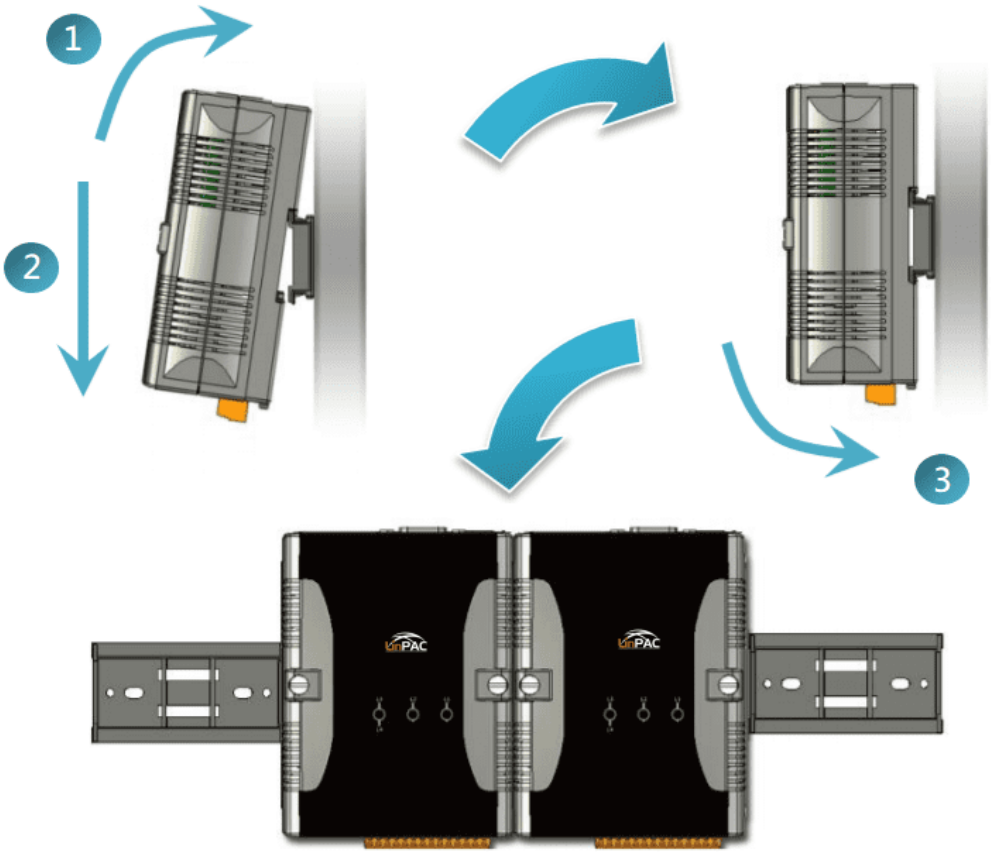




2.1.2. Mounting the LP-52xx

DIN-Rail mounting

The LP-5231 has simple rail clips for mounting reliably on a standard 35 mm DIN-Rail.



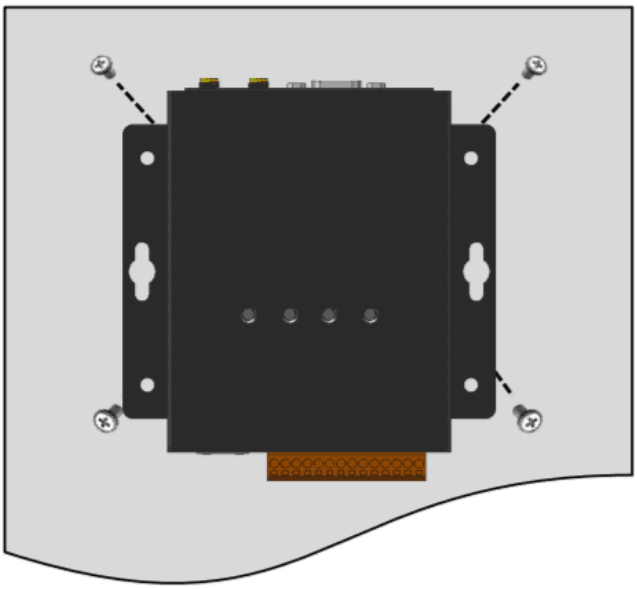
Remove the LP-5231 from the DIN-Rail

Wall/Panel mounting

The LP-5231M/LP-5231PM-3GWA/LP-5231PM-4GE/LP-5231PM-4GC can be mounted either directly to a wall/panel.

Step 1: Install the four mounting screws into the 4 keyhole mounting holes.

Step 2: Fasten the screws securely.



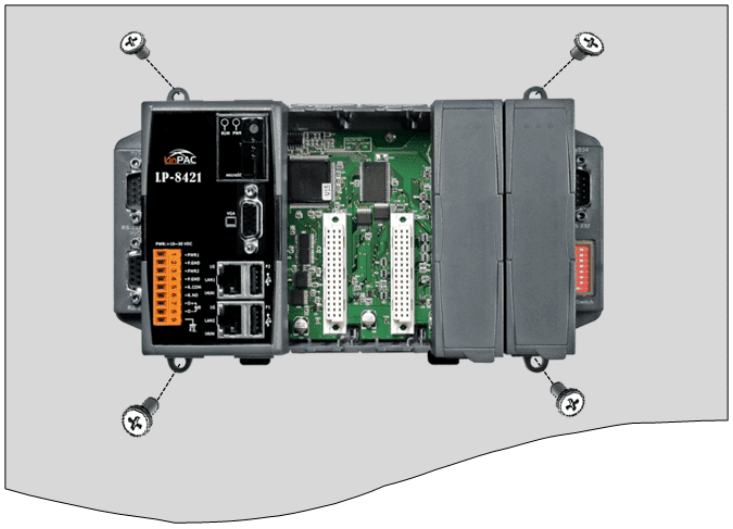
2.1.3. Mounting the LP-8x2x

Wall/Panel mounting

The LP-8x2x can be mounted either directly to a wall/panel or onto a standard 35mm DIN-Rail.

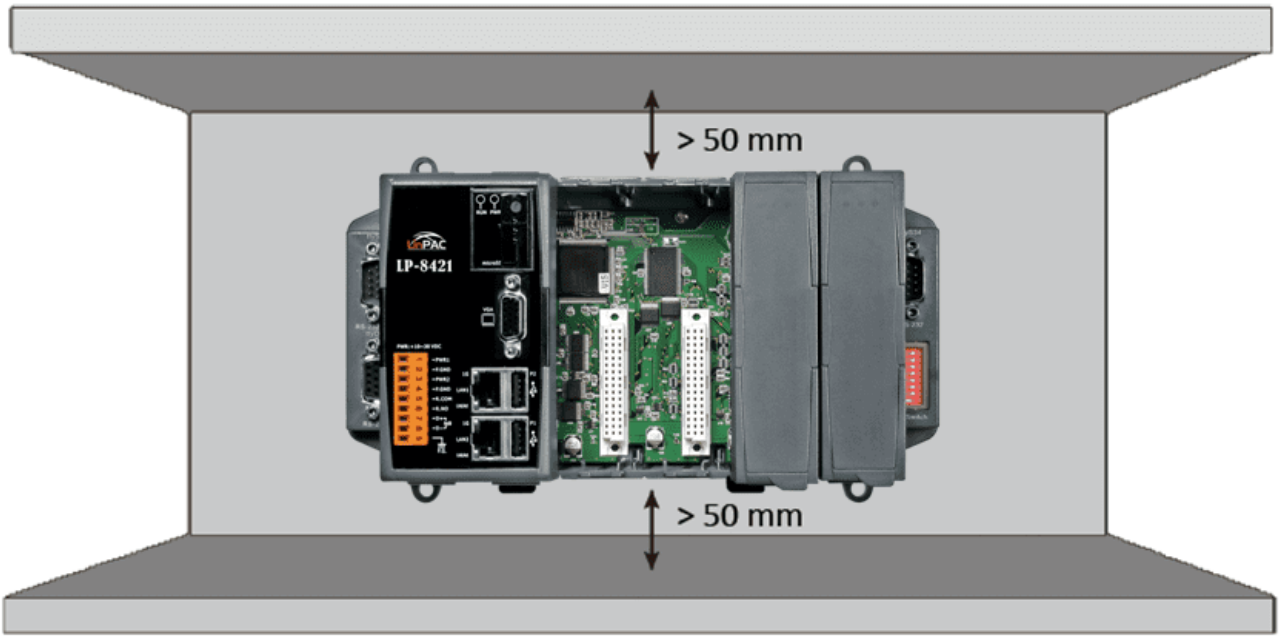
Step 1: Install the four mounting screws into the 4 keyhole mounting holes.

Step 2: Fasten the screws securely.



Tips & Warnings

There must be a minimum clearance of 50mm between the LP-8x2x and the top and bottom side of the enclosure panel.

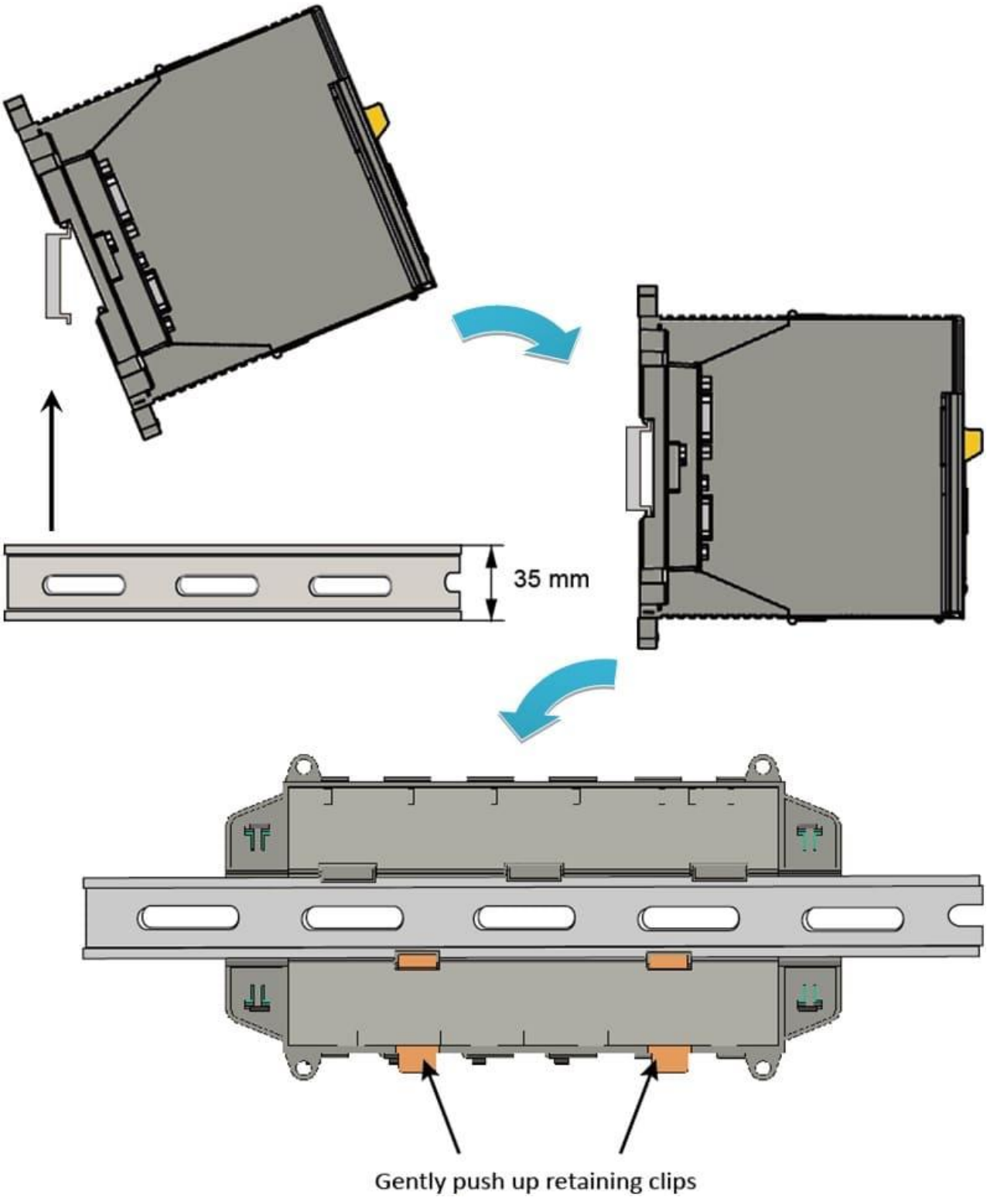


 E-mail: service@icpdas.com

DIN-Rail mounting

Step 1: Hook upper tab over upper flange of DIN-Rail.

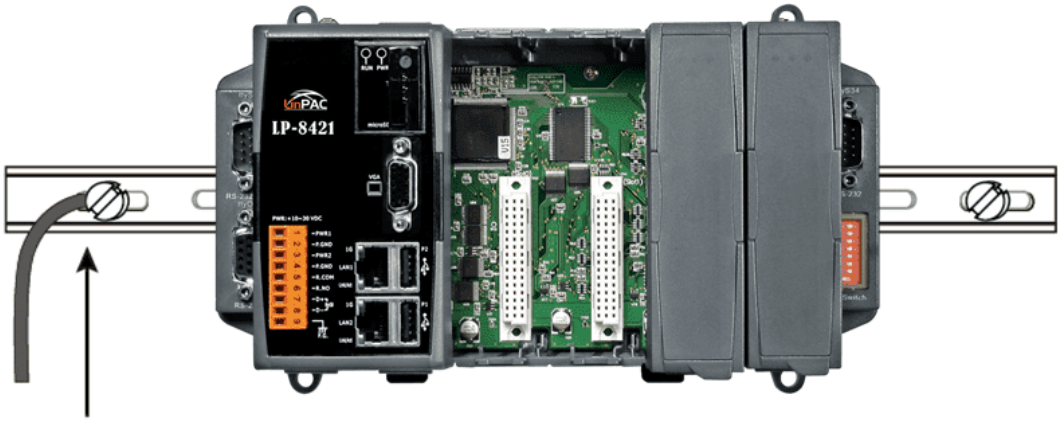
Step 2: Tilt the module toward DIN-Rail until it snaps securely to DIN-Rail.



 E-mail: service@icpdas.com

Tips & Warnings

A good common ground reference (earth ground) is essential for proper operation of the LP-8x2x. One side of all control circuits, power circuits, and the ground lead must be properly connected to earth ground by either installing a ground rod in close proximity to the enclosure or by connecting to the incoming power system ground. There must be a single-point ground (i.e. copper bus bar) for all devices in the enclosure that require an earth ground.



Connect the ground lead to the ground screw

 E-mail: service@icpdas.com

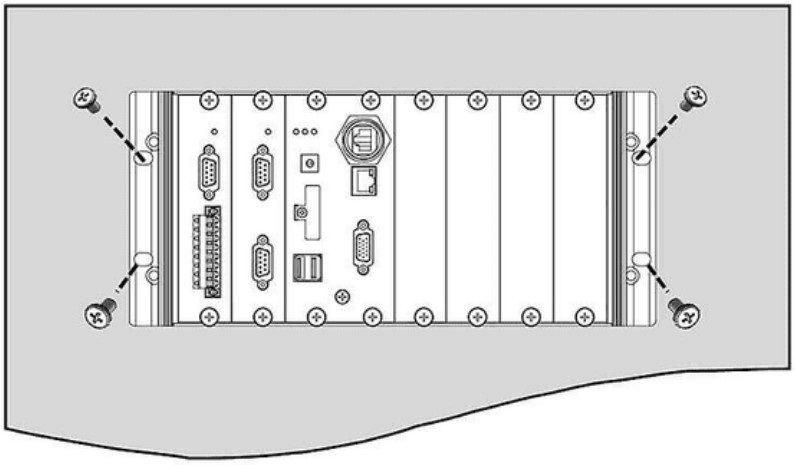
2.1.4. Mounting the LP-9x2x

Wall/Panel mounting

The LP-9x2x can be mounted either directly to a wall/panel, or onto a stainless 35mm DIN-Rail.

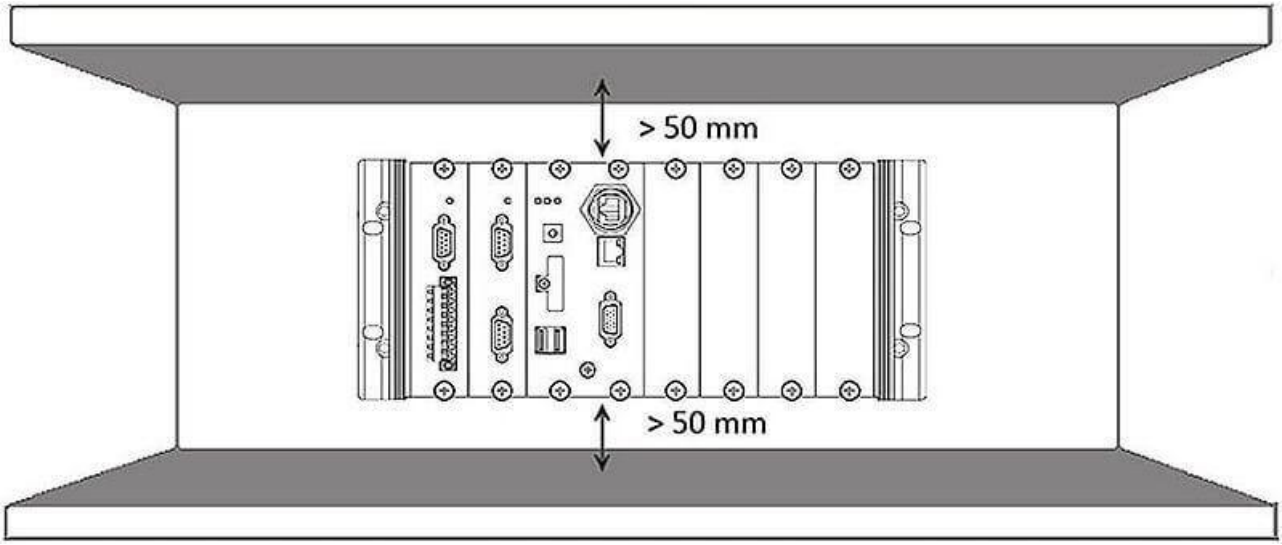
Step 1: Install the four mounting screws into the 4 keyhole mounting holes.

Step 2: Fasten the screws securely.



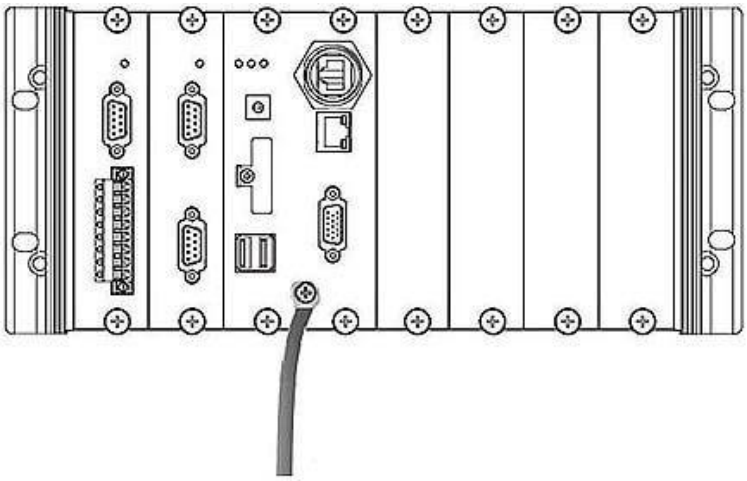
Tips & Warnings

There must be a minimum clearance of 50 mm between the LP-9x2x and the top and bottom side of the enclosure panel.



E-mail: service@icpdas.com

Step 3: Connect the ground lead to the frame ground point.

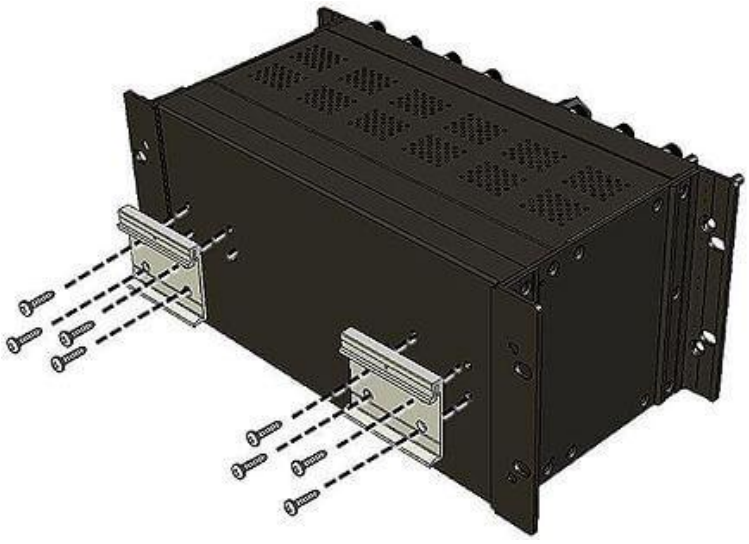


Tips & Warnings

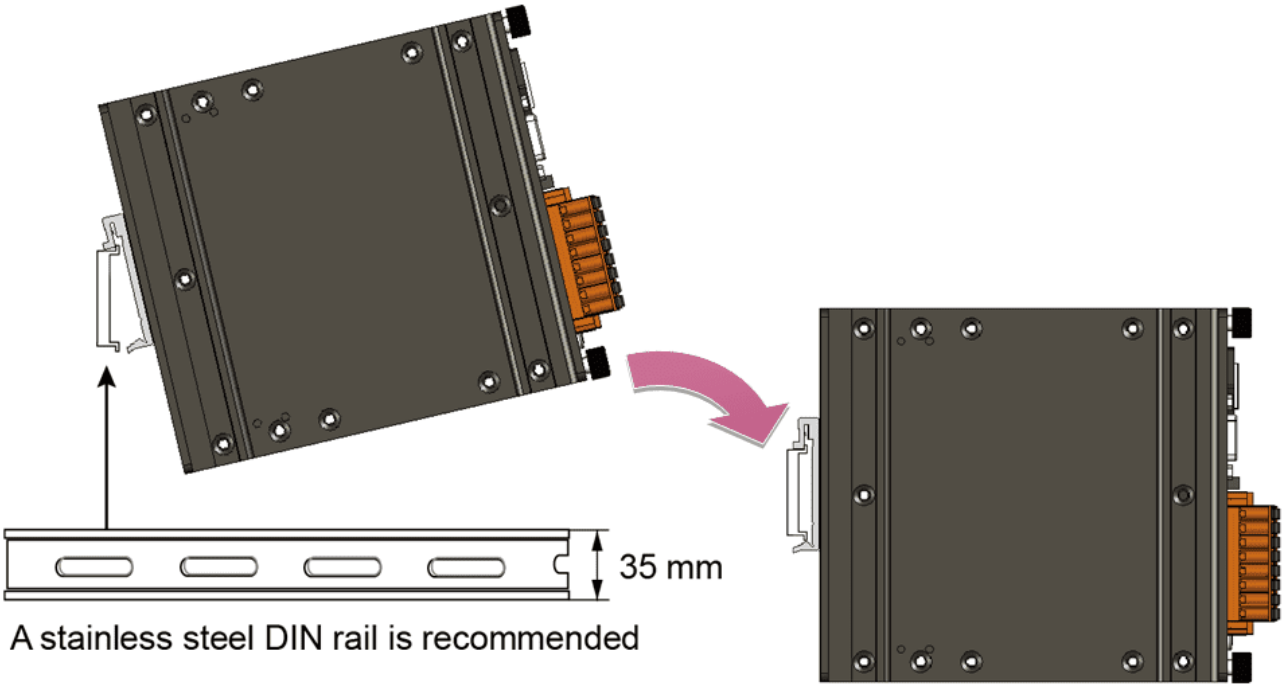
A good common ground reference (earth ground) is essential for proper operation of the LP-9x2x. One side of all control circuits, power circuits, and the ground lead must be properly connected to earth ground by either installing a ground rod in close proximity to the enclosure or by connecting to the incoming power system ground. There must be a single-point ground (i.e. copper bus bar) for all devices in the enclosure that require an earth ground.

DIN-Rail mounting

Step 1: Fasten the DIN-Rail clip to the LP-9x2x.

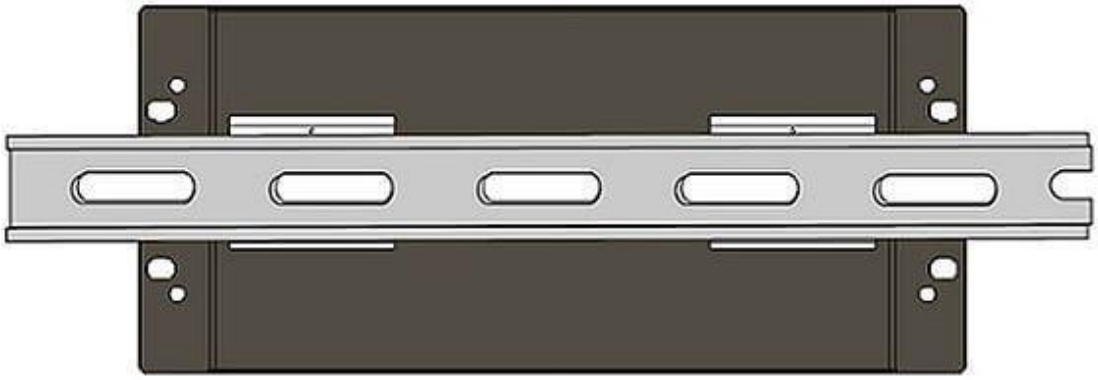


Step 2: Clip the device onto a stainless DIN-Rail.

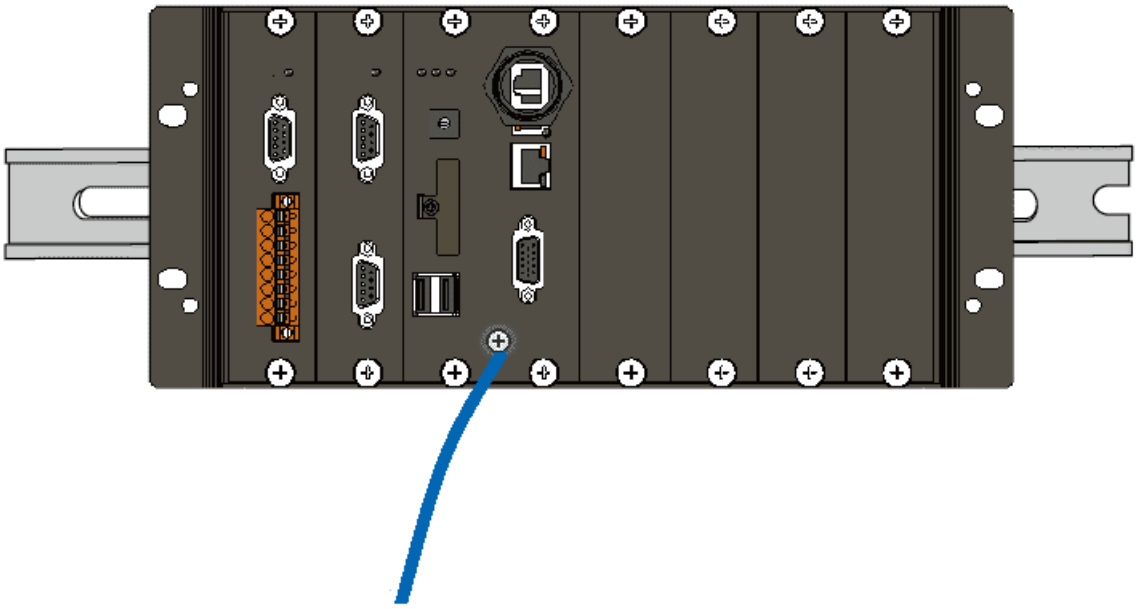


Tips & Warnings

For DIN-Rail mounting, it is strongly recommended that only a stainless steel DIN-Rail be used to support the weight of the LP-9x2x system, providing stability and preventing LP-9x2x from leaning.



Step 3: Connect the ground lead to the frame ground point.



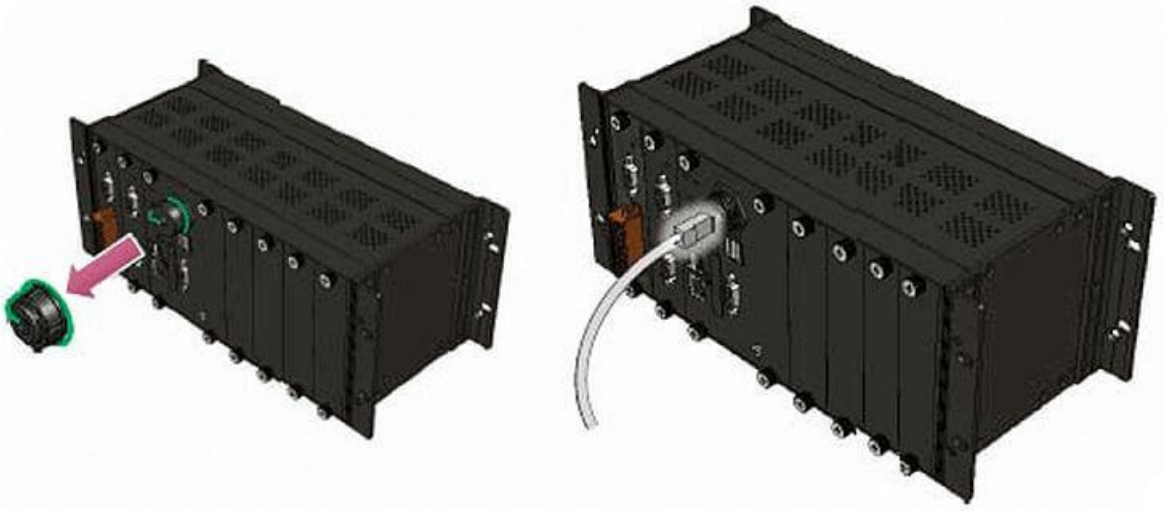
Tips & Warnings

A good common ground reference (earth ground) is essential for proper operation of the LP-9x2x. One side of all control circuits, power circuits, and the ground lead must be properly connected to earth ground by either installing a ground rod in close proximity to the enclosure or by connecting to the incoming power system ground. There must be a single-point ground (i.e. copper bus bar) for all devices in the enclosure that require an earth ground.

2.1.4.1. Installing the RJ-45 waterproof connector assembly

The LP-9x2x series is equipped with an RJ-45 waterproof connector to withstand contaminant in the dusty environment.

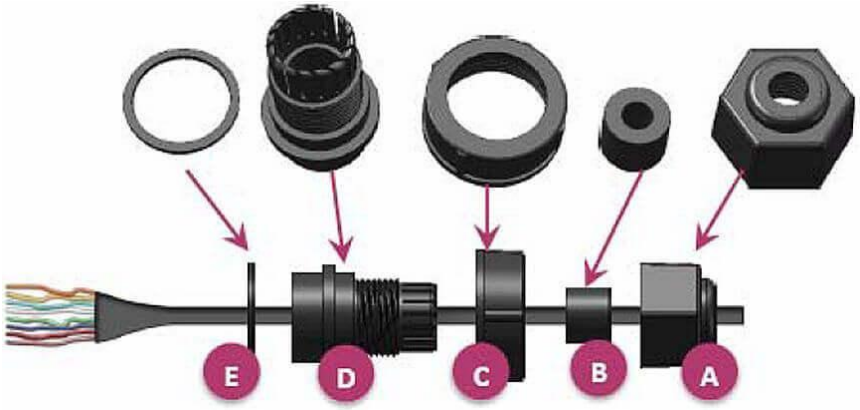
The RJ-45 waterproof connector is optional for use with LAN1 port. If you do not need the RJ-45 waterproof connector, you can remove the cap and just plug in a regular Ethernet cable.



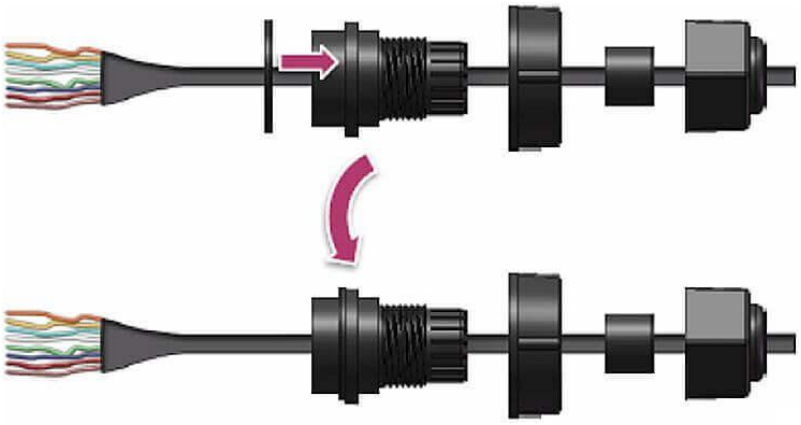
If you want to use the RJ-45 waterproof connector for protecting the connection, follow the instructions below.

Step 1: Remove the RJ-45 connector from the RJ-45 cable.

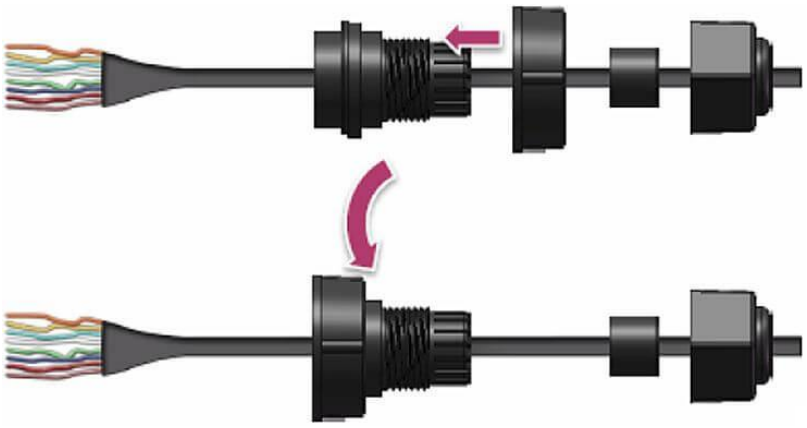
Step 2: Feed the end of the RJ-45 cable through the (A) sealing nut, (B) rubber sealing insert, (C) clamping ring, (D) cable gland base and (E) panel gasket.



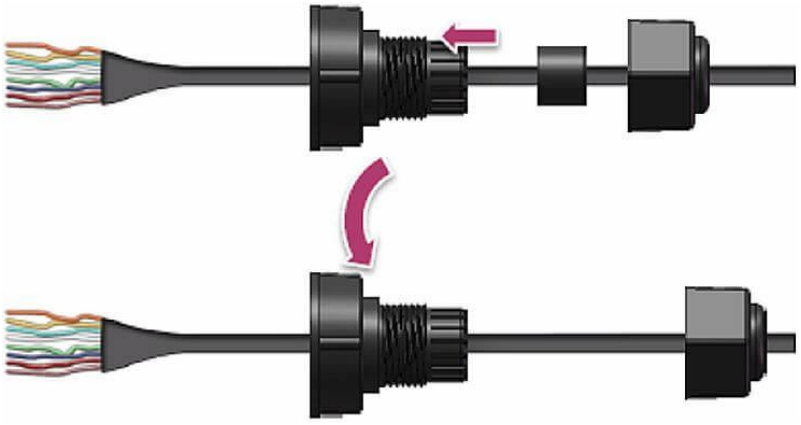
Step 3: Wrap the (E) panel gasket base around the (D) cable gland base.



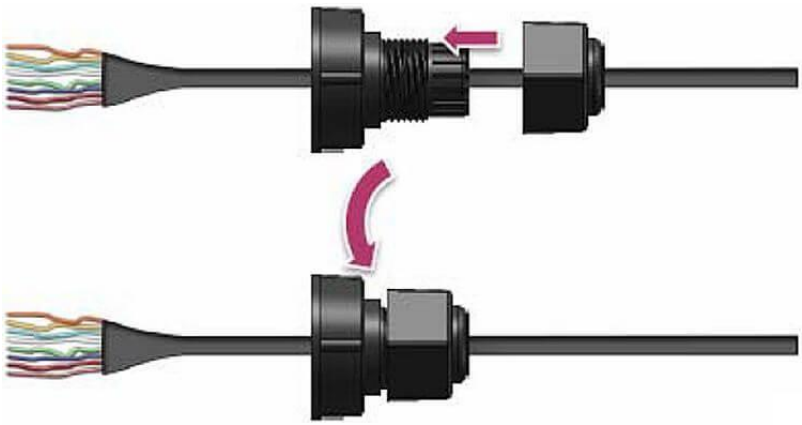
Step 4: Wrap the (C) clamping ring around the (D) cable gland base.



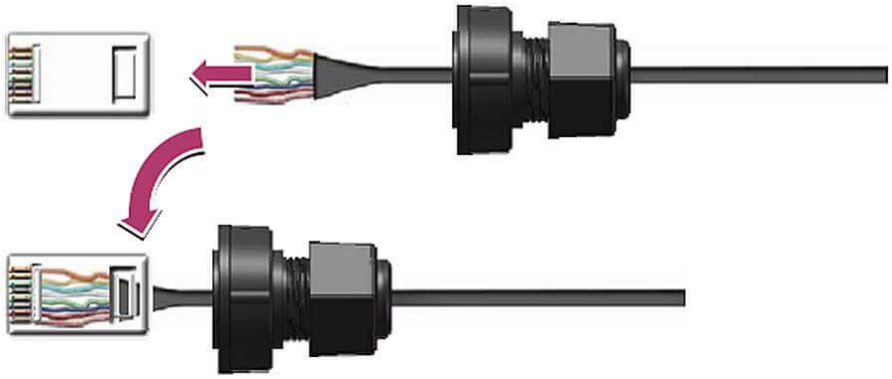
Step 5: Insert the (B) rubber sealing insert into the (D) cable gland base.



Step 6: Push the (A) sealing nut forward and Hand-tighten it to seal the assembly.

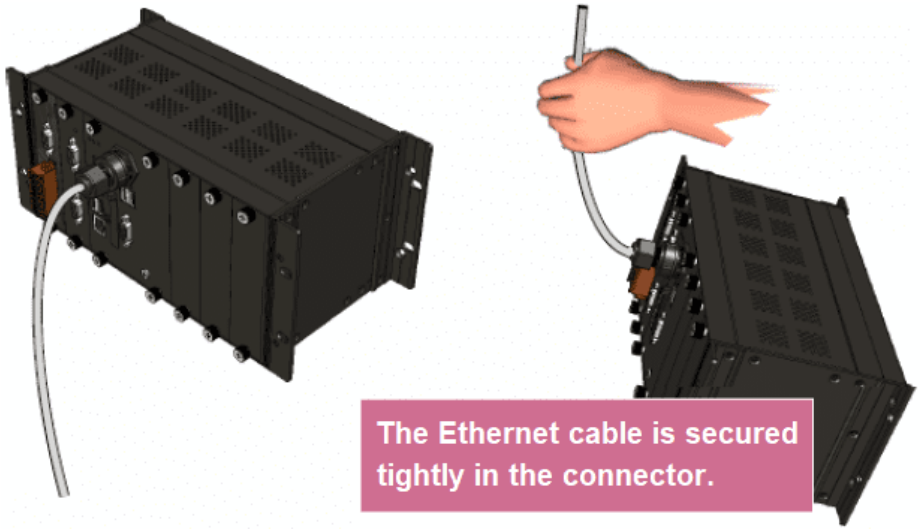


Step 7: Insert the RJ-45 cable into the RJ-45 connector.



Step 8: Push the RJ-45 waterproof connector assembly forward.

Step 9: Insert the Ethernet cable and screw the RJ-45 waterproof into the receptacle.



 E-mail: service@icpdas.com

2.2. Deploying a Basic System

The LinPAC AM335x series PAC provides a variety of communication interface to suit a range of application. Here is a simple application for using the LP-22xx/52xx/8x2x/9x2x.

Tips & Warnings

1. The input range of power supply is +10 V_{DC} to +30 V_{DC} for LP-8x2x/9x2x and +12 V_{DC} to +48 V_{DC} for LP-22xx/52xx.
2. The LP-8x2x/9x2x have two power inputs that can be connected simultaneously to the two independent power sources. If one power source fails, the other source takes over automatically. Redundant power input help assure non-stop operation of the LP-8x2x/9x2x.

2.2.1. Installation for LP-22xx

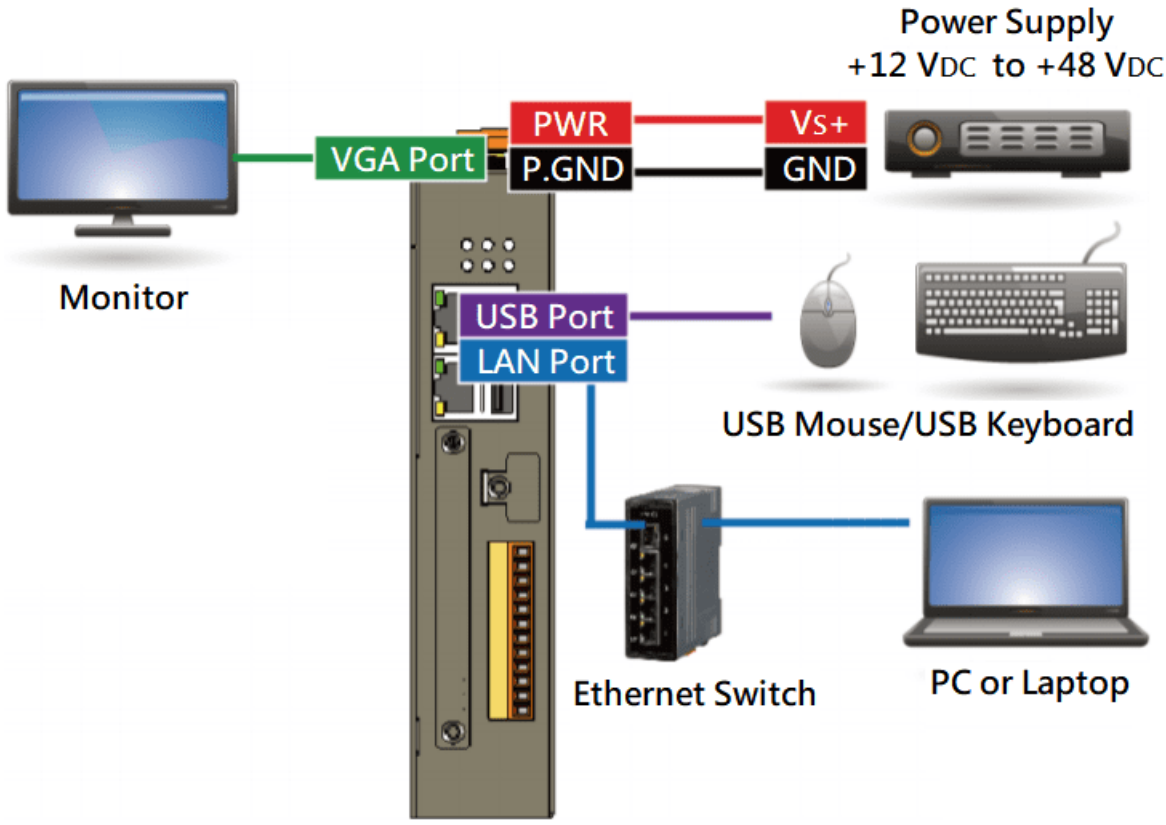
Connecting to a PC, the USB device, and the power supply.

Step 1: Connect the positive terminal (+) of the power supply to the terminal PWR and the negative terminal (-) of the power supply to the P.GND.

Step 2: Connect the USB mouse or the USB keyboard to the USB port.

Step 3: Connect the monitor to the VGA port.

Step 4: Connect to PC or the laptop to the LAN port via an Ethernet switch.



 E-mail: service@icpdas.com

2.2.2. Installation for LP-52xx

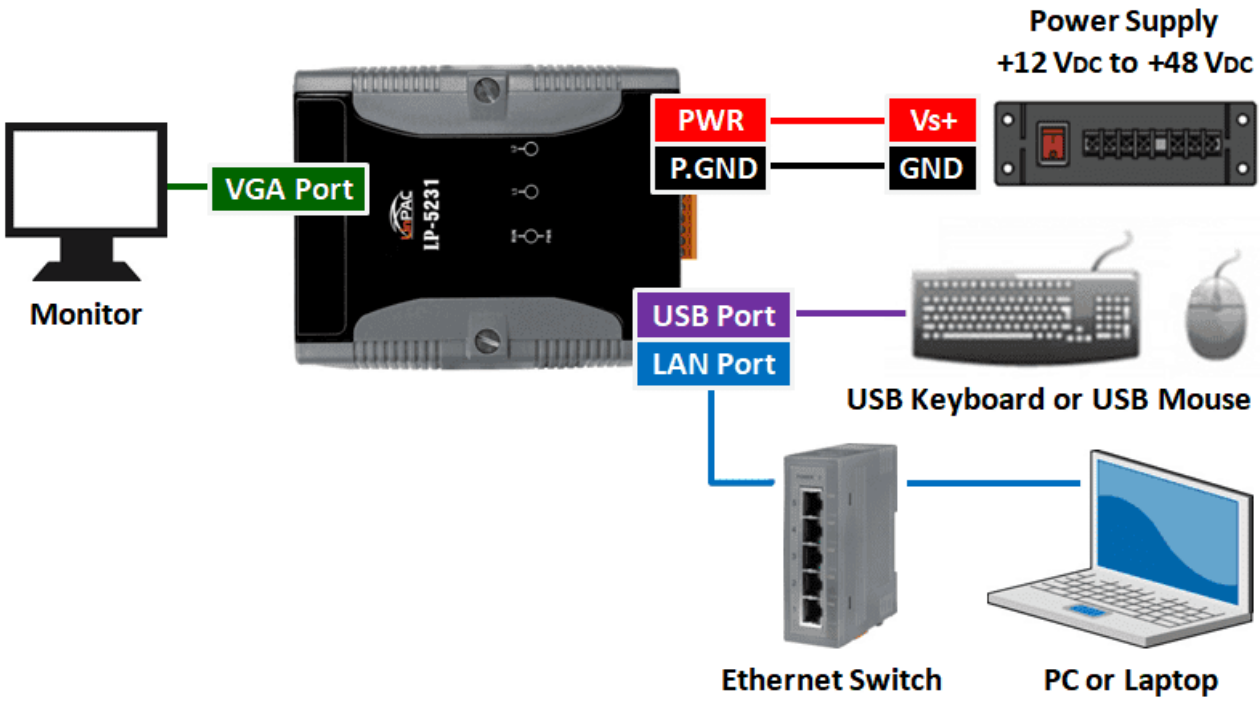
Connecting to a PC, the USB device, and the power supply.

Step 1: Connect the positive terminal (+) of the power supply to the terminal PWR and the negative terminal (-) of the power supply to the P.GND.

Step 2: Connect the USB mouse or the USB keyboard to the USB port.

Step 3: Connect the monitor to the VGA port.

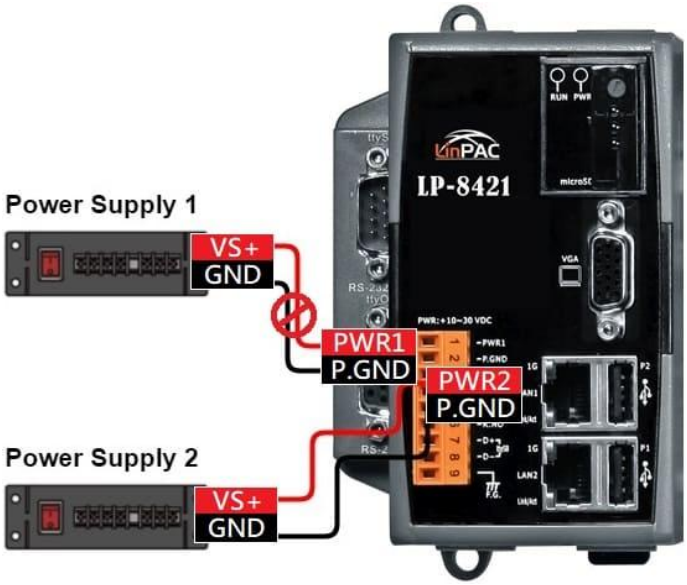
Step 4: Connect to PC or the laptop to the LAN port via an Ethernet switch.



2.2.3. Installation for LP-8x2x

Connecting to a PC, the USB device, and the power supply.

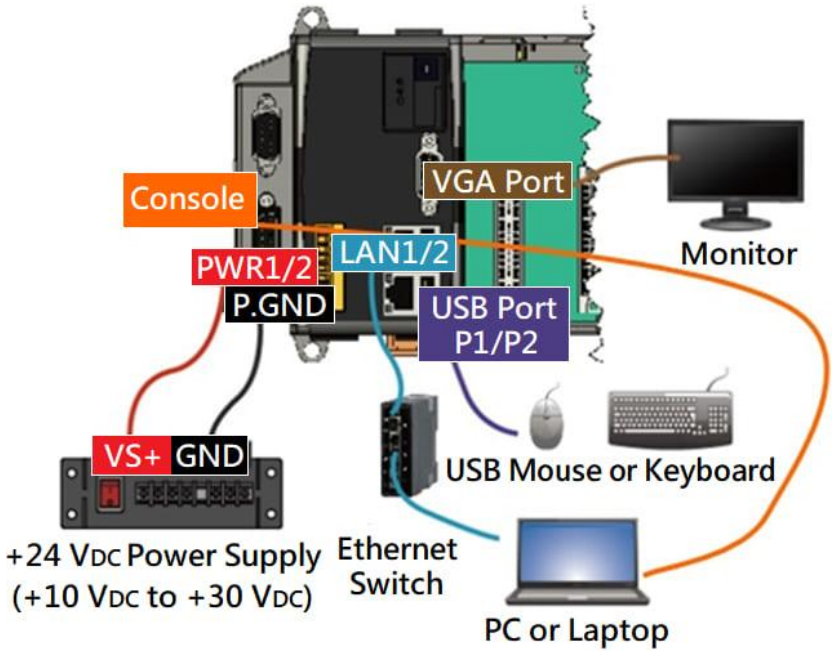
Step 1: Connect the positive terminal (+) of the power supply to the terminal PWR1/2 and the negative terminal (-) of the power supply to the P.GND.



Step 2: Connect the USB mouse or the USB keyboard to the USB port.

Step 3: Connect the monitor to the VGA port.

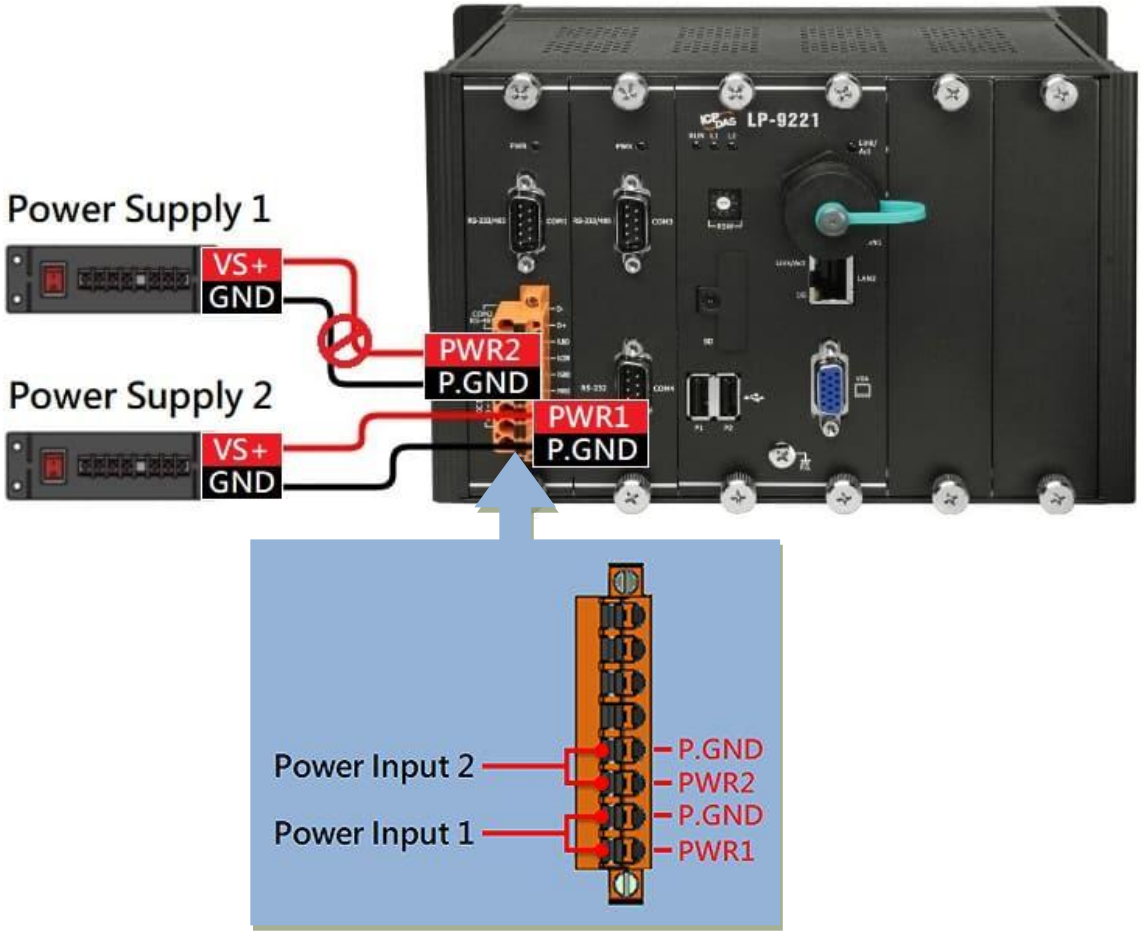
Step 4: Connect to PC or the laptop to the LAN port via an Ethernet switch.



2.2.4. Installation for LP-9x2x

Connecting to a PC, the USB device, and the power supply.

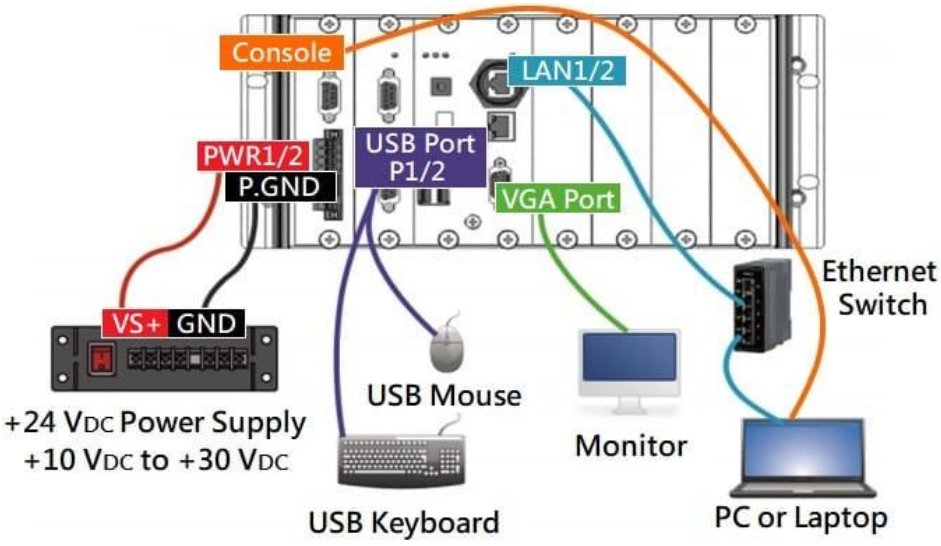
Step 1: Connect the positive terminal (+) of the power supply to the terminal PWR1/2 and the negative terminal (-) of the power supply to the P.GND.



Step 2: Connect the USB mouse or the USB keyboard to the USB port.

Step 3: Connect the monitor to the VGA port.

Step 4: Connect to PC or the laptop to the LAN port via an Ethernet switch.



2.3. Inserting the I/O Modules

LinPAC controller is equipped with rich I/O expansion ability, all kinds of I/O modules as described in the following:

Type	Number of RS-232 and 485 ports	Number of slots	I-7K	I-8K and I-87K	I-9K and I-97K	XV-board
LP-22xx	3	1	✓	-	-	✓
LP-52xx	3	1	✓	-	-	✓
LP-8x2x	2 or 3	1 or 4 or 8	✓	✓	-	-
LP-9x2x	4	2 or 4 or 8	✓	-	✓	-

Note: '✓': Support ; '-': Not Support

Before choosing the right I/O modules, you first need to know the I/O expansion capacities in order to choose the best expansion module for achieving maximal efficiency. For more information about the I/O expansion modules that are compatible with the LinPAC AM335x series PAC, please refer to the following website links:

LP-22xx/52xx series:

http://www.icpdas.com/root/product/solutions/hmi_touch_monitor/touchpad/xv-board_selection.html

LP-8x2x/9x2x series:

http://www.icpdas.com/root/product/solutions/remote_io/remote_io_products.php

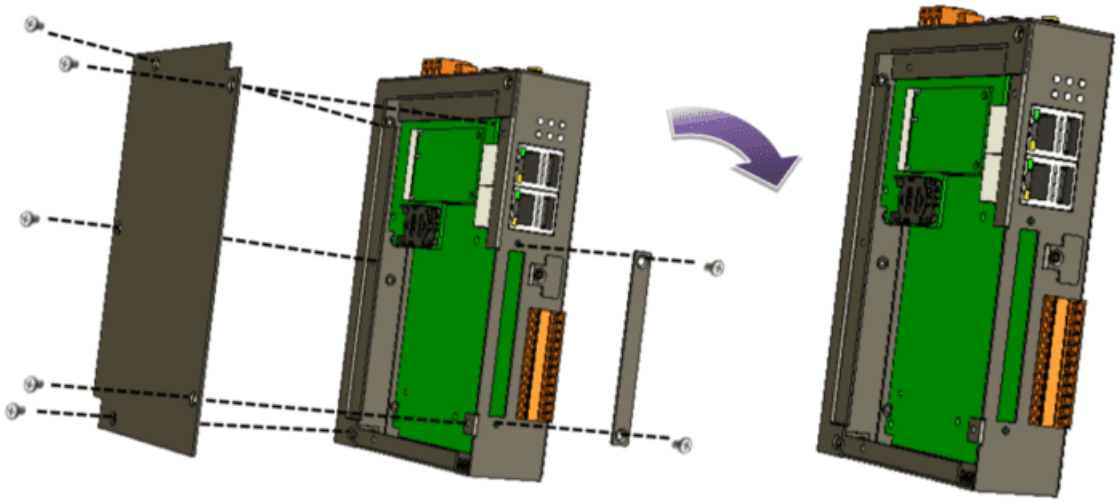
2.3.1. Adding an I/O Device for LP-22xx

2.3.1.1. Installing the XV-Board

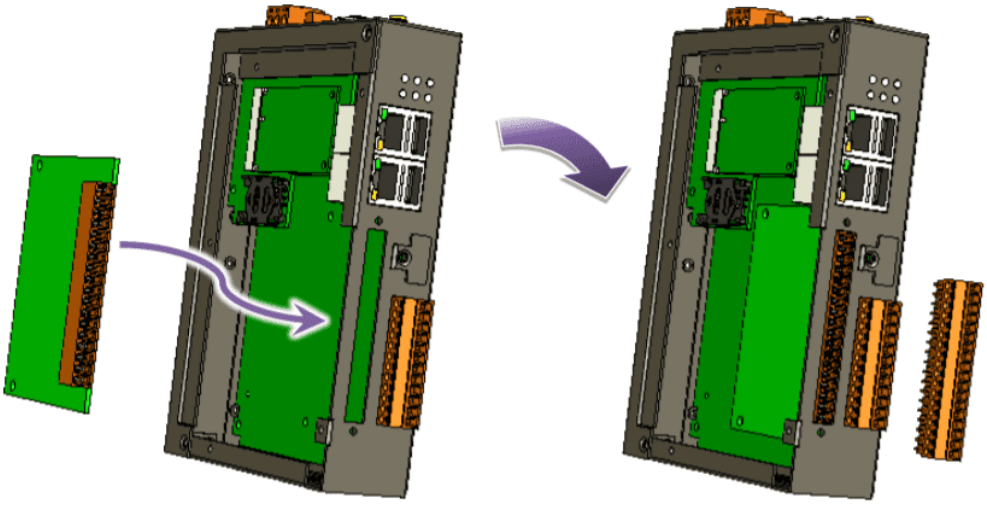
LP-22xx has one expansion I/O slots to expand the functions. For more information about the I/O expansion modules that are compatible with the LP-22xx, please refer to:

http://www.icpdas.com/root/product/solutions/hmi_touch_monitor/touchpad/xv-board_selection.html

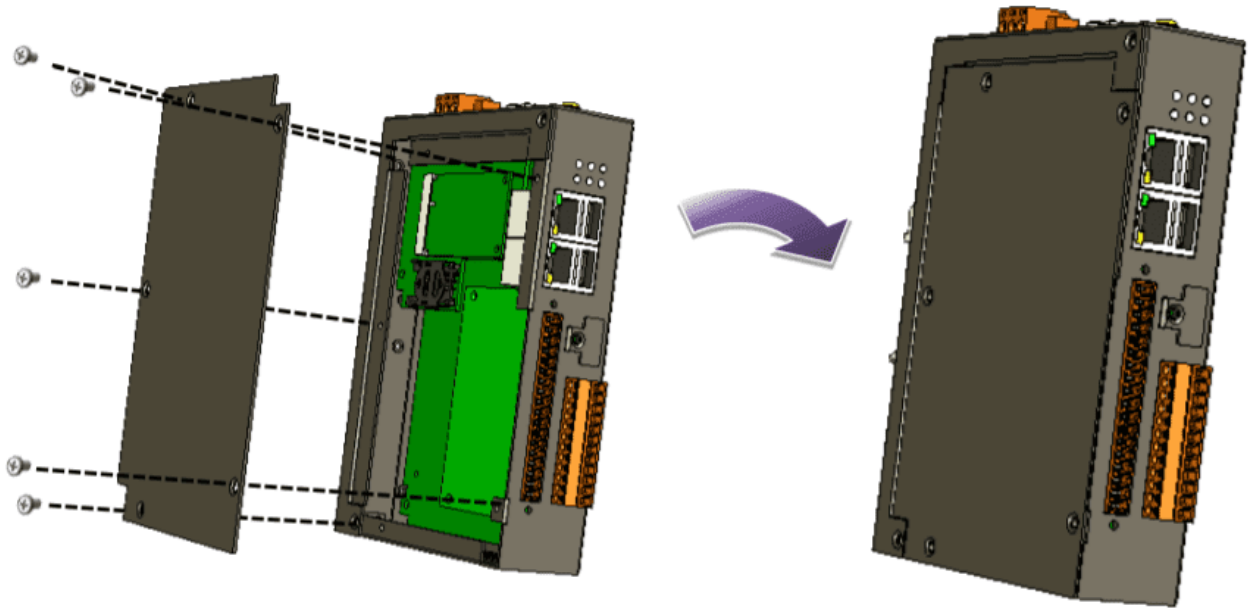
Step 1: Remove stripped screws and then remove the cover.



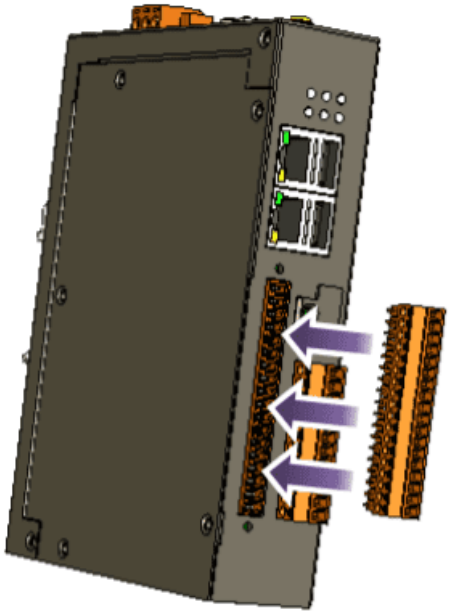
Step 2: Hold the XV-board vertically and align the socket, and then carefully press the XV-board onto the socket.



Step 3: Close the cover and then fasten the screws.



Step 4: Insert the I/O terminal and then stick the I/O sticker.



 E-mail: service@icpdas.com

2.3.2. Adding an I/O Device for LP-52xx

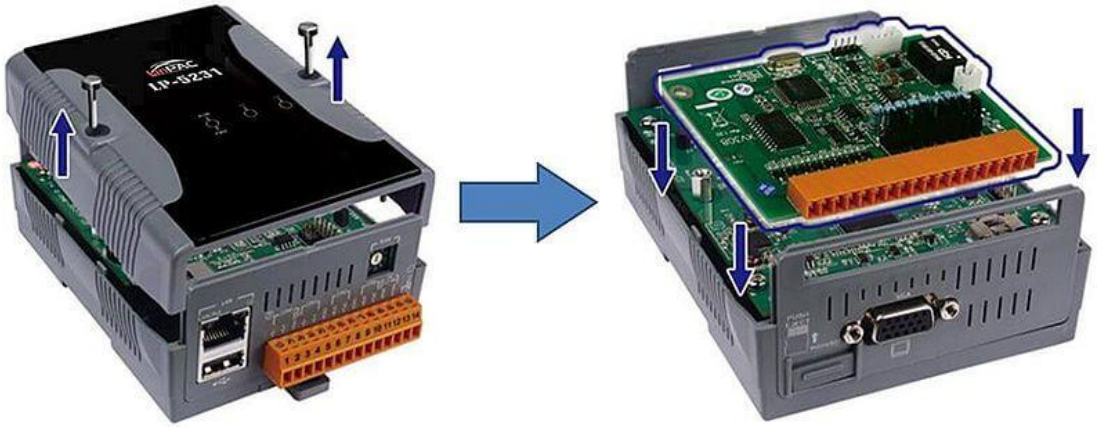
2.3.2.1. Installing the XV-Board

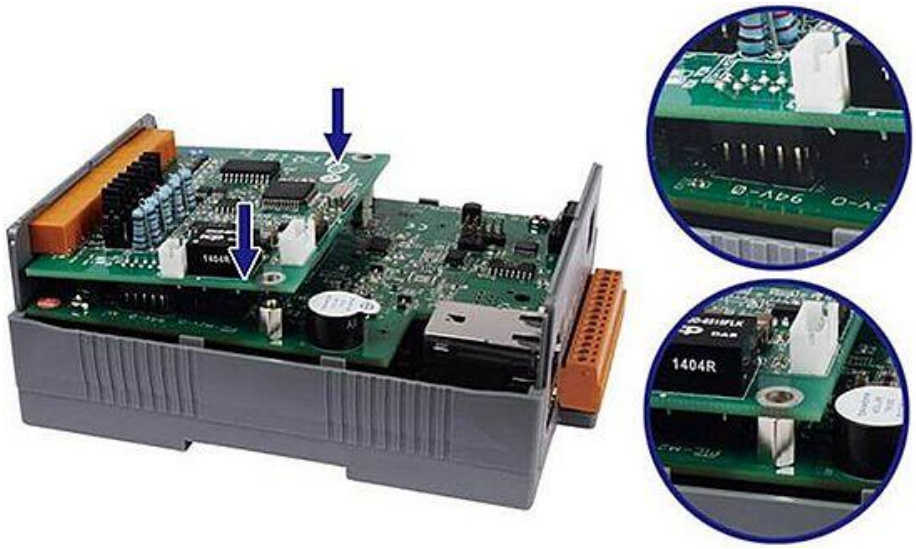
LP-52xx has one expansion I/O slots to expand the functions. For more information about the I/O expansion modules that are compatible with the LP-52xx, please refer to:

http://www.icpdas.com/root/product/solutions/hmi_touch_monitor/touchpad/xv-board_selection.html

Step 1: Remove stripped screws and then remove the cover.

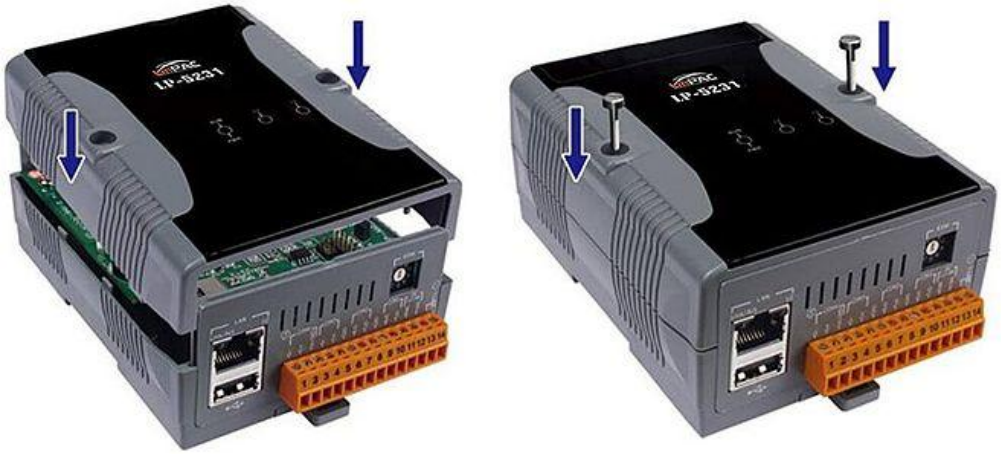
Step 2: Hold the XV-board vertically and align the socket, and then carefully press the XV-board onto the socket.





 E-mail: service@icpdas.com

Step 3: Close the cover and then fasten the screws.

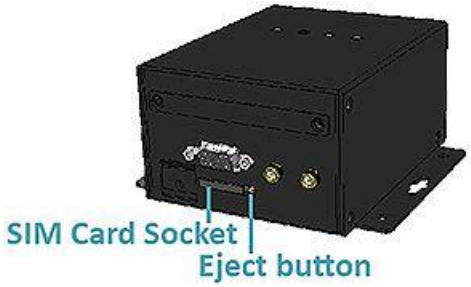


Step 4: Insert the I/O terminal and then stick the I/O sticker.

2.3.2.2. Inserting the SIM card

The SIM card tray is located on the top side of the module. The eject button is on the right side of the tray door.

Step 1: Push the ejection button until the SIM card tray pops out.



Step 2: Pull out the tray completely and set it on a flat surface.

Step 3: Put the SIM card in the tray, and then push the tray back into the socket.

Tips & Warnings

Make sure to turn the LP-5231PM-3GWA/LP-5231PM-4GE/LP-5231PM-4GC off before inserting or removing the SIM card. Do not bend or scratch the SIM card.

2.3.2.3. Installing the antenna

The LP-5231PM-3GWA/LP-5231PM-4GE/LP-5231PM-4GC has 2 antenna connectors that can be used to connect the 3G/4G antenna. To install the antenna, just screw the antenna tightly into the connector, and put the antenna in the purpose place.

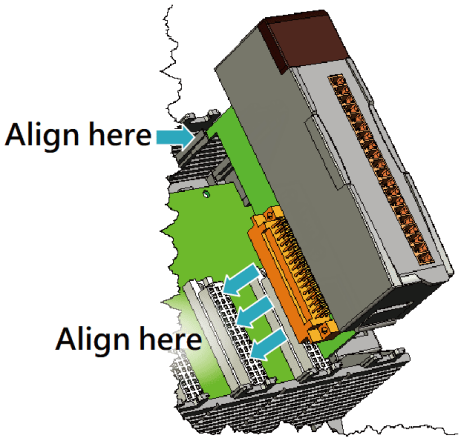
2.3.3. Adding an I/O Device for LP-8x2x

All I/O Web Page include the I/O module specifications, pin assignments and wiring connections.

For example, Pin Assignments and Wiring connections for the I-87054W module are as follows:

http://www.icpdas.com/root/product/solutions/remote_io/rs-485/i-8k_i-87k/i-87054w.html

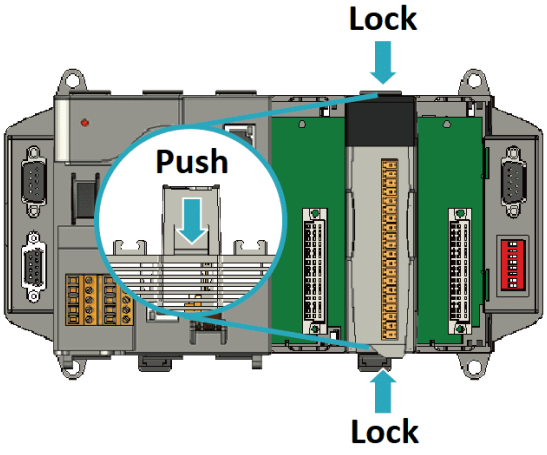
Step 1: Align circuit card with slot and press firmly to seat module into connector.



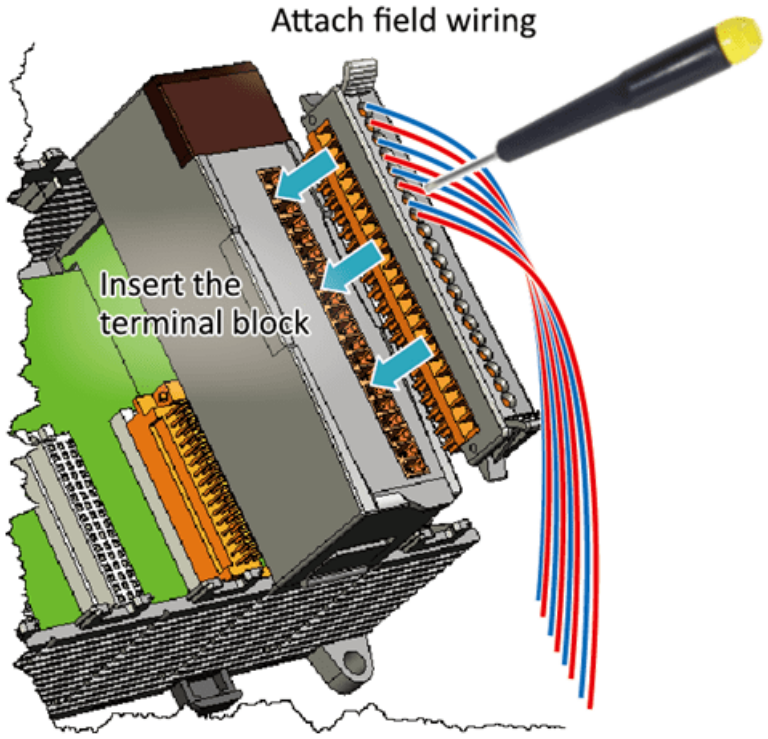
Tips & Warnings

It is recommended that the power to the LP-8x2x is switched off when wring the I/O module which are plugging in the LP-8x2x slots.

Step 2: Pull top and bottom locking tabs toward module face. Click indicates lock is engaged.



Step 3: Attach field wiring using the terminal block, and then insert the terminal block.



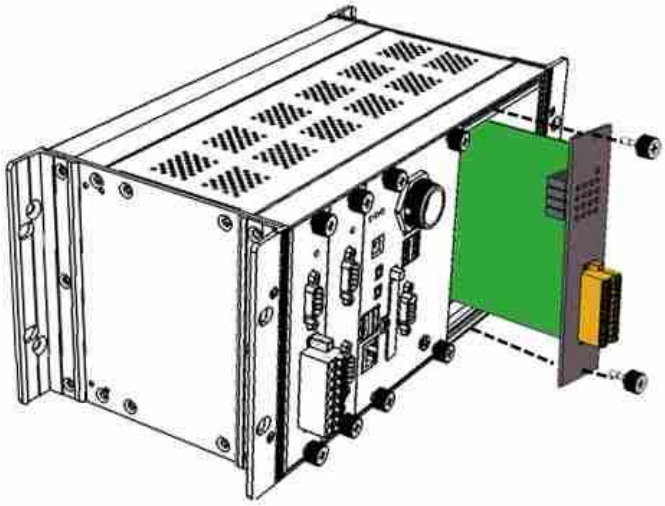
2.3.4. Adding an I/O Device for LP-9x2x

LP-9000 has 2/4/8 I/O expansion slots to support I-9K and I-97K series I/O modules. Before choosing the right I/O modules, you first need to know the I/O expansion capacities in order to choose the best expansion module for achieving maximal efficiency.

For more information about the I/O expansion modules that are compatible with the LP-9000, please refer to:

http://www.icpdas.com/root/product/solutions/remote_io/i-9k_i-97k/i-9k_i-97k_selection.html

Step 1: Insert the I/O module.

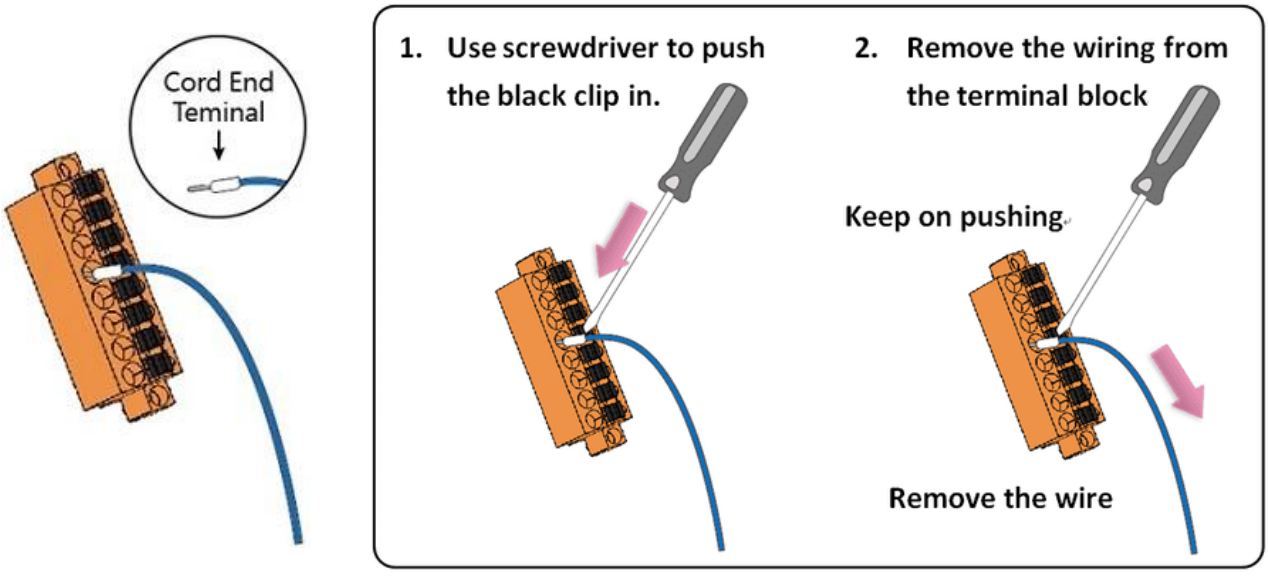


Tips & Warnings

If you do not expand the I/O module full, please keep the top case of the unused slot to protect the backplane from dirt, dust and damage from foreign objects.

Step 2: Wiring connection.

The metal part of the cord end terminal on the wire can be direct wired to the terminal of LP-9x2x.



2.4. Console Port Connection

The LinPAC AM335x PAC support remote connection from the ‘Console’ port. The user can follow below steps to connect to the LinPAC AM335x PAC.

Step 1: User can choose the software (Putty or others) through the ‘Console’ to connect the LinPAC AM335x PAC.

Step 2: If user chose the ‘Console’, user can set the baud rate ‘115200’ to connect the device. (Refer to the Figure 2.4-1)

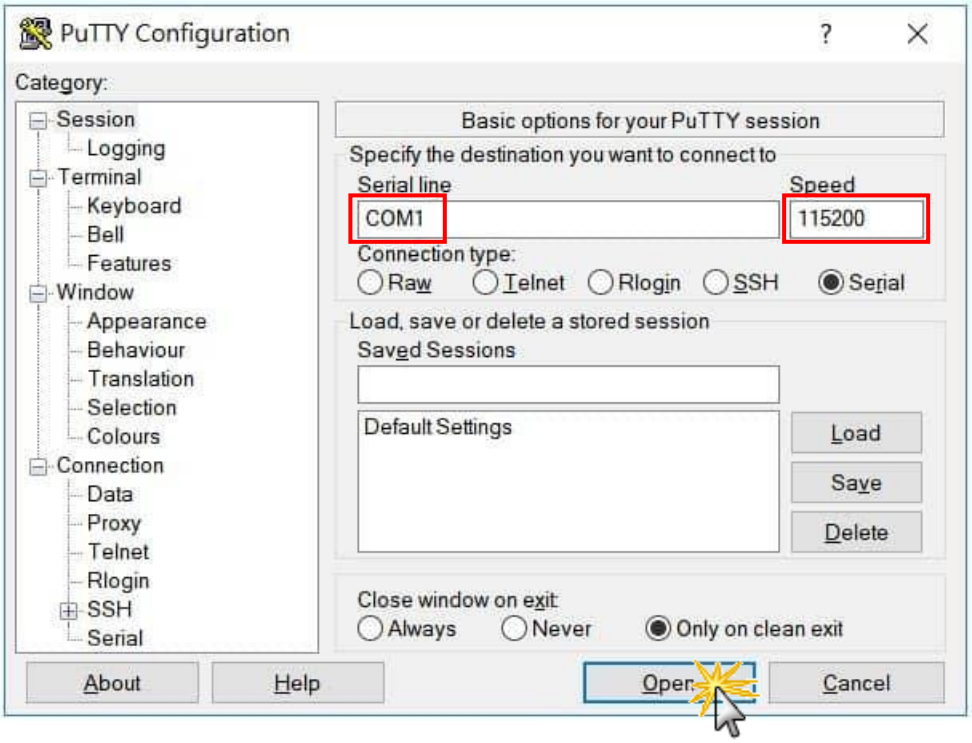


Figure 2.4-1. Console connection

Step 3: After user connect to the LinPAC AM335x PAC from the ’Console’ port, user can input default ID ’root’ and password ’icpdas’ to login.

2.5. LAN1/LAN2 Network Configuration

After logging into the LinPAC AM335x PAC successfully, the user can use '**ifconfig**' command to get the IP address of LAN1/LAN2. The LAN 1/2 of factory setting use DHCP. If user would prefer to setup the IP address for static mode, the following steps for reference:

Step 1: Using the Linux command 'vi' to modify the file '/etc/network/interfaces'.

Step 2: Using the '#' to mark the default configuration.

Remove the '**#**' comment from each line in the static IP block and comment out the DHCP block by adding '**#**' to each entry. Entry the relevant IP, Netmask and Gateway details in the respective Assign IP block entries. (Refer to the Figure 2.5-1)

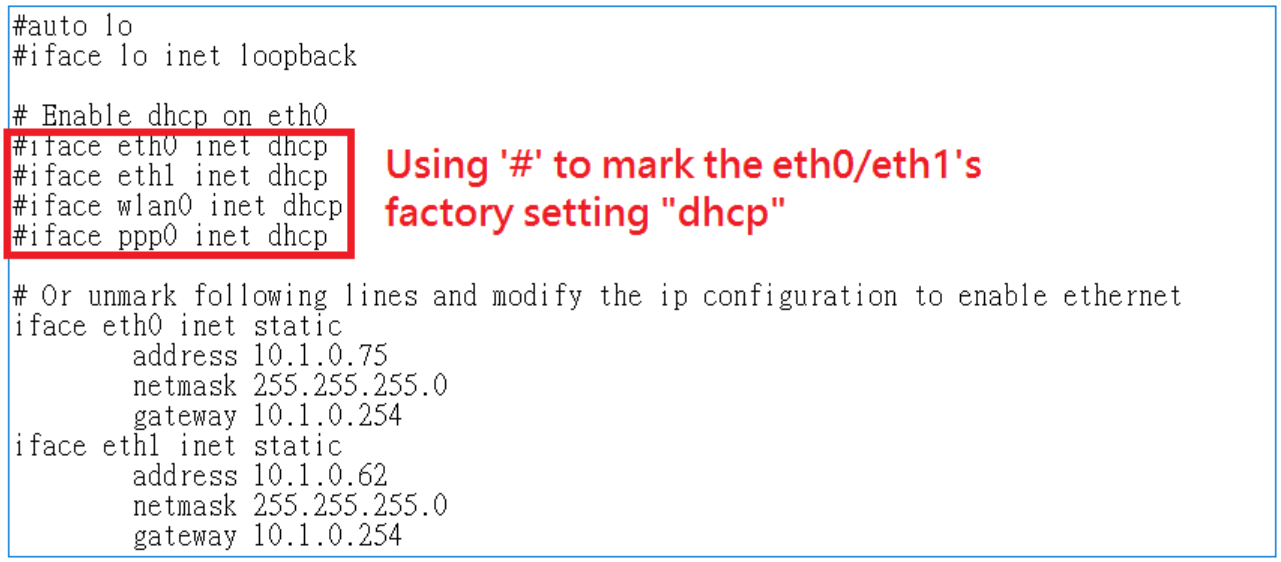


Figure 2.5-1. Network configuration

Step 3: After user save the file and use 'reboot' command to reset the device, user can use the new network configuration on LinPAC AM335x PAC.

2.6. LAN1/LAN2 Network Connection

The user can use '**ifconfig**' command to get the IP address of LAN1/LAN2 and the SSH client software (Putty or others) to connect the LinPAC AM335x PAC.

Step 1: Using 'ifconfig' command to check the IP address of LAN1/2. (Refer to the Figure 2.6-1)

```
root@icpdas:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0d:e0:6e:08:16
          inet addr:10.1.0.75  Bcast:10.1.255.255  Mask:255.255.0.0
          inet6 addr: fe80::20d:e0ff:fe6e:816/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:593929 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8928 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:39831066 (39.8 MB)  TX bytes:5713563 (5.7 MB)
          Interrupt:17 Memory:febe0000-fec00000
eth1      Link encap:Ethernet  HWaddr 00:0d:e0:b0:97:0b
          inet addr:10.1.0.62  Bcast:10.1.255.255  Mask:255.255.0.0
          inet6 addr: fe80::20d:e0ff:feb0:970b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:587861 errors:0 dropped:1 overruns:0 frame:0
          TX packets:36 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:28616065 (28.6 MB)  TX bytes:7795 (7.7 KB)
```

Figure 2.6-1. Typing 'ifconfig' command to check the IP address

Step 2: User can use SSH client software (Putty or others) to connect the LinPAC. (Refer to the Figure 2.6-2)

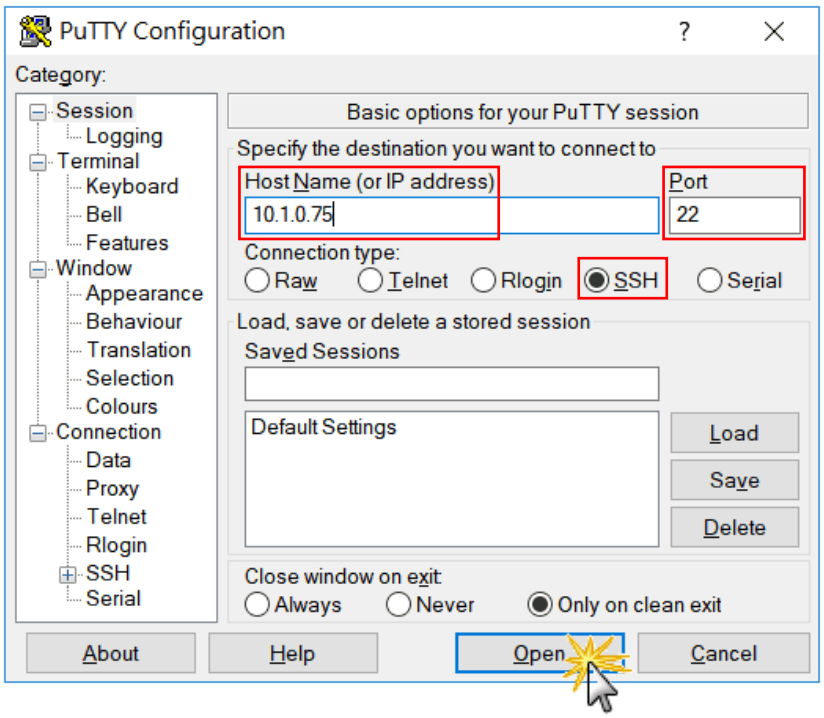


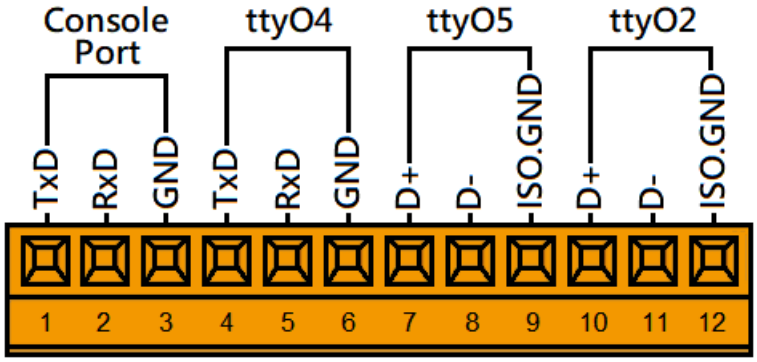
Figure 2.6-2. Using 'Putty' utility to connect the LinPAC

2.7. Overview of the Serial Ports

The following is a description of the functionality for the three serial ports contained in the LinPAC AM335x series embedded controller, and are based on the RS-232 or RS-485 interfaces.

2.7.1. Introduction to Serial port for LP-22xx

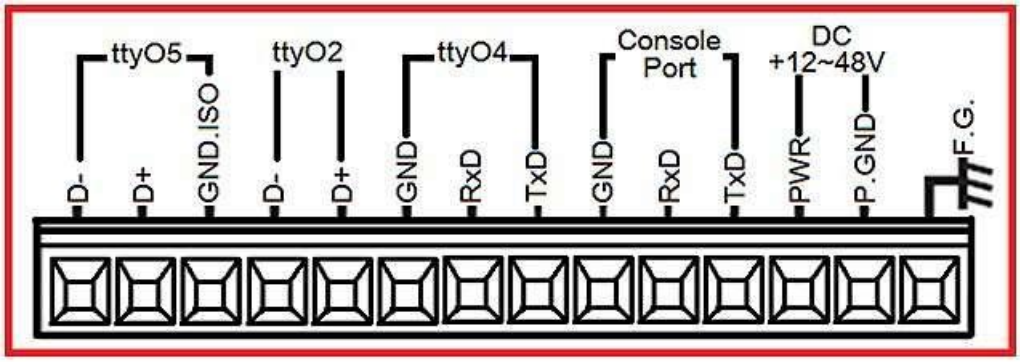
The following illustrates the ports contained on the LP-22xx. The information is organized as follows table:



Device name	Definition in LP-22xx SDK	Description	Default Baud rate
-	/dev/ttyO1 or COM1	Internal communication with the XV-board modules	115200
-	Console port	RS-232 (RxD, TxD and GND); Non-isolated	115200
ttyO4	/dev/ttyO4 or COM4	RS-232 (RxD, TxD and GND); Non-isolated	9600
ttyO2	/dev/ttyO2 or COM2	RS-485 (Data+, Data-); Non-isolated	9600
ttyO5	/dev/ttyO5 or COM5	RS-485 (Data+, Data-); 2500 VDC isolated	9600

2.7.2. Introduction to Serial port for LP-52xx

The following illustrates the ports contained on the LP-52xx. The information is organized as follows table:



Device name	Definition in LP-52xx SDK	Description	Default Baud rate
-	/dev/ttyO1 or COM1	Internal communication with the XV-board modules	115200
-	Console port	RS-232 (RxD, TxD and GND); Non-isolated	115200
ttyO4	/dev/ttyO4 or COM4	RS-232 (RxD, TxD and GND); Non-isolated	9600
ttyO2	/dev/ttyO2 or COM2	RS-485 (Data+, Data-); Non-isolated	9600
ttyO5	/dev/ttyO5 or COM5	RS-485 (Data+, Data-); 2500 VDC isolated	9600

2.7.3. Introduction to Serial port for LP-8x2x

Figure 2.7.3-1 illustrates the ports contained on the LP-8821 and Figure 2.7.3-2 illustrates those on the LP-8121. The information is organized as follows:

- ttyO4– Internal communication with the I-87KW modules in slots
- ttyO5 – RS-232 (RxD, TxD and GND); Non-isolation; Console
- ttyS0 – RS-485 (D2+, D2-; self-tuner ASIC inside)
- ttyS1 – RS-232/RS-485
  (RXD, TXD, CTS, RTS and GND for RS-232, Data+ and Data- for RS-485)
- ttyS34 – RS-232 (RXD, TXD, CTS, RTS, DSR, DTR, CD, RI and GND)

Device name	Definition in LP-8x21 SDK	Default baudrate
ttyO4	COM1	115200
ttyO5 (RS-232/console)	None	115200
ttyS0 (RS-485)	COM2	9600
ttyS1 (RS-232/485)	COM3 (LP-8421/8821 only)	9600
ttyS34 (RS-232)	COM36 (LP-8421/8821 only)	9600

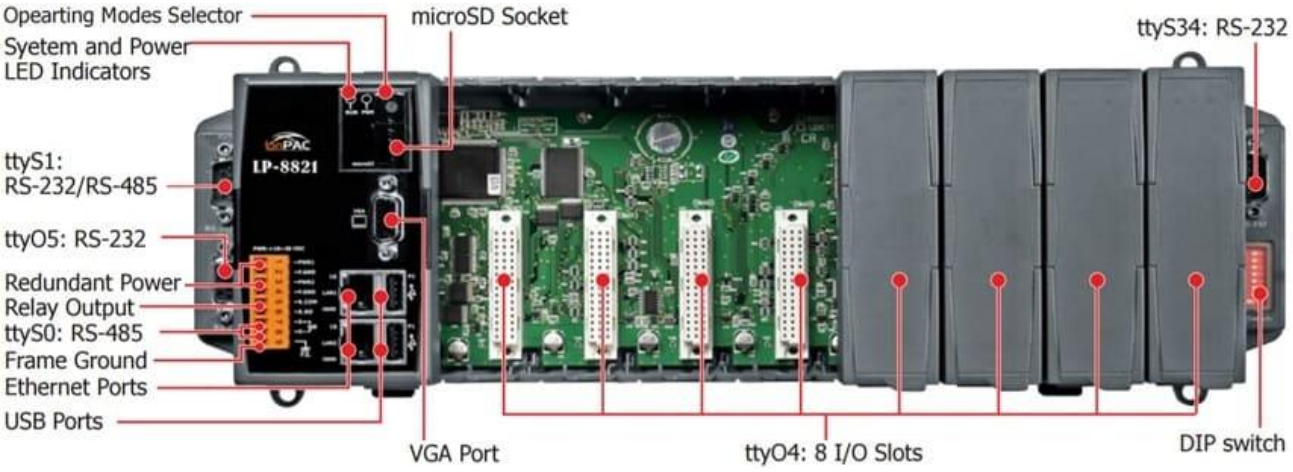


Figure 2.7.3-1. Serial port mapping on the LP-8821

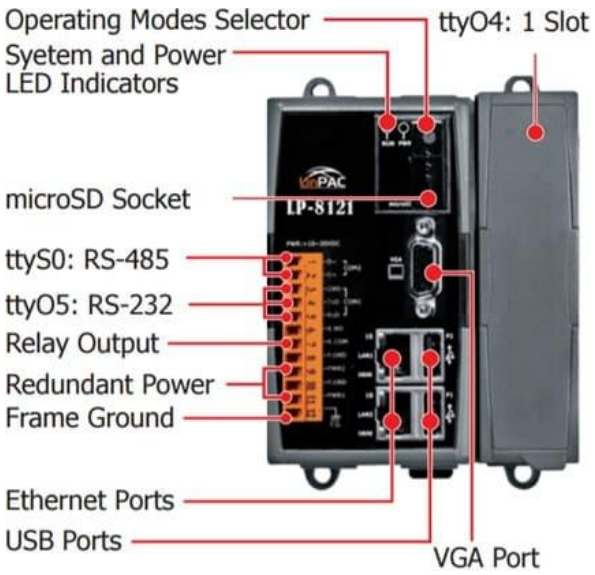


Figure 2.7.3-2. Serial port mapping on the LP-8121

2.7.4. Introduction to Serial port for LP-9x2x

Figure 2.7.4-1 illustrates the ports contained on the LP-9821. The information is organized as follows:

- ttyO4– Internal communication with the I-97KW modules in slots
- ttyO5 – RS-232/RS-485; Non-isolation; Console
- ttyS0 – RS-485 (D2+, D2-; self-tuner ASIC inside)
- ttyS1 – RS-232/RS-485
  (RXD, TXD, CTS, RTS and GND for RS-232, Data+ and Data- for RS-485)
- ttyS34 – RS-232 (RXD, TXD, CTS, RTS, DSR, DTR, CD, RI and GND)

Device name	Definition in LP-9x21 SDK	Default baudrate
ttyO4	COM1	115200
ttyO5 (RS-232/485/console)	None	115200
ttyS0 (RS-485)	COM2	9600
ttyS1 (RS-232/485)	COM3 (LP-9421/9821 only)	9600
ttyS34 (RS-232)	COM36 (LP-9421/9821 only)	9600

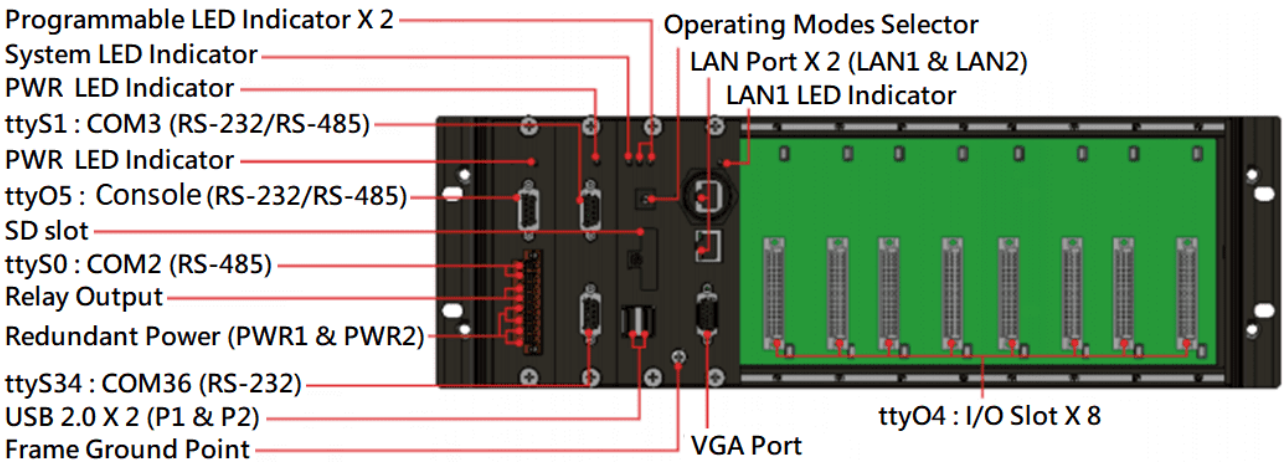


Figure 2.7.4-1. Serial port mapping on the LP-9821

2.7.5. Accessing the common serial ports

2.7.5.1. Internal communication for COM1 port

The COM1 port is an internal I/O expansion port on the LinPAC and is used to connect to the series module inserted into the LinPAC embedded controller. The I-87K, I-97K and XV-board series is based on a serial interface, which is provided for combining a variety of I/O function within the LP-8x2x, LP-9x2x and LP-22xx/52xx controllers. The differences between the three series are listed as follows:

- ❒ I-87K series can used with LP-8x2x, visit to this website:
  http://www.icpdas.com/root/product/solutions/remote_io/rs-485/i-8k_i-87k/i-8k_i-87k_dio.html
- ❒ I-97K series can used with LP-9x2x, visit to this website:
  http://www.icpdas.com/root/product/solutions/remote_io/i-9k_i-97k/i-9k_i-97k_aio.html
- ❒ XV-board series can used with LP-22xx/52xx, visit to this website:
  http://www.icpdas.com/root/product/solutions/hmi_touch_monitor/touchpad/xv-board_selection.html

A serial command must be used to control the I-87KW/97K series module. For more information about serial command usage, see the chapter 3.2. i-Talk Utility.

To control the series module, the Com port parameters and call the **Open_Com()** function to open the COM1 port based on the appropriate settings. Finally, call the **ChangeToSlot(slot)** function to specify which slot will be controlled, the I-87KW/97K series could be necessary. This is like the serial address, meaning that control commands can be sent to an I/O module that is inserted in the specified slot.

For Example:

```
int slot=1; char data=8, parity=0, stopbit=1 ;
unsigned char port=1;                    // for all modules in COM1 port of LP-8x21
DWORD baudrate=115200;
Open_Slot(slot);
Open_Com(port, baudrate, data, parity, stopbit);
ChangeSlotToI-87k(slot);
// send command...
Close_Com(port);
Close_Slot(slot);
```

2.7.5.2. RS-232 port

The following is RS-232 serial port for the LP-22xx, LP-52xx, LP-8x2x and LP-9x2x, as illustrated in Figures 2.7.5.2-1, 2.7.5.2-2 and 2.7.5.2-3 below.

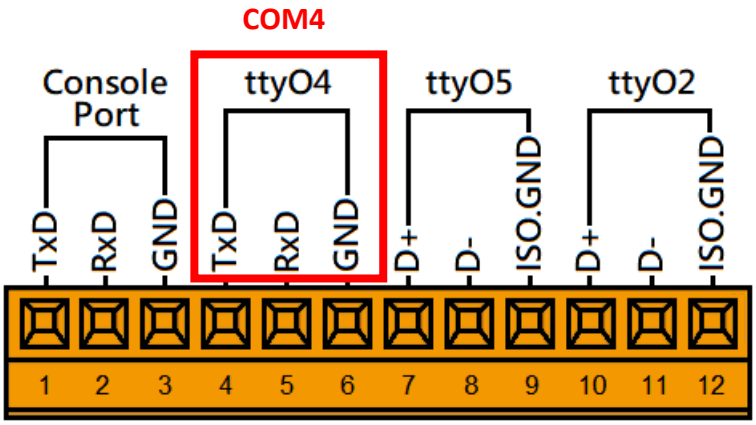


Figure 2.7.5.2-1. COM4 serial port for the LP-22xx

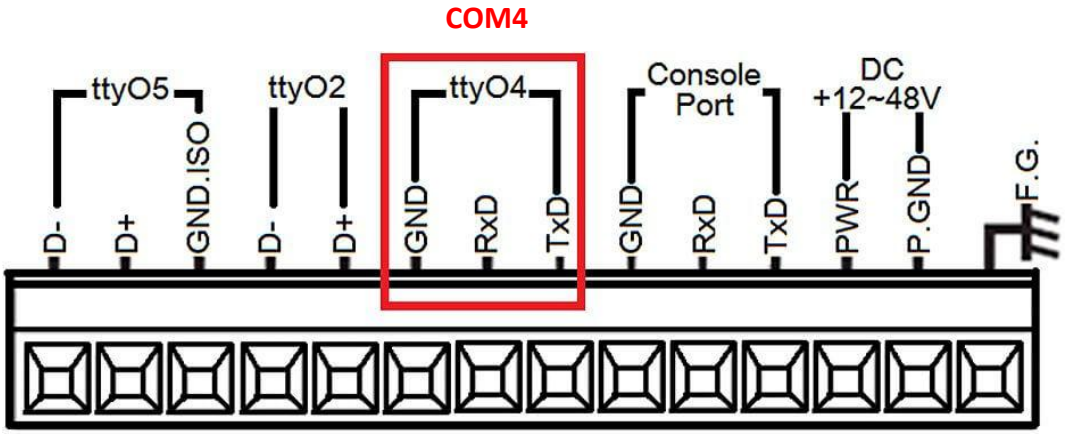


Figure 2.7.5.2-2. COM4 serial port for the LP-52xx

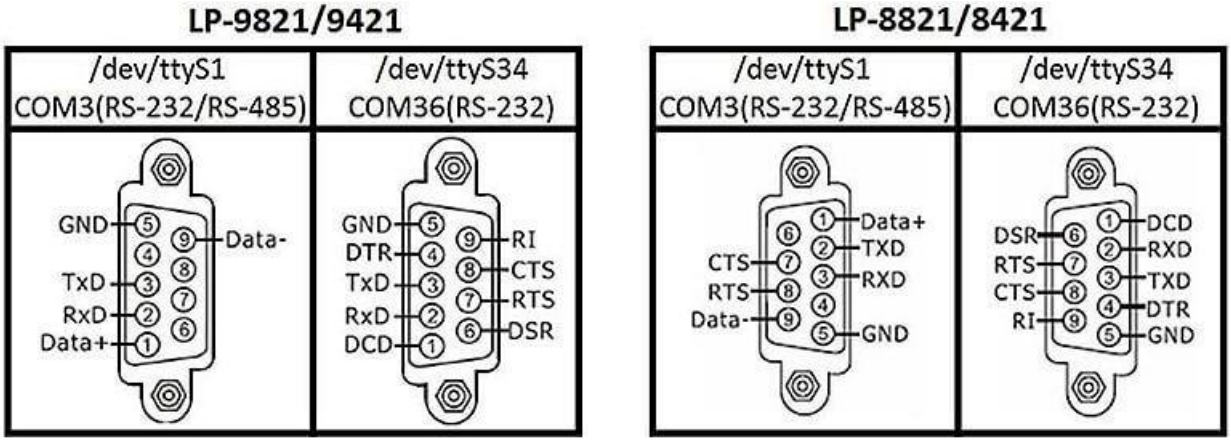


Figure 2.7.5.2-3. COM3 and COM36 serial port for the LP-8x2x/9x2x

This /dev/ttyS1, /dev/ttyS34 or /dev/ttyO4 port is located on the right-upper corner on the LP-22xx, LP-52xx, LP-8x2x and LP-9x2x and is a standard **RS-232** serial port that provides TxD, RxD, RTS, CTS, GND, non-isolated and a maximum speed of 115200 bps.

The RS-232 port can also be used to connect to an I-7520 module in order to provide general RS-485 communication functionality, and also can be used to connect to a wireless modem so that the module controlled from a remote device. The application example and code is demonstrated below:

- Test using C language:

```
unsigned char port=3;
DWORD baudrate=9600;
char data=8;
char parity=0;
char stopbit=1;
Open_Com(port, baudrate, data, parity, stopbit);
// Send a command...
Close_Com(port);
```

- Test using the command line interface: (PC connected to /dev/ttyS1 on the LP-8x21- see Figure 2.7.5.2-4)

(A) Open '**Hyper Terminal**' on the Host PC to monitor the test process. The default settings for COM3 port are 9600, 8, N, 1

(B) Send data via /dev/ttyS1 port:

On the LP-8x21:

Type the command: **echo send-232 >/dev/ttyS1**

Check that the word 'send-232' is displayed on the 'Hyper Terminal' screen on the PC

(C) Receive data via the /dev/ttyS1 port:

On the LP-8x21:

Type the command: **cat /dev/ttyS1**

On the PC:

Enter some text in the 'Hyper Terminal' screen on the PC

Check that the some words on the LP-8x21.

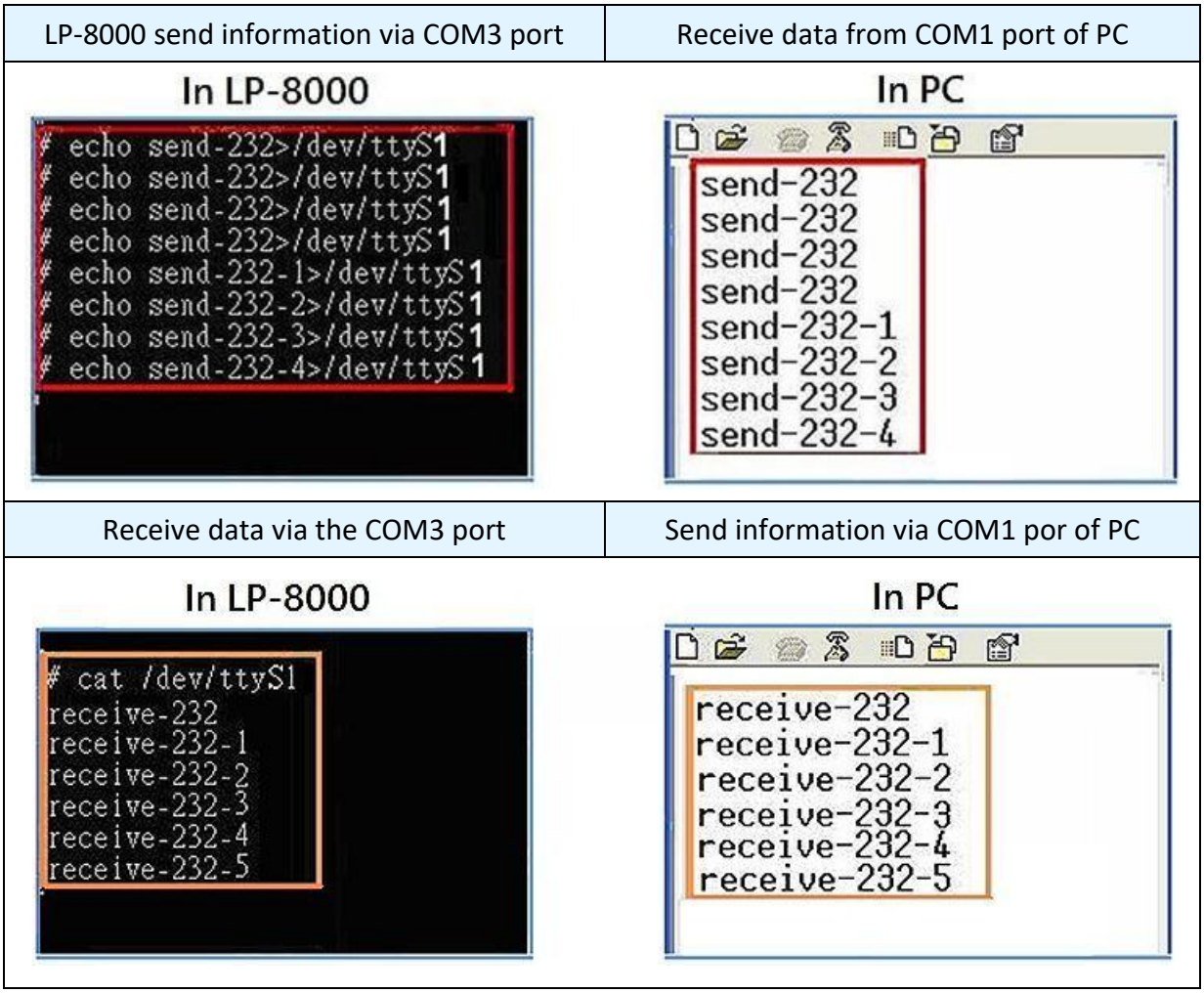


Figure 2.7.5.2-4. Using the command line to test

2.7.5.3. RS-485 port

The following is 2-wire RS-485 serial port for the LP-22xx, LP-52xx, LP-8x2x and LP-9x2x, as illustrated in Figures 2.7.5.3-1, 2.7.5.3-2 and 2.7.5.3-4 below.

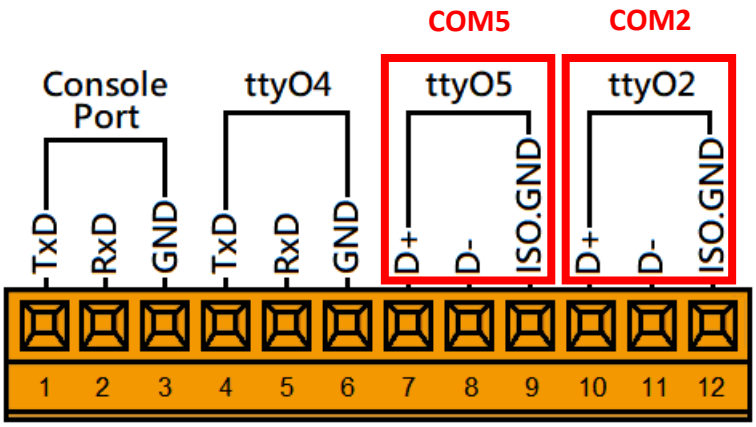


Figure 2.7.5.3-1. RS-485 connections of COM2 and COM5 for LP-22xx

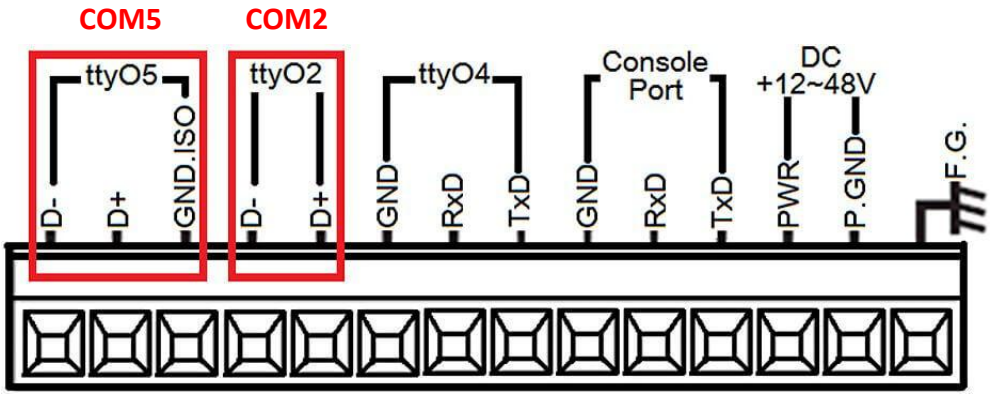


Figure 2.7.5.3-2. RS-485 connections of COM2 and COM5 for LP-52xx

Use the ‘setexdo’ command to set digital output value to a serial module. (Refer to Figure 2.7.5.3-3)

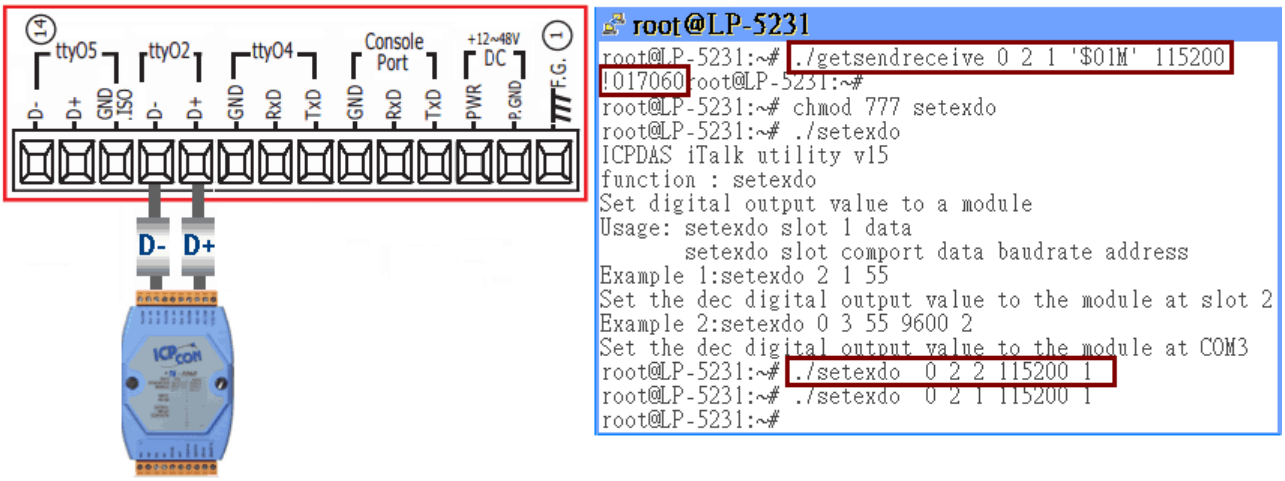


Figure 2.7.5.3-3. Using the command line to test on LP-5231

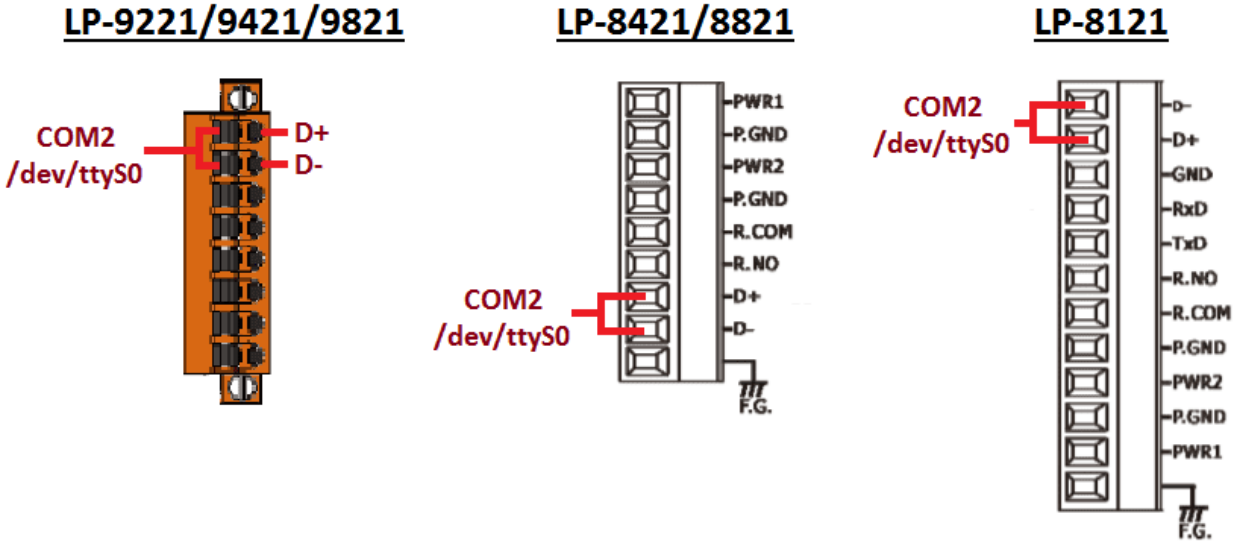


Figure 2.7.5.3-4. RS-485 connections of COM2 for LP-8x2x/9x2x

This port provides **RS-485** serial communication functionality (DATA+ and DATA-) and is located on the bottom-right corner on the LP-22xx, LP-52xx, LP-8x2x and LP-9x2x. This port allows a connection to be made to modules that contain an RS-485 interface such as the I-7000 serial modules (DCON Module), meaning that ICP DAS I/O series modules can be directly controlled via this port with any converter. ICP DAS provides a very easy to use library of functions (libi8k.a) that can use to easily communicate with I-7000, I-8000, I-9000, I-87k and I-97k series modules. Below is an application example of the program code demo.

- Test using C language:

  unsigned char port=36;

  DWORD baudrate=9600;

  char data=8, parity=0, stopbit=1;

  Open_Com(port, baudrate, data, char parity, stopbit);

  // send command…

- Test using command line: (PC <--> i-7520 <--> /dev/ttyS on the LP-8x21 - see Figure 2.7.5.3-5)

  (A) Open '**Hyper Terminal**' on the Host PC to monitor the test process. The default settings for the /dev/ttyS0 port are 9600, 8, N, 1

  (B) Send data via /dev/ttyS0 port:

  On the LP-8x21:

  Type command: **echo send-485 >/dev/ttyS0**

  Check that the word 'send-485' is displayed on the 'Hyper Terminal' screen on the PC.

  (C) Receive data via the /dev/ttyS0 port:

  On the LP-8x21:

  Type the command: **cat /dev/ttyS0**

  On the PC:

  Enter some words in the 'Hyper Terminal' screen on the PC

  Check that the same text displayed on the LP-8x21.

LP-8000 send information via COM2 port	Receive data from COM1 port of PC
In LP-8000 #echo send-485>/dev/ttyS0 #echo send-485-1>/dev/ttyS0 #echo send-485-2>/dev/ttyS0 #echo send-485-3>/dev/ttyS0 # #echo send-485-4>/dev/ttyS0 #echo send-485-5>/dev/ttyS0 # #	In PC send-485 send-485-1 send-485-2 send-485-3 send-485-4 send-485-5

Receive data via the COM2 port	Send information via COM1 por of PC

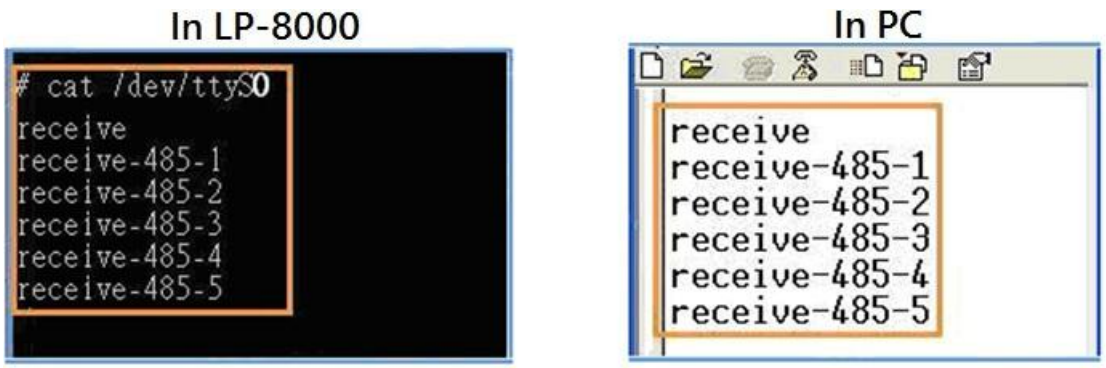


Figure 2.7.5.3-5. Using the command line to test

2.7.6. Serial Port configuration

Use the **'stty**' command to query or configure the COM port. For example, to modify the baud rate 9600 to 115200 bps via **/dev/ttyS1** port:

```
# stty -F /dev/ttyS1 ispeed 115200 ospeed 115200
```

Use the '**getsendreceive**' command to query or configure the COM port. (Refer to Figure 2.7.6-1)

For example, the I-7060 module is connected with COM2 port of the LP-9x2x, and sending the command '**$01M**' to query the module name which baud rate is 115200 bps connect with /dev/ttyS0 port, it will get a response: '**!017060**'.

Figure 2.7.6-1. Use the 'getsendreceive' command to query or configure the COM port

3. Instructions for LinPAC AM335x PAC

This chapter provides a brief introduction of the LinPAC AM335x PAC service tools and its benefits. There are several tools and utilities built-in and designed for use with LinPAC AM335x PAC. Some of these are pre-installed on LinPAC AM335x PAC and can work directly on LinPAC AM335x PAC, and some of these are supporting tools and can help you to manage the LinPAC AM335x PAC remotely on a PC.

3.1. Basic Linux Command

The Linux basic command can be used to set Linux OS or get system information in the LinPAC AM335x PAC. This section provides an introduction to some of the more commonly used Linux instructions. These Linux instructions are similar to those used in DOS and are generally expressed in lower case letters.

3.1.1. ls: lists the file information (Equivalent DOS Command: dir)

Parameter	Description	Example
–l	Lists detailed information related to the files	ls –l
–a	Lists all files, including hidden files	ls –a
–t	Lists the files arranged in date/time order	ls –t

3.1.2. cd directory: Changes directory (Equivalent DOS Command: cd)

Parameter	Description	Example
..	Move to the parent directory	cd ..
~	Move back to the root directory	cd ~
/	Path component separator	cd /root/i8k

3.1.3. mkdir: creates a subdirectory (Equivalent DOS Command: md)

Parameter	Description	Example
-p	No error if the subdirectory exists, and creates the parent directories as needed	mkdir -p directory

3.1.4. rmdir: deletes the subdirectory which must be empty (Equivalent DOS Command: rd)

Parameter	Description	Example
-p	Removes the specified DIRECTORY, then attempts to remove each parent directory component with the same path name	rmdir -p directory

3.1.5. rm: deletes (removes) the file or directory (Equivalent DOS Command: delete)

Parameter	Description	Example
–i	Displays a warning message before deleting	rm –i test.exe
–r	Deletes the directory even if it isn't empty	rm –r test.exe
–f	No warning message displayed when deleting	rm -f test.exe

3.1.6. cp: copies one or more files (Equivalent DOS Command: copy)

Parameter	Description	Example
–R	Performs a recursive copy	cp -R test bak
–i	Displays a confirmation prompt before overwriting	cp –i test bak
-l	Links the files instead of copying them	cp –l test bak

3.1.7. mv: moves or renames a file or directory (Equivalent DOS Command: move)

Parameter	Description	Example
-f	Does not display a confirmation prompt before overwriting	cp –f sour des
–i	Displays a confirmation prompt before overwriting	cp –i /sour /des

3.1.8. pwd: displays the full path of the current working directory

3.1.9. who: displays a list of the users current logged on

3.1.10. chmod: changes the access permissions for a file

Syntax: chmod ??? file → ??? means owner: group: all users

For example: chmod 754 test.exe

7 5 4 → 111 (read, write, execute)

101 (read, write, execute)

100 (read, write, execute)

The first number 7: the **owner** can read and write and execute files

The second number 5: the **group** can only read and execute files

The third number 4: **all users** can only read files

3.1.11. uname: displays the Linux version information

3.1.12. ps: displays a list of the currently active procedures

3.1.13. ftp: transfers a file using the file transfer protocol (FTP)

Syntax: ftp IPAdress (Example: ftp 192.168.0.200 → connect to ftp server)

Syntax	Description
!	temporarily exits the FTP
exit	back to the ftp
bin	transfers files in 'binary' mode
get	downloads a file from the LinPAC to the Host (Forexample: get /mnt/hda/test.exe c:/test.exe)
put	uploads a file from the host to the LinPAC (For example: put c:/test.exe /mnt/hda/test.exe)
bye	exits FTP

3.1.14. telnet: establishes a connection to another PC via Telnet terminal

Syntax: telnet IPAddress

For example telnet 192.168.0.200 (will allow remote control of the LinPAC AM335x PAC)

3.1.15. date: prints or sets the system date and time

3.1.16. hwclock: queries and sets the hardware clock (RTC)

Parameter	Description	Example
-r	Reads the hardware clock and prints the time on a standard output.	hwclock -r
–w	Sets the hardware clock to the current system time	hwclock -w

3.1.17. netstat: displays the current state of the network

Parameters: [-a]: list all states (For example: netstat -a)

3.1.18. ifconfig: displays the ip and network mask information (Equivalent DOS Command: ipconfig)

3.1.19. ping: used to test whether the host in a network is reachable

Syntax: ping IPAddress

For example ping 192.168.0.1

3.1.20. clear: clears the screen

3.1.21. passwd: used to change the password

3.1.22. reboot: reboots the LinPAC (or use 'shutdown –r now')

3.1.23. wget: get the file from the web link

Syntax: wget [option] [URL]

For example:

wget http://ftp.icpdas.com/pub/cd/linpac/napdos/lx-series/sdk /linpac_x86_sdk.tgz

Note: Using static IP address may have to be configured in the '/etc/resolv.conf' file.

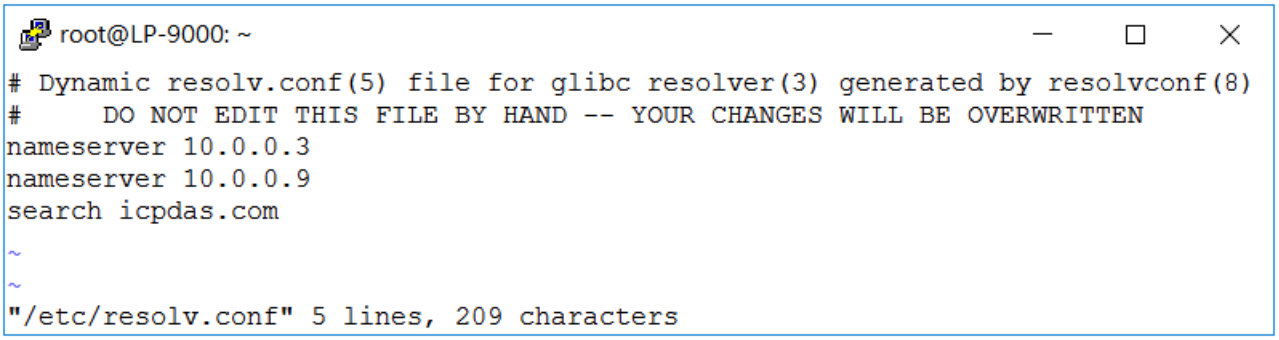
At the Command Prompt, edit the **/etc/resolv.conf** file by modifying the settings.

For example:

nameserver 10.0.0.3

nameserver 10.0.0.9

search icpdas.com



3.1.24. update-rc.d: install and remove System-V style init script links

update-rc.d [-n] name defaults (Example: update-rc.d hello defaults)

update-rc.d [-n] [-f] name remove (Example: update-rc.d –f hello remove)

3.2. i-Talk Utility

The **i-Talk utility** can make the convenient for users to access the modules and hardware in the LinPAC AM335x PAC and can be found in the path /usr/sbin/iTalk. An overview of the i-Talk utility functions is given below:

LP-8x2x/9x2x

Instruction	Description
getlist	Lists the names of all modules inserted in the LP-8x2x/9x2x
setdo	Sets the Digital Output value for I-8K modules
getdi	Reads the Digital Input value for I-8K/9K modules
setport	Sets the Port offset value for the module
getport	Reads the Port offset value for the module
getsendreceive	Send ASCII command and wait response from a serial
getreceive	Get ASCII response from a serial module
getmrtu	Send a Modbus/RTU command and wait for a response from a serial module
getmtcp	Send a Modbus/TCP command and wait for a response from a serial module
getmasc	Send a Modbus/ASC command and wait for a response from a serial module
getpactype	List the FPGA version (Shown as **'0x82'** on the **LP-8x21**, and **'0x93'** on the **LP-9x21**)
wdt	Set the WatchDog Timer (WDT) for the LP-8x2x/9x2x

Below table lists the demos that show how to use the I-talk utility. In the demo, the I-8024W (**AO Module**), I-8017HW (**AI Module**) and I-8055W (**DIO Module**) are all used and they are plugged into the slots 1, 2 and 3 of the LinPAC separately. Typing the name of the instruction will display usage details for the instruction.

Instruction	Example
getlist	*getlist* Lists the names of all modules inserted in the LinPAC-8x2x/9x2x
setdo	*setdo {slot} {data}* ***setdo 3 3*** Sets channels 1 and 2 on the I-8055W module to ON
getdi	*getdi {slot} {type}* ***getdi 3 8*** Reads the 8-bit Digital Input value from the I-8055W module
setport	*setport {slot} {offset} {data}* **setport 1 0 20** Set the dec value 20 to offset 0 of slot 1
getport	*getport {slot} {offset}* **getport 1 0** Get the dec value from offset 0 of slot 1
getsendreceive	*getsendreceive {slot} {1} {timeout} {command}* **getsendreceive 2 1 1 '$00M'** Send command $00M to the module at slot 2 and wait response *getsendreceive {slot} {comport} {timeout} {command} {baudrate}* **getsendreceive 0 3 1 '$01M' 9600** Send command $01M to the module at COM3 and wait response
getreceive	*getreceive {slot} {comport} {timeout} {baudrate} {format}* **getreceive 0 2 5 9600 8n1** Get response from the module at COM2 with 8n1 format in 0.5s

LP-22xx/52xx

Instruction	Description
setxvdo	Set digital output value to XV-Board
setxvao	Set analog output value to XV-Board
getxvdi	Get digital input value from XV-Board
getxvai	Get analog input value from XV-Board
getxvdo	Get digital output value from XV-Board
getxvao	Get analog output value from XV-Board
setmodbus	Set a Modbus command to modbus device
getmodbus	Get the status of modbus device
Rsw	Get the rotary switch ID
Led	Set LED (L1~L2) for LP-52xx Set LED (L1~L3) for LP-22xx

4. Getting started with the LinPAC AM335x SDK

The 'LinPAC_AM335x SDK' is a development toolkit provided by ICP DAS, which can be used to easily develop custom applications for the LP-22xx/52xx/8x2x/9x2x embedded controller platform. The toolkit consists of the following items:

- LinPAC_AM335x SDK (Linaro GCC toolchain, Libraries, header, examples files, etc.)
- Code:: Blocks project file (Windows platform only)
- Basic Linux commands (Windows platform only)

The topic provides LinPAC_AM335x SDK installation instructions for the following platforms:

- Linux
  - Download/Install LinPAC_AM335x SDK on Linux
- Windows
  - Download/Install LinPAC_AM335x SDK on Windows
  - Integrating LinPAC_AM335x SDK with Code:: Blocks IDE

4.1. Introduction of the LinPAC AM335x SDK

This section will discuss some of the techniques that are adopted in the LinPAC_AM335x SDK, including detailed explanations that describe how to easily use the LinPAC_AM335x SDK. The LinPAC_AM335x SDK is based on Cygwin and is also a Linux-like environment for Microsoft Windows systems, and provides a powerful GCC cross-compiler and an IDE (Integrated Development Environment) that enables LinPAC_AM335x SDK applications to be quickly developed. Therefore, once an application has been created, the LinPAC_AM335x SDK can be used to compile it into an executable file that can be run on the LinPAC_AM335x SDK embedded controller.

Note:

1. The latest Linux AM335x SDK is integrate AM335x series SDK. Select the appropriate software for your controller.
2. The names of all the I/O module's API functions must begin with the prefix 'I8K'.
3. The I-8K and I-9K I/O modules using the same API function and examples.
4. More detailed information, user can refer to **readme.txt** file here:
   C:\cygwin\LinPAC_am335x_SDK\examples\readme.txt file
   or root@LinuxPC-ICPDAS:/icpdas/linpac_am335x_sdk/i8k/examples/readme.txt

4.1.1. Introduction to Cygwin

Cygwin is a collection of free software tools originally developed by Cygnus Solutions to allow various versions of Microsoft Windows to act somewhat like a UNIX system. Cygwin is a Linux-like environment for Windows consisting of two parts:

(1) A DLL (cygwin1.dll) which acts as a Linux emulation layer providing substantial Linux API functionality.

(2) A collection of tools that provide users with the Linux look and feel.

4.1.2. Introduction to Cross-Compilation

Generally, program compilation is performed by running a compiler on the build platform. The compiled program will then run on the target platform. Usually, these two processes are intended for use on the same platform. However, if the intended platform is different, the process is called **cross compilation**, where source code on one platform can be compiled into executable files to be used on other platforms. For example, if the '**arm-linux-gnueabihf-gcc**' cross-compiler is used on an x86 windows platform, the source code can be compiled into an executable file that can run on an arm-linux platform.

So why use cross compilation? In fact, cross compilation is sometimes more complicated than normal compilation, and errors are easier to make. Therefore, this method is often only employed if the program cannot be compiled on the target system, or if the program being compiled is so large that it requires more resources than the target system can provide. For many embedded systems, cross compilation is the only possible approach.

4.1.3. Download the LinPAC AM335x SDK

- **For Windows systems: (**Extract the .exe file into to the **C:\ driver.)**

LinPAC	Download Path
LP-22xx/52xx	ftp://ftp.icpdas.com/pub/cd/linpac/napdos/lp-5000/lp-52xx/lp-5231/sdk/lp52xx_am335x_sdk_for_windows.exe
LP-8x2x/9x2x	ftp://ftp.icpdas.com/pub/cd/linpac/napdos/lp-9x2x/sdk/linpac_am335x_sdk_for_windows.exe

- **For Linux systems: (**Extract the .bz2 file into to the **root (/) directory.)**

LinPAC	Download Path
LP-22xx/52xx	ftp://ftp.icpdas.com/pub/cd/linpac/napdos/lp-5000/lp-52xx/lp-5231/sdk/lp52xx_am335x_sdk_for_linux.tar.bz2
LP-8x2x/9x2x	ftp://ftp.icpdas.com/pub/cd/linpac/napdos/lp-9x2x/sdk/linpac_am335x_sdk_for_linux.tar.bz2

Note: We recommend user to change user ID to become root by 'sudo' or 'su' command.

4.2. Quick Installation of the LinPAC AM335x SDK

4.2.1. Download/Install LinPAC AM335x SDK on Linux

1. To create a '**icpdas**' folder in root directory, maybe you need to change the root user by '**sudo**' or '**su**' command. (Refer to Figure 4.2.1-1)

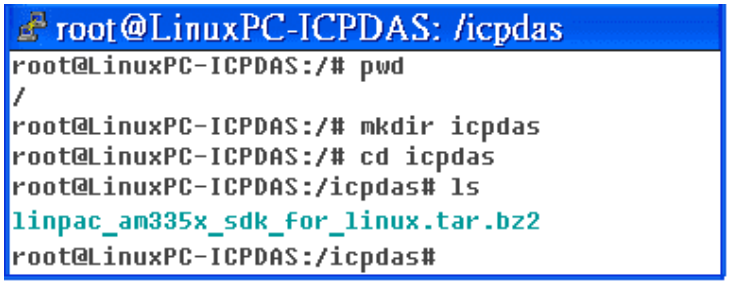


Figure 4.2.1-1. Create a directory named 'icpdas'

2. Insert the installation CD into your CD-ROM driver. (Refer to Figures 4.2.1-2 and 4.2.1-3) Locate the '**linpac_am335x_sdk_for_linux.tar.bz2**' file in the \napdos\lp-9x2x\SDK\ folder or visit the ICP DAS website to download the latest version of the LinPAC_AM335x SDK.

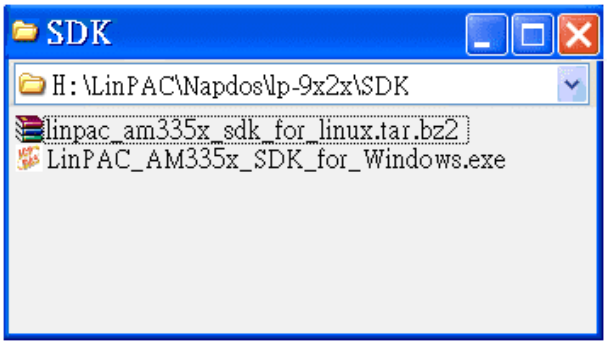


Figure 4.2.1-2.

Figure 4.2.1-3.

3. Try the following command to decompress file. (Refer to Figure 4.2.1-4)

```
#  tar  jxvf  linpac_am335x_sdk_for_linux.tar.bz2
```

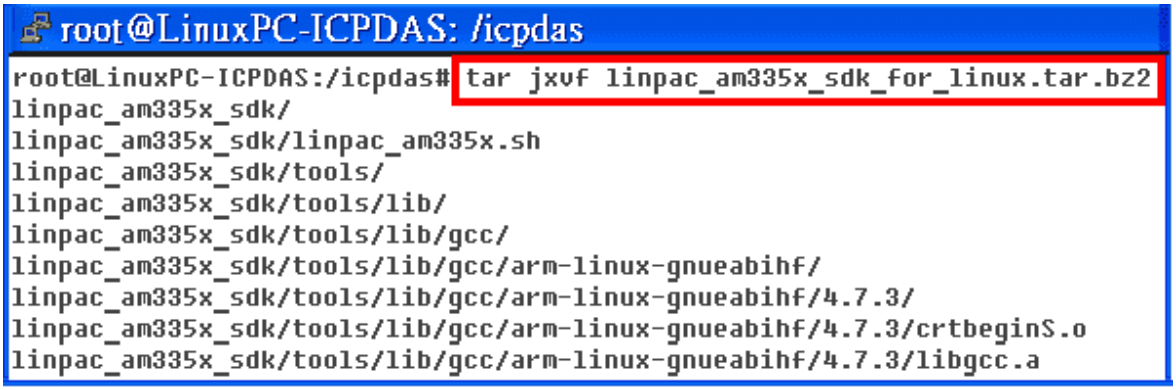


Figure 4.2.1-4. Decompress '.tar.bz2' file

4. Before compiling the program, you need to set LinPAC_AM335x SDK path in environment variables: using the provided environment variable script, which is called **linpac_am335x.sh** (Refer to Figure 4.2.1-5).

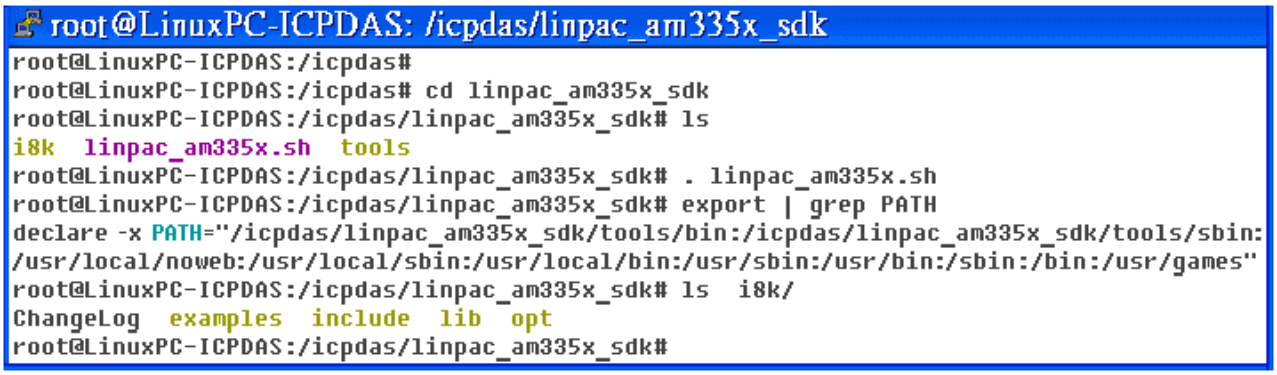


Figure 4.2.1-5. Setting environment variables for LinPAC_AM335x SDK

5. Type '**make**' on the command line it will execute the compile command according to the Makefile. (Refer to Figure 4.2.1-6)

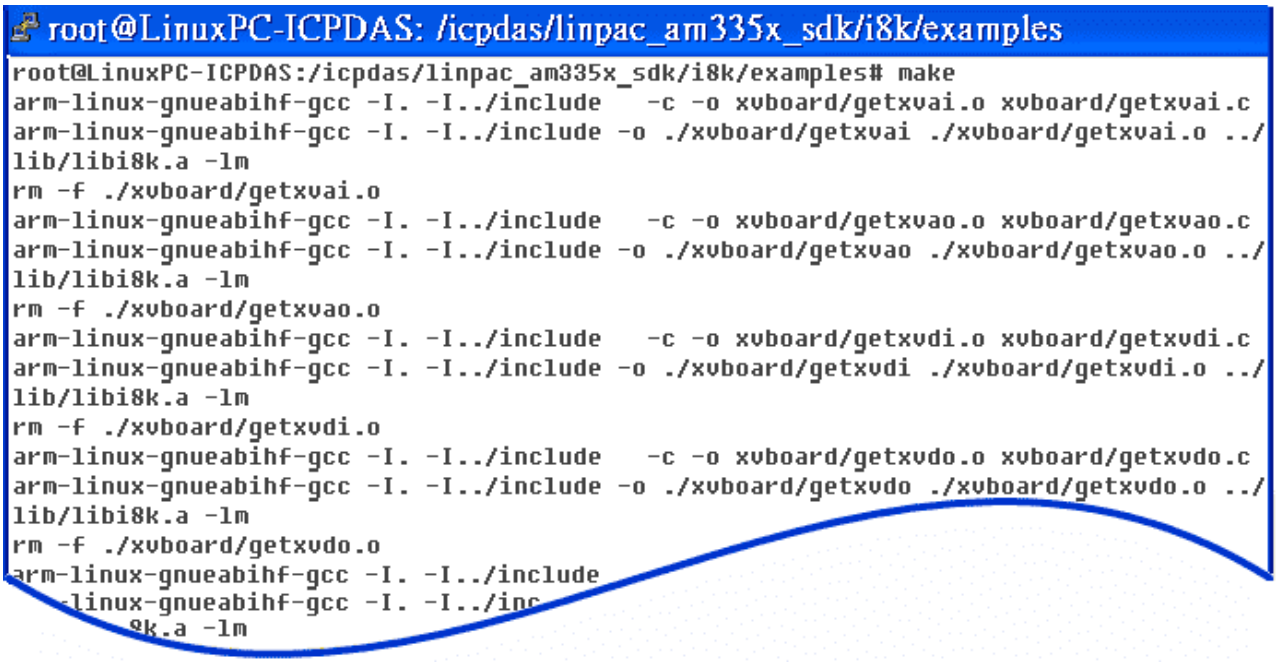


Figure 4.2.1-6. Compiling demo code according to the Makefile

4.2.2. Download/Install LinPAC AM335x SDK on Windows

The LinPAC_AM335x_SDK_for_Windows.exe provides compilers, library, header, examples, and IDE workspace file (for Code::Blocks project).

- ❑ Insert the installation CD into your CD-ROM driver.
- ❑ Open the \napdos\LP-9x21\SDK\ folder and double-click the icon for the 'LinPAC_AM335x_SDK_for_Windows.exe' file, when the Setup Wizard is displayed, click the 'Next>' button to continue, refer to Figures 4.2.2-1 and 4.2.2-2.

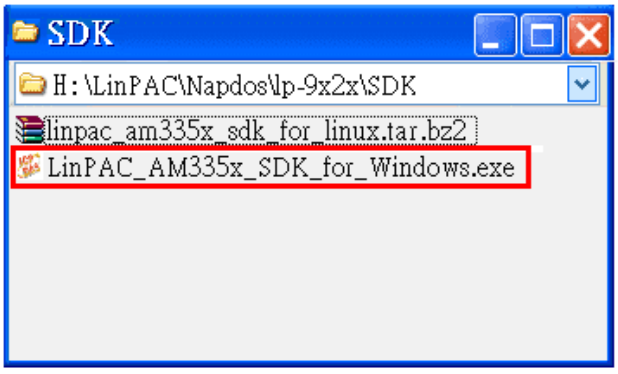


Figure 4.2.2-1.

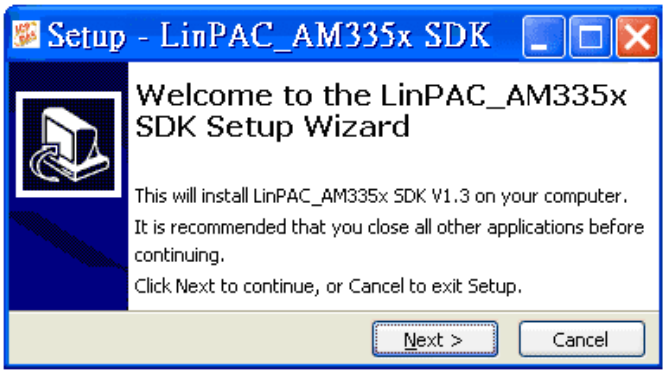


Figure 4.2.2-2.

- ❑ Click the '**I accept the agreement**' option and then click the '**Next**' button (refer to Figure 4.2.2-3), and select Start Menu Folder option and then click the '**Next**' button, refer to Figure 4.2.2-4.

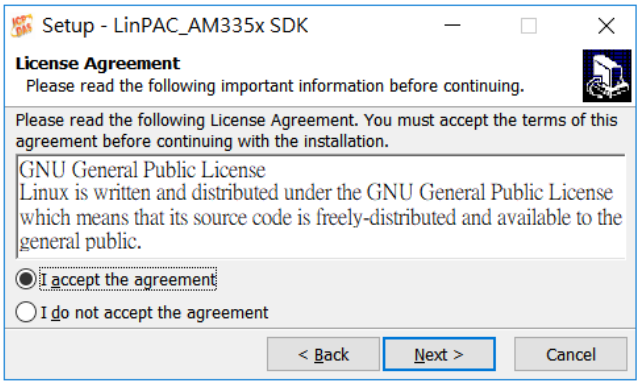


Figure 4.2.2-3.

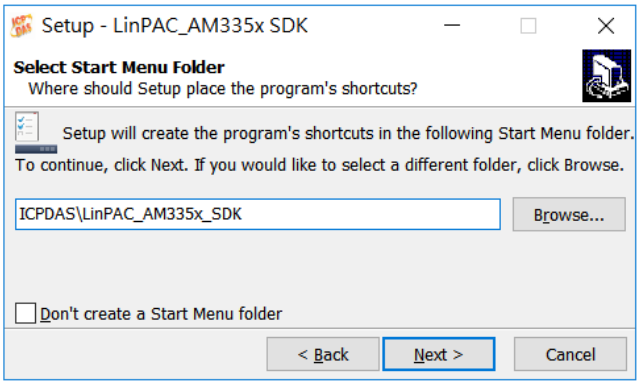


Figure 4.2.2-4.

- The LinPAC_AM335x SDK files will be extracted and installed and a progress bar will be displayed to indicate the status, refer to Figure 4.2.2-5.
- Once the software has been successfully installed, click the '**Finish**' button to complete the development toolkit installation, refer to Figure 4.2.2-6.

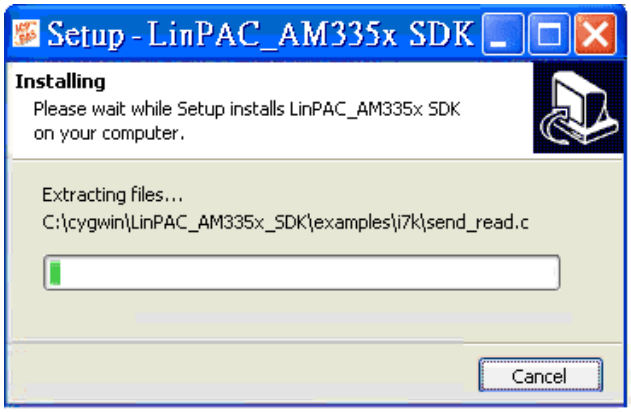


Figure 4.2.2-5.

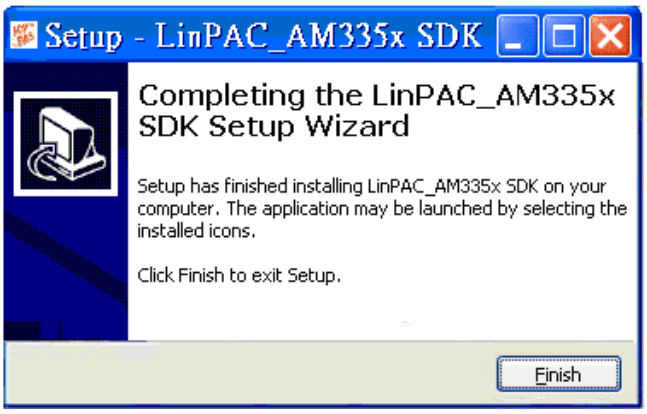


Figure 4.2.2-6.

- Open the LinPAC_AM335x SDK installation directory, the default data directory location is '**C:\cygwin**', the user can see the contents of the folder. Refer to Figures 4.2.2-7 and 4.2.2-8.

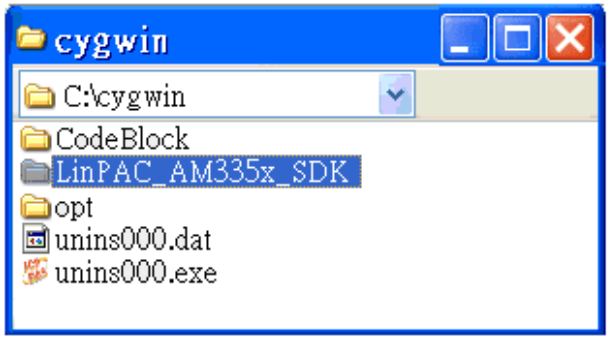


Figure 4.2.2-7.

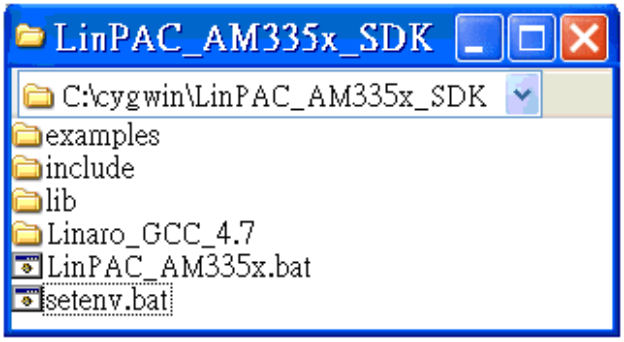


Figure 4.2.2-8.

- From the desktop, double-click the shortcut icon for the 'LinPAC_AM335x Build Environment' or click the 'Start' > 'Programs' > 'ICPDAS' > 'LinPAC_AM335x_SDK' > 'LinPAC_AM335x Build Environment'.

  A Command Prompt window will then be displayed that allows applications for the LinPAC_AM335x to be compiled. Refer to Figures 4.2.2-9 and 4.2.2-10.

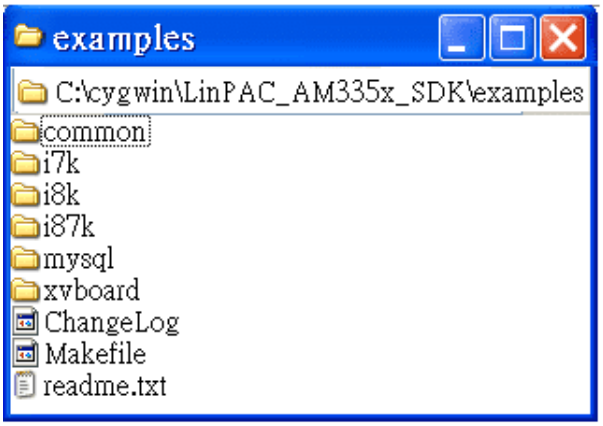


Figure 4.2.2-9.

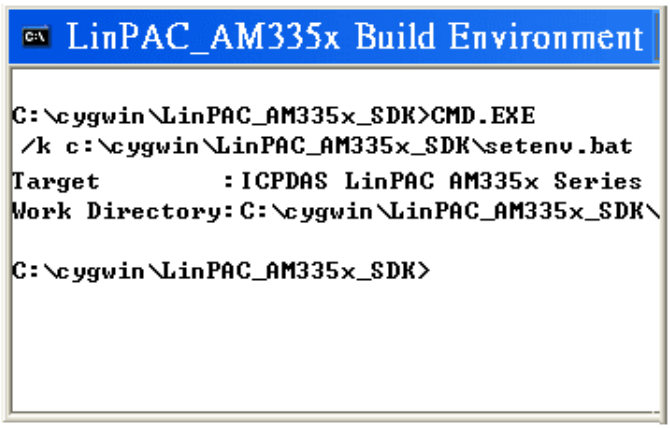


Figure 4.2.2-10.

- Type '**make**'. A Command Prompt window will then be displayed that allows applications for the LinPAC_AM335x to be compiled. Refer to Figure 4.2.2-11.

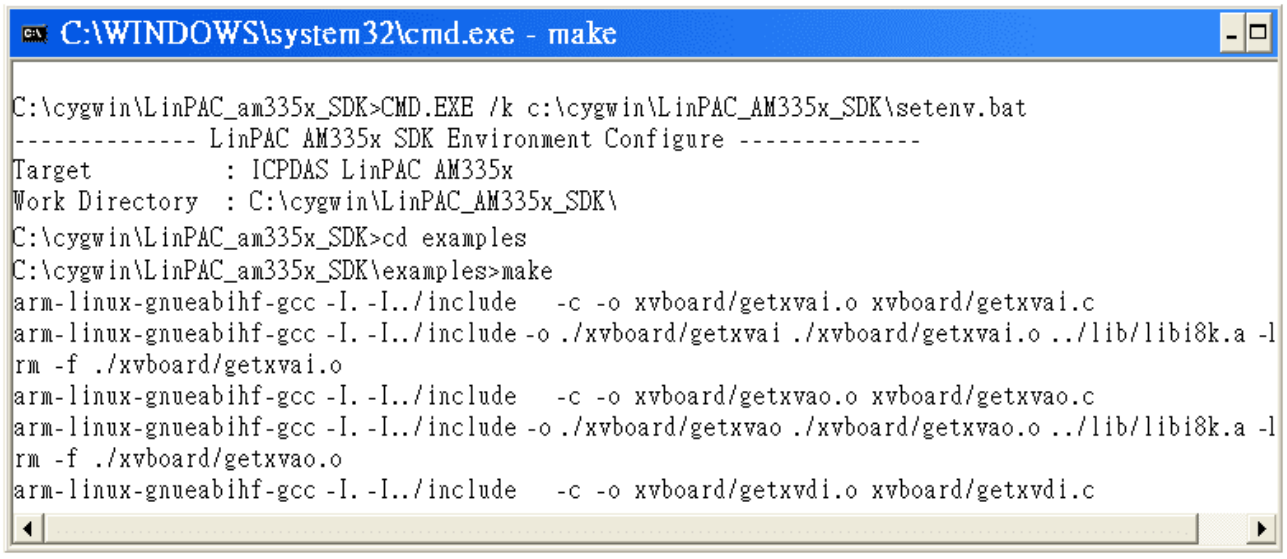


Figure 4.2.2-11. Compiling demo code according to the Makefile

4.2.3. Integrating LinPAC AM335x SDK with Code::Blocks IDE

This tutorial gives you easy-to-follow instructions, with screenshots, for setting up a compiler (the Linaro GCC compiler), a tool that will let you turn the code that you write into programs, and Code::Blocks IDE, a free development environment. This tutorial explains how to integrate LinPCA AM335x SDK with Code::Blocks IDE on Windows platform.

Step 1: Download Code::Blocks IDE.

- ❑ Go to this website: http://www.codeblocks.org/downloads/binaries
- ❑ Go to the Windows 2000/XP/Vista/7 section, and download Windows version.

Step 2: Install Code::Block IDE.

- ❑ The default install location is the C:\Program Files\CodeBlocks folder.
- ❑ A complete manual for Code::Blocks is available here: http://www.codeblocks.org/user-manual

Step 3: Running in Code::Block IDE.

- ❑ All files and settings that are included in a LinPAC_AM335x_SDK workspace file.
- ❑ Open the **C:\cygwin\CodeBlock** folder, and double click the '**LinPAC_AM335x_SDK**' as below (Refer to Figure 4.2.3-1):

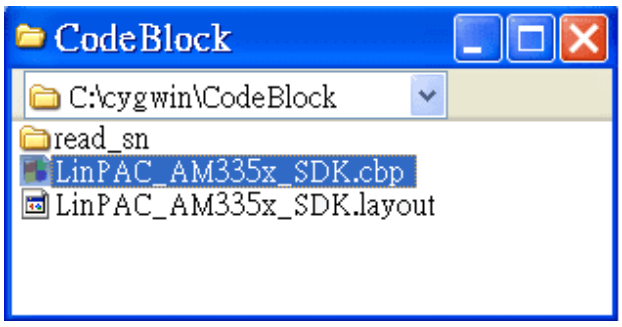


Figure 4.2.3-1. Startup the LinPAC AM335x SDK

❑ Following window will come up (Refer to Figure 4.2.3-2):

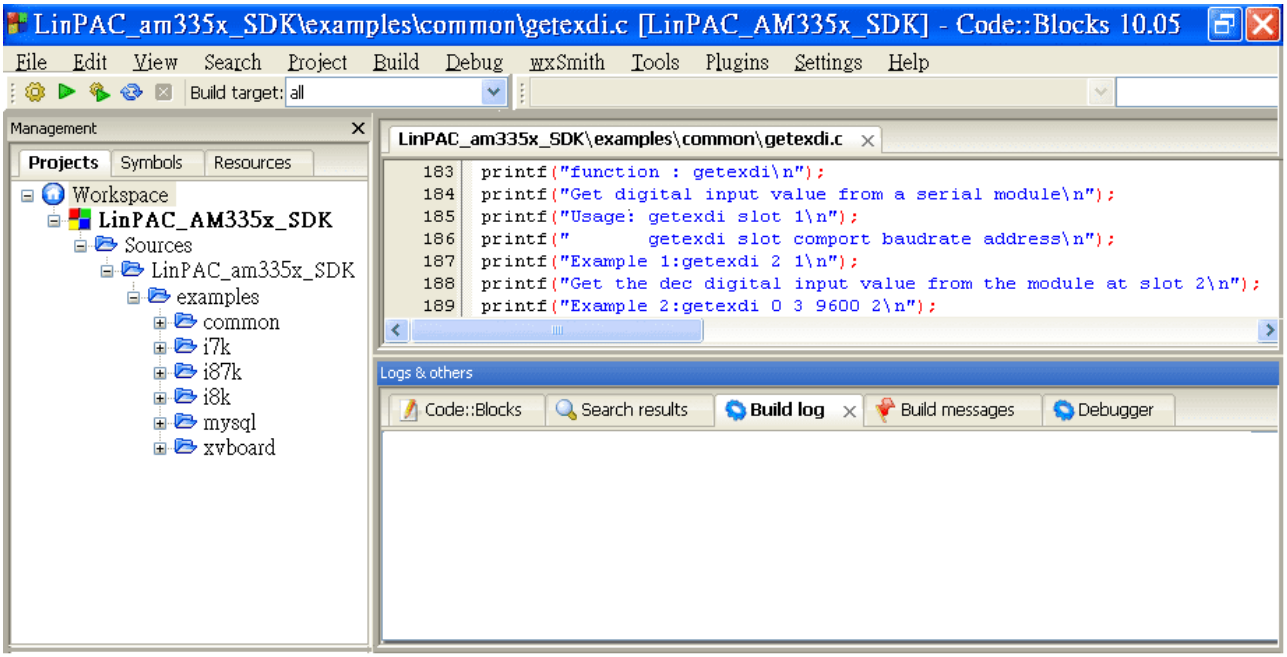


Figure 4.2.3-2. Startup the LinPAC AM335x SDK

❑ Check compiler settings for Linaro GCC cross compiler: Click 'Settings' > 'Compiler' > 'Toolchain executables tab' (Refer to Figure 4.2.3-3):

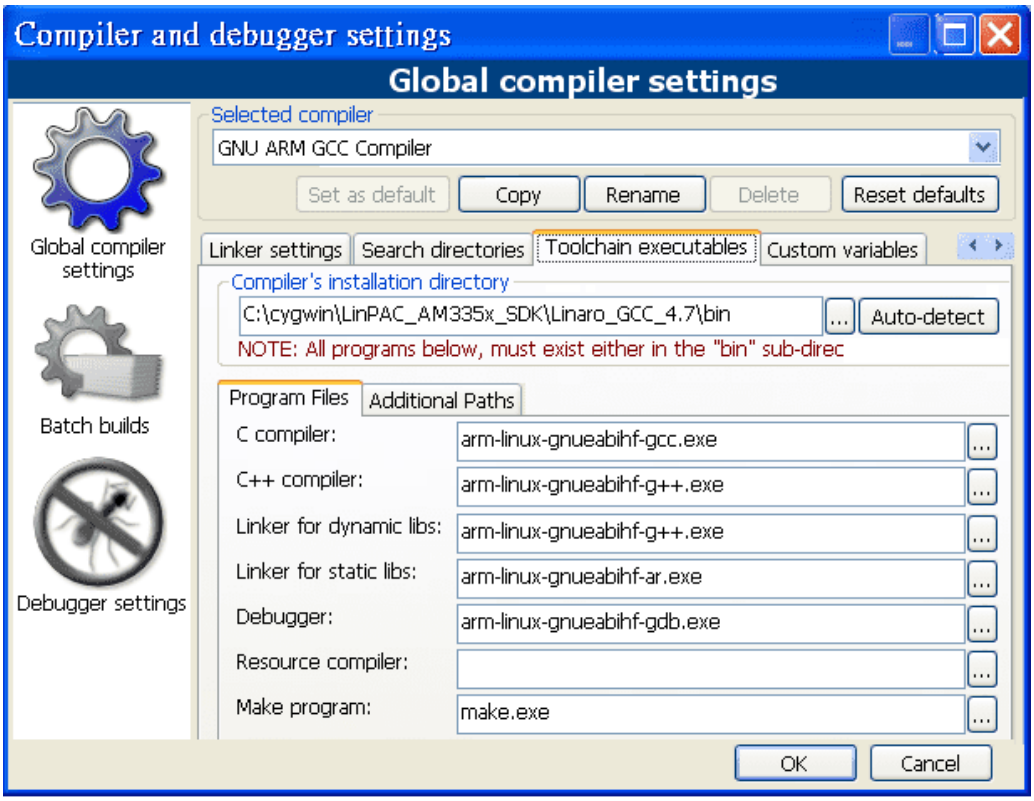


Figure 4.2.3-3. Check compiler settings

- Click **Build** options, and it will compile the LinPAC_AM335x project completely. (Refer to Figure 4.2.3-4)

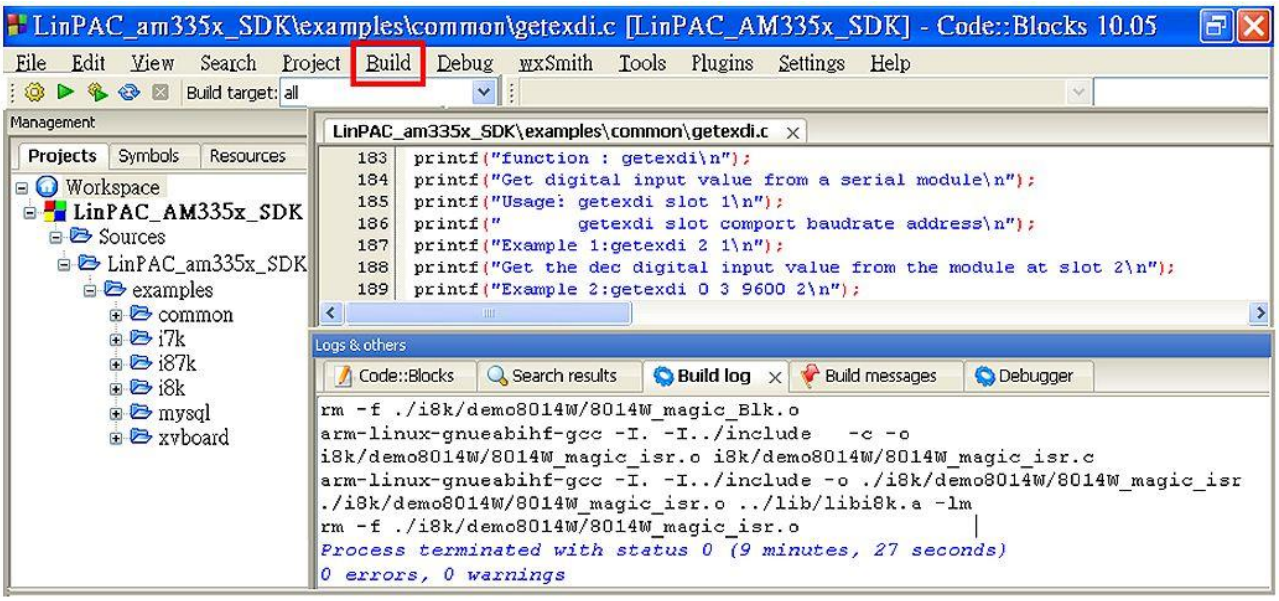


Figure 4.2.3-4. Compiling a C program

【**Note**】**If you observer some characters may not display properly in cmd.exe**, change the code page for the console only, do the following:

- Double-click the shortcut icon for the 'LinPAC_AM335x Build Environment'. (Refer to Figure 4.2.3-5)

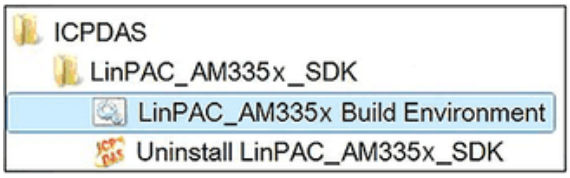


Figure 4.2.3-5. Click the 'LinPAC_AM335x Build Environment'

- Type command: **chcp 65001**. (Refer to Figures 4.2.3-6 and 4.2.3-7)

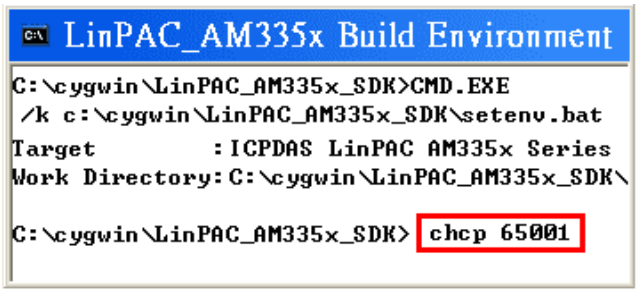


Figure 4.2.3-6.

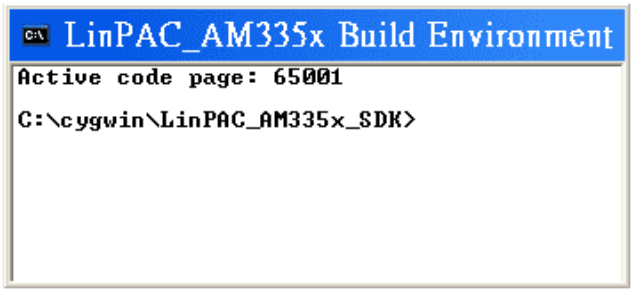


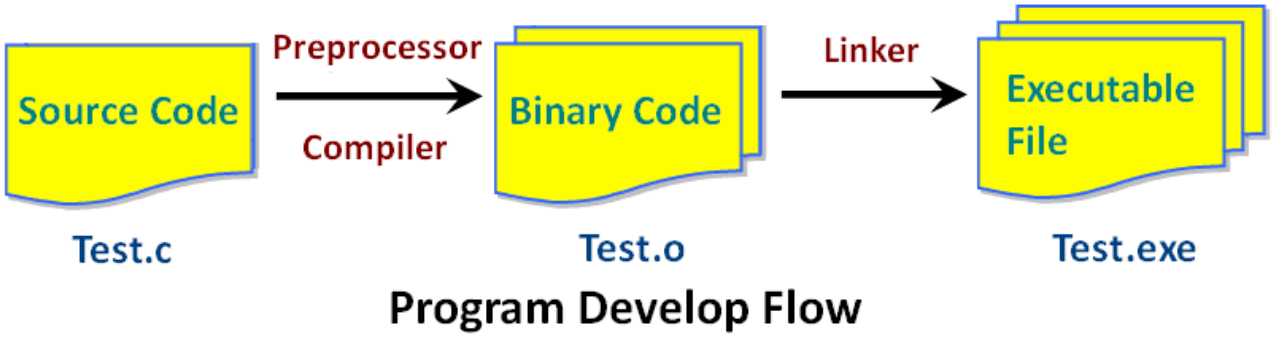
Figure 4.2.3-7.

4.3. Your First Program

In this section, we will introduce how to compile the helloworld.c file to helloworld **executable file** and executes this on the LinPAC AM335x PAC. In this example, no ICP DAS modules are used. To create a demo program with C language that includes the following main steps:

1. Find demo 'helloworld.c' in SDK
2. Compile the demo on Windows/Linux PC using SDK
3. Upload and execute the demo on LinPAC AM335x PAC
4. Execute the application on LinPAC AM335x PAC at boot time

All main steps will be described in the following subsection.



4.3.1. A simple example- helloworld.c

There are three choices available to you:

1. **Coding a helloworld.c file**
2. **To modify/create demo 'helloworld.c'**

Using a programmer's editor, such as PSPad. Create a demo - helloworld.c file. Note that the code is case-sensitive. Refer to Figure 4.3.1-1 for more details.

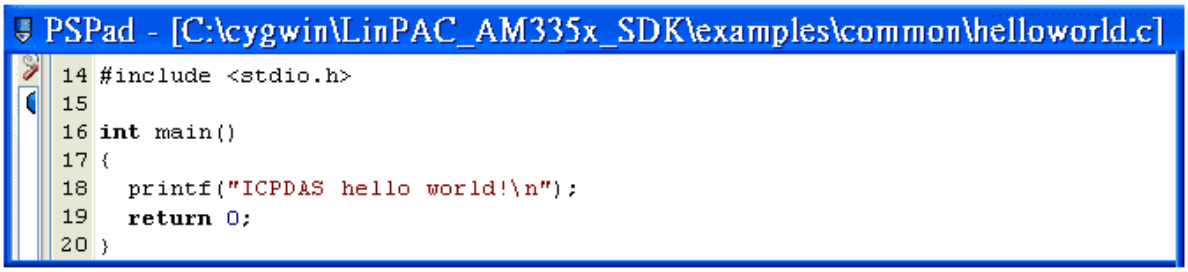


Figure 4.3.1-1. Creating helloworld.c

3. **Find the demo in SDK**

The process can be divided into two steps, which are described below:

- Open the LinPAC_AM335x SDK and then type '**cd examples/common**' to change the path to **C:/cygwin/LinPAC_AM335x_SDK/examples/common**.
- Type '**dir/w**' or '**ls**' command and to display the contents of the directory and confirm that the helloworld.c file is present. Refer to Figure 4.3.1-2 for more details.

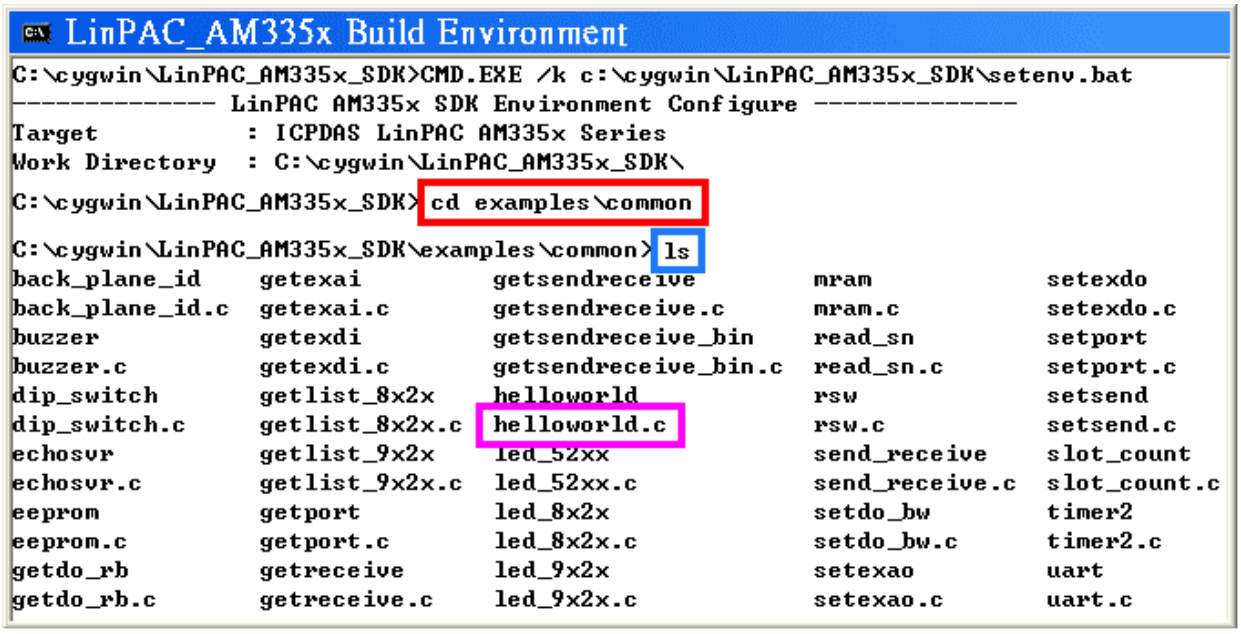


Figure 4.3.1-2. Display and confirm the contents of the common directory

4.3.2. Compile Demo- helloworld.c

Type the command '**arm-linux-gnueabihf-gcc –o helloworld.exe helloworld.c**' to compile helloworld.c into helloworld.exe, then type '**dir/w**' or '**ls**' command to display the contents of the directory and confirm that the helloworld.exe file has been created. (Refer to Figure 4.3.2-1)

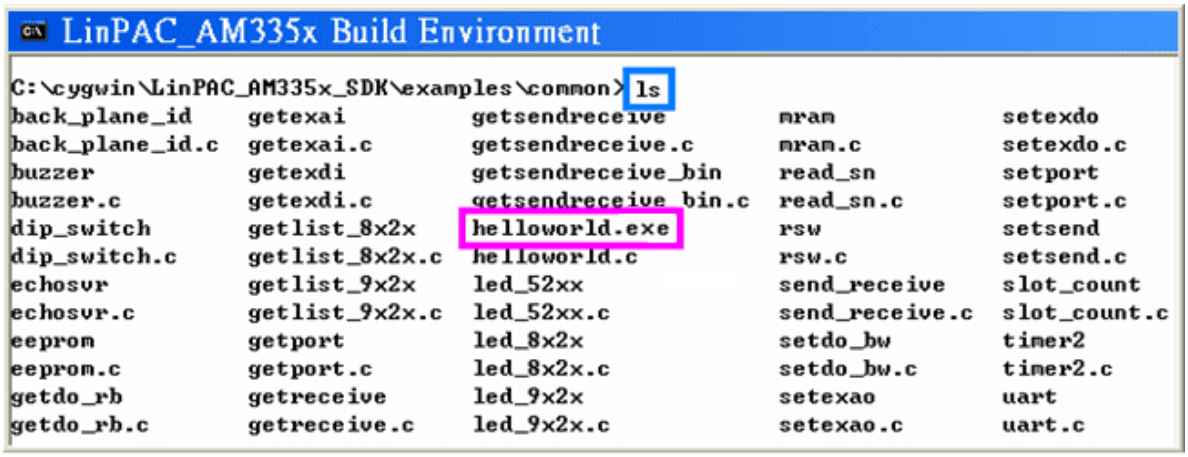


Figure 4.3.2-1. Executable file - helloworld.exe

4.3.3. Execute Demo- helloworld.exe

The user can refer to below steps to transfer and execute helloworld.exe.

Use 'Dos Command Prompt' and 'FTP' tools to transfer program

Two methods can be used to transfer files to the LinPAC AM335x PAC, with LP-8x21 as an example:

＜Method one＞ Using the 'DOS Command Prompt'

(1) Open a 'DOS Command Prompt' or double-click the shortcut icon for the '**LinPAC_AM335x Build Environment**' and type the ftp IP Address of the LP-8x21 for example, **ftp 192.168.0.200** to establish a connection to the FTP Server on the LP-8x21. When prompted, type the **User_Name (default value is 'root')** and **Password (default value is 'icpdas')** to establish a connection to the LP-8x21.

(2) Before transferring the files to the LP-8x21, type the 'bin' command to ensure that the file is transferred to the LP-8x21 in **binary mode**.

(3) Type the command '**put helloworld.exe**' to transfer the helloworld.exe file to the LP-8x21.

(4) Once the message '**Transfer complete**' is displayed, then transfer process has been completed. To disconnect from the LP-8x21, type the '**bye**' command to return to the PC console. (Refer to Figure 4.3.3-1)

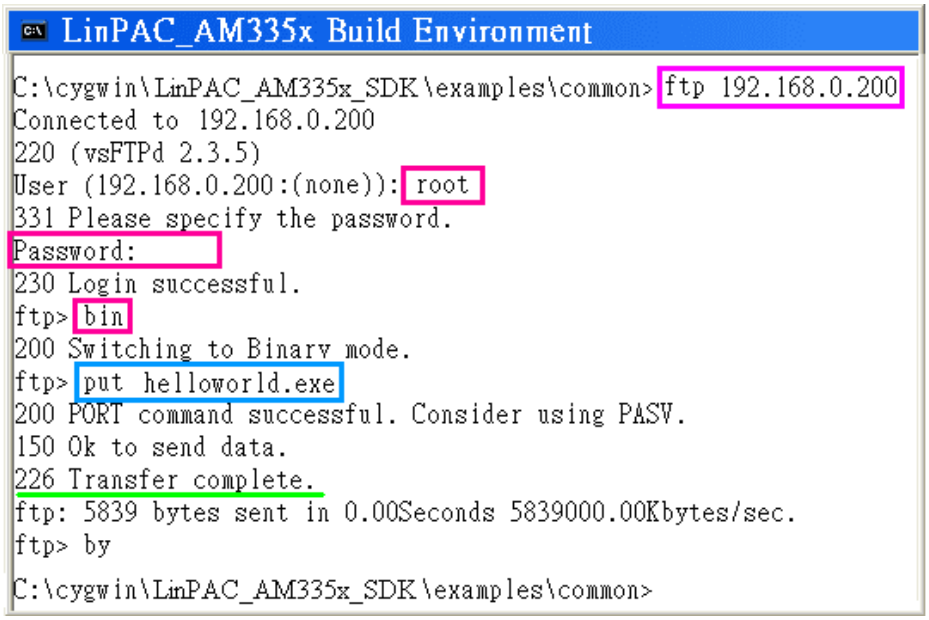


Figure 4.3.3-1. Transfer the helloworld.exe file to the LP-8x21

＜Method two＞ Using an FTP Client:

(1) Open the FTP Software and add an FTP Host to the LP-8x2x. (for example, FileZilla - The free FTP solution for both client and server, https://filezilla-project.org/)

(2) Type the User_Name (default value is '**root**') and Password (default value is '**icpdas**'). Then click the '**Quickconnect** Connect' button to establish a connection to the ftp server on the LP-8x2x. (refer to Figure 4.3.3-2)

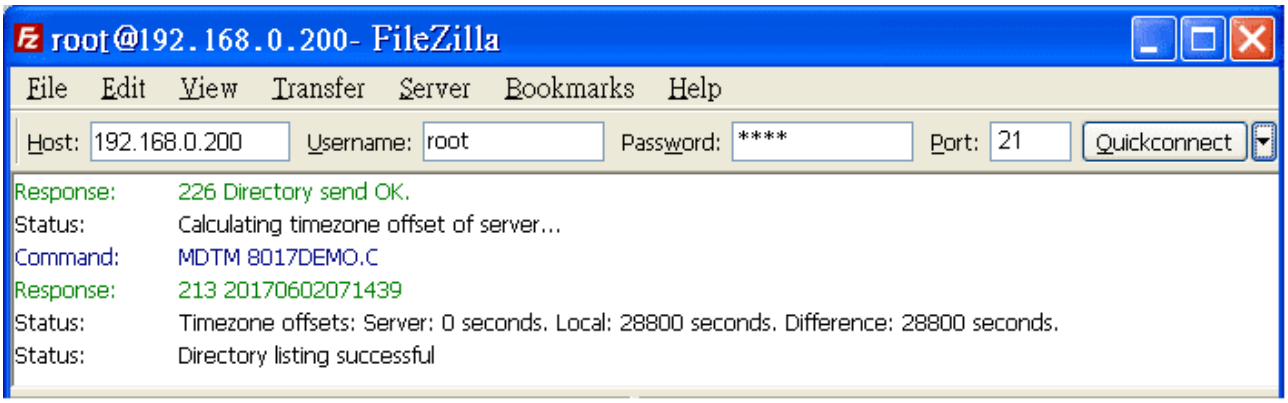


Figure 4.3.3-2. Establish a connection to the ftp server on the LP-8x21

(3) Upload the '**Helloworld.exe**' file to the LP-8x21. (Refer to Figure 4.3.3-3)

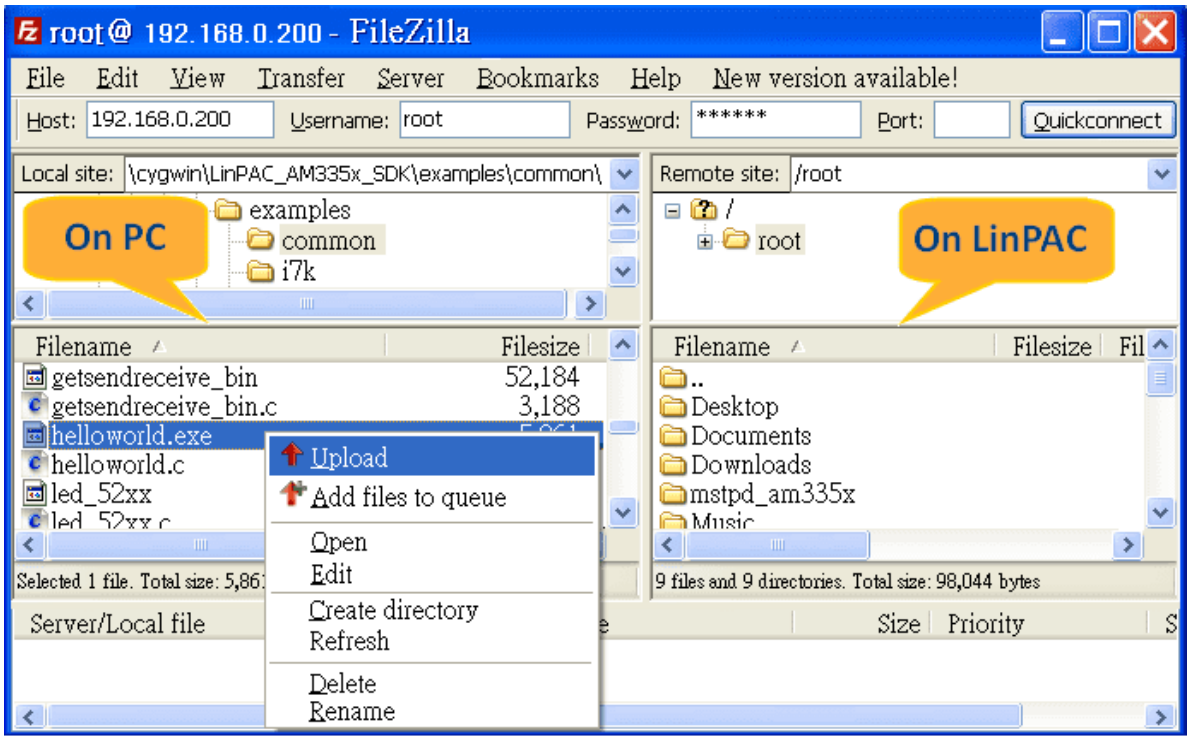


Figure 4.3.3-3. Upload file to the LinPAC

(4) Click the helloworld.exe file in the LP-8x21 to select it and then right-click the file icon and click the '**File Permissions**' option. In the Properties dialog box, type 777 into the Numeric textbox, and then click the OK button. Refer to Figures 4.3.3-4 and 4.3.3-5 for more details.

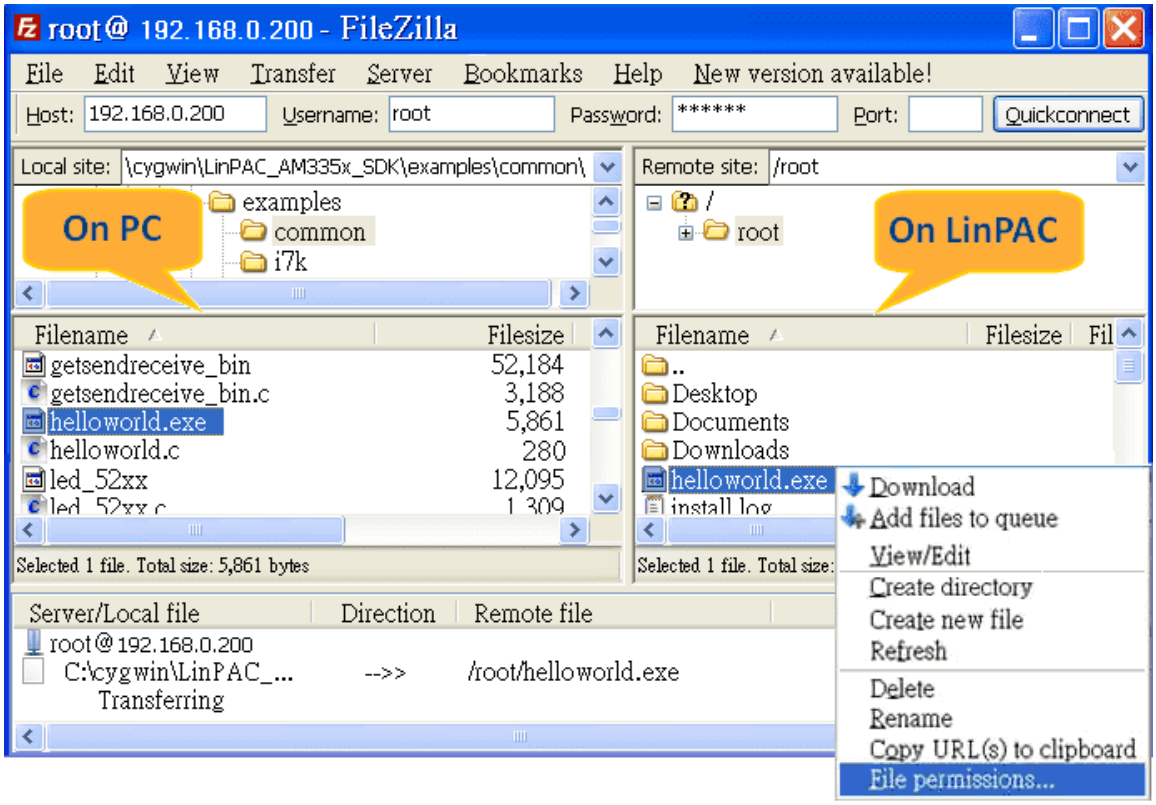


Figure 4.3.3-4. Modify the permissions of files

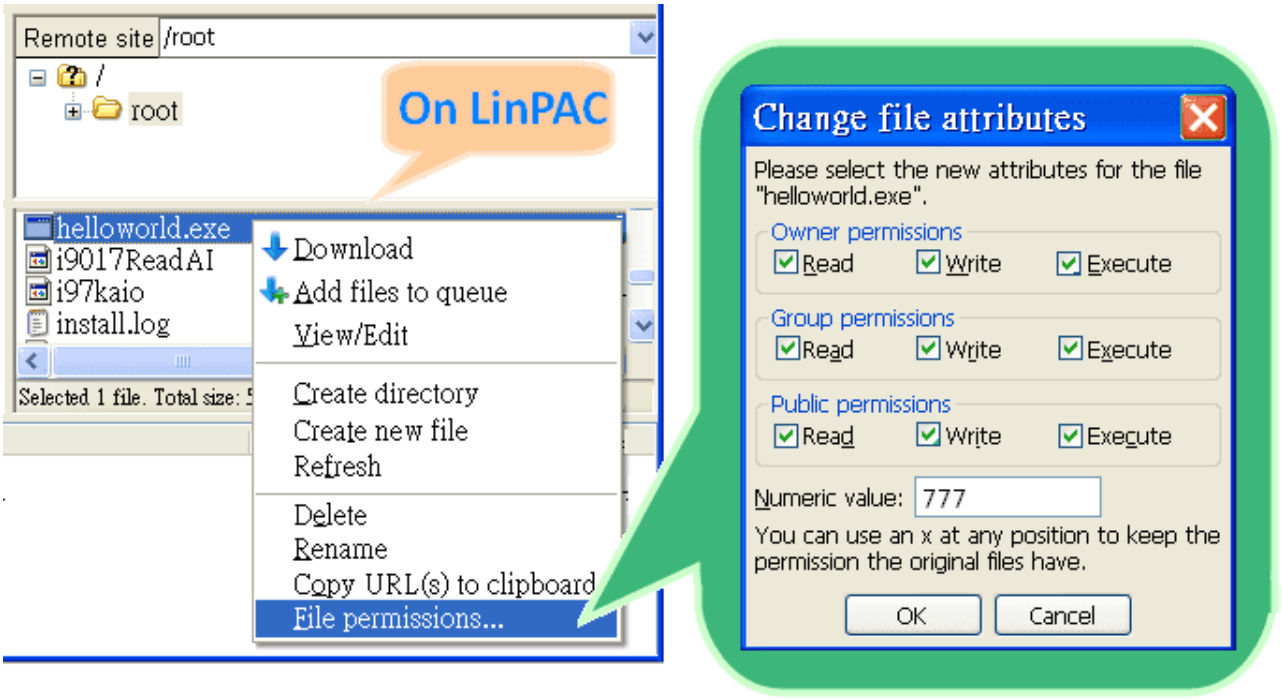


Figure 4.3.3-5. Modify the permissions of files

Use SSH to access LinPAC and execute program

(1) Putty – the free PuTTY is an SSH and telnet client. Download PuTTY tool: http://www.putty.org/

(2) Open a 'Putty Prompt' and type the IP Address of the LinPAC, and the connection type is set to SSH. When prompted, type the User_Name and Password to establish a connection to the LP-8x2x. If the '**#**' prompt character is displayed, it signifies that a connection to the telnet server on the LP-8x2x has been successfully established. (refer to Figure 4.3.3-6)

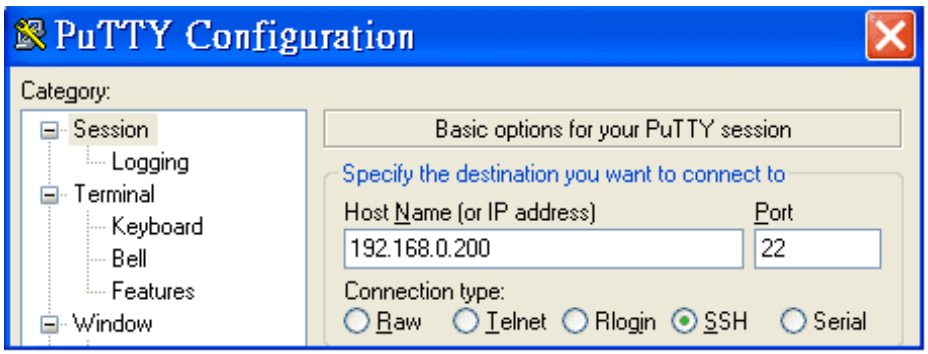


Figure 4.3.3-6. Establish a connection to the telnet server

(3) Type the '**chmod 777 helloworld.exe**' command to change the permissions for the helloworld.exe file. Type the '**ls -l**' command again to list all the files in the /root directory and verify the permissions assigned to the 'helloworld.exe' file. This means that the file is executable. Execute the './helloworld.exe' file by typing and the message 'ICPDAS hello world!' will be displayed.

The compile, transfer and execution processes are now complete. (refer to Figure 4.3.3-7)

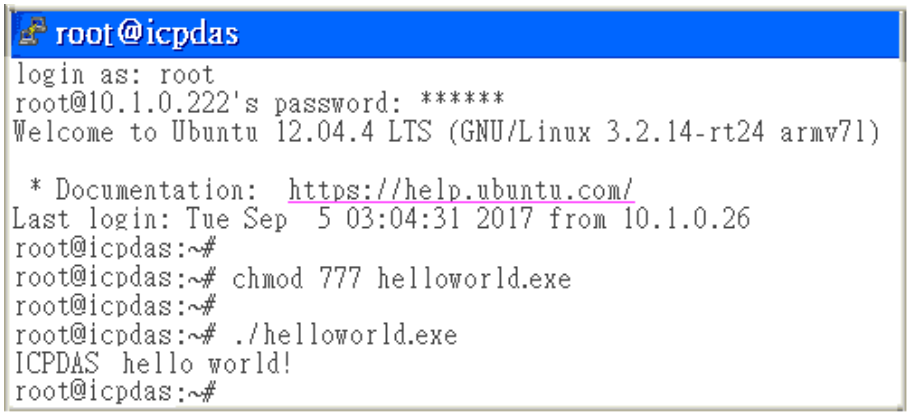


Figure 4.3.3-7. Modify the permissions of files

4.3.4. Execute the application on LinPAC AM335x PAC at boot time

User can refer to below steps to auto-execute demo 'helloworld' at boot time in LinPAC AM335x PAC.

1. **Copy SDK demo 'examples/common/helloworld' to '/usr/sbin' directory**
2. **Create script file in '/etc/init.d' directory. Check the '/etc/init.d/pppon' or '/etc/init.d/single' file for an example.**

User can use '**vi**' command to create the script file in '/etc/init.d' directory and add below script language to the file.

```
root@ LinPAC-AM335x:~# vi /etc/init.d/hello
```

```
#!/bin/sh

### BEGIN INIT INFO
# Provides: ICP DAS
# Required-Start:
# Required-Stop:
# Should-Start:
# Should-Stop:
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Start and stop hello
# Description: hello
### END INIT INFO

helloworld > /tmp/test.log
```

3. **Type 'chmod 755 /etc/init.d/hello' command to change the access permissions for the file.**
4. **Use 'update-rc.d' command to add the script 'hello' automatically.**

```
root@ LinPAC-AM335x:~# chmod +x /etc/init.d/hello
root@ LinPAC-AM335x:~# update-rc.d hello defaults
```

5. **Reboot the LinPAC. After setting the file, the LP-22xx/52xx/8x2x/9x2x will execute binary 'helloworld' at boot time.**

5. Application for LinPAC AM335x PAC

In this chapter, ICP DAS provides extra module supported and instructions to enhance LinPAC AM335x PAC functionality and affinity.

5.1. Package management with APT

The '**apt-get**' utility is the Ubuntu package manager used to download and install software packages from local package repositories or ones located on the Internet. Package management via apt-get runs hand-in-hand with the /etc/apt/sources.list file. This page describes how to handle the packages on LinPAC AM335x PAC using apt-get and related commands.

- To install a package run the following commands:

```
# apt-get update
# apt-get install <package>
```

- To remove a package run the following commands:

```
# apt-get remove --purge <package>          // Remove the package
# apt-get autoremove --purge <package>      // Rmove the dependencies packages
```

- To search available package run the following commands:

```
# apt-cache search <package name>
```

5.2. SFTP(secure file transfer program)

The LinPAC AM335x series PAC had supported SFTP(or SCP), user can transfer the file from Windows(or Linux). For examples, using Windows Program 'WinSCP' to access the device over network, please follow below steps:

1. **Choosing the 'SFTP' or 'SCP' protocol and type IP address, default ID(root) and password(icpdas) to login. (Refer to Figure 5.2-1)**

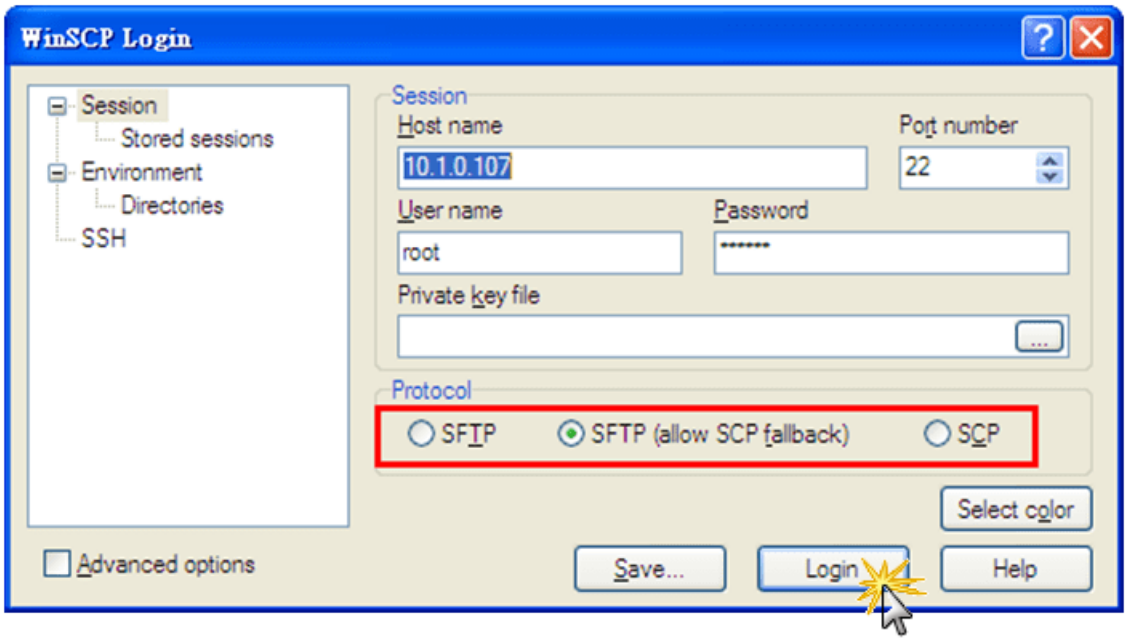


Figure 5.2-1. Using Windows Program 'WinSCP' to access the device over network

2. **Drag and drop file. (Refer to Figure 5.2-2)**

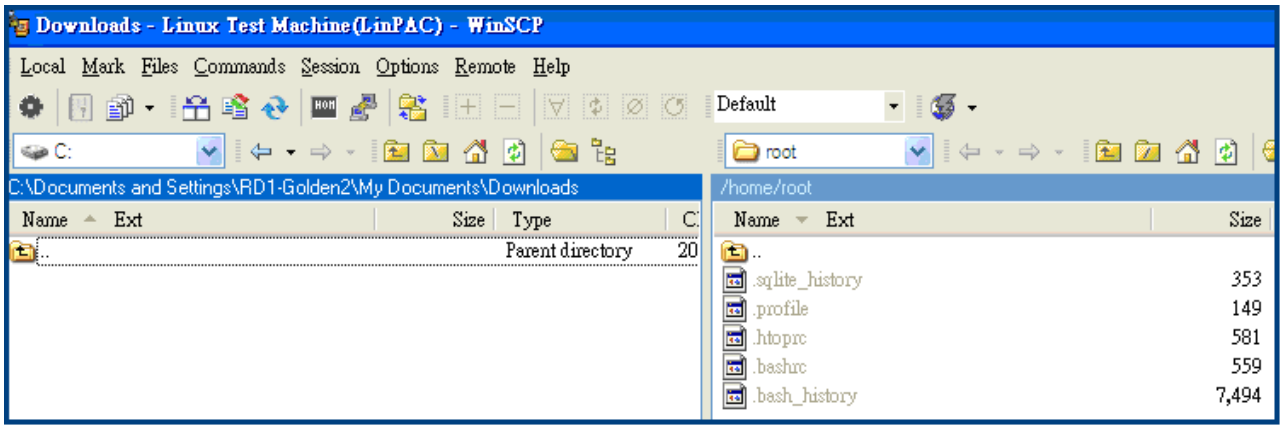
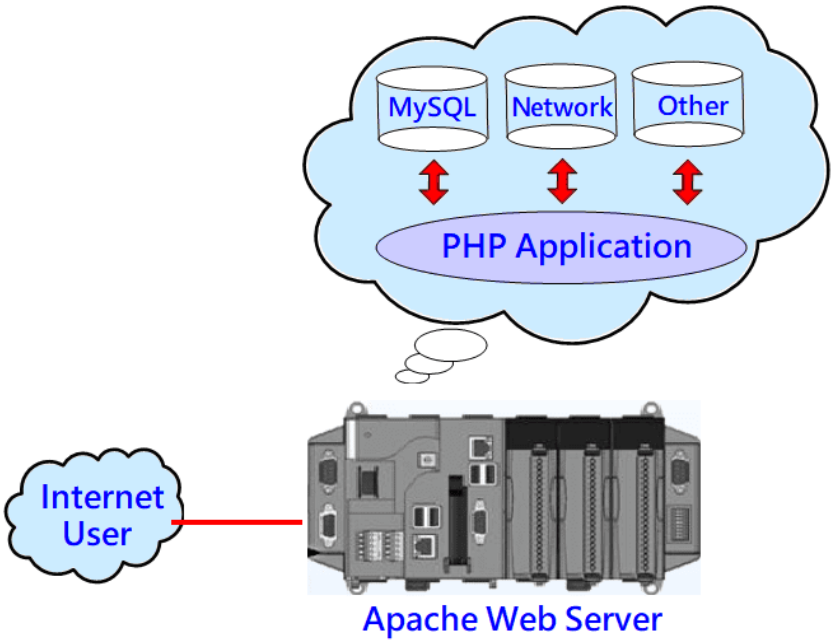


Figure 5.2-2. Drag and drop file

5.3. LAMP Server

The LAMP (Apache2 + PHP5 + MySQL) server has been built in the LinPAC AM335x PAC and it will be started automatically at boot time. As a solution stack, LAMP is suitable for building dynamic web sites and web applications. The default path of web page in the '**/var/www**' directory. If user want to change the web page's path, user can use command 'vi' to modify the configuration file '**/etc/apache2/sites-enabled/000-default**' of daemon 'apache2'. User can use the web browser and input the device IP to connect to default index page 'index.php' to get detail information.

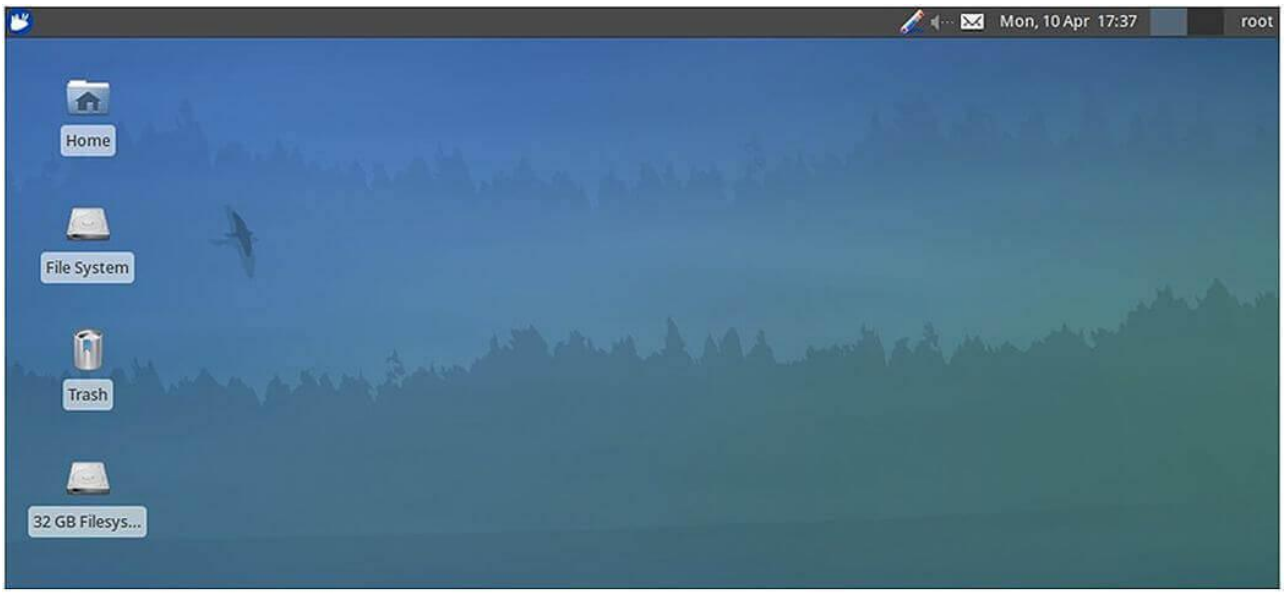


PHP Version 5.3.10-1ubuntu3.25

System	Linux LP-5231 3.2.14-rt24 #86 PREEMPT RT Tue Jun 6 09:26:29 CST 2017 armv7l
Build Date	Oct 3 2016 16:40:05
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/pdo.ini
PHP API	20090626
PHP Extension	20090626

5.4. XFCE(secure file transfer program) GUI Desktop

XFCE is a lightweight desktop environment for UNIX-like operating systems. It aims to be fast and low on system resources, while still being visually appealing and user friendly. Now the LinPAC AM335x series Linux provides the XFCE package, after user type '**root**' and password '**icpdas**' to login, the local terminal would execute the XFCE Desktop.



 E-mail: service@icpdas.com

5.5. SysVinit Support

SysVinit is a system and service manager for Linux operating systems. User can start/stop/enable /disable software service by using Linux command ‘service’ and ‘update-rc.d’. Refer to Figure 5.5-1 and 5.5-2 for more details.

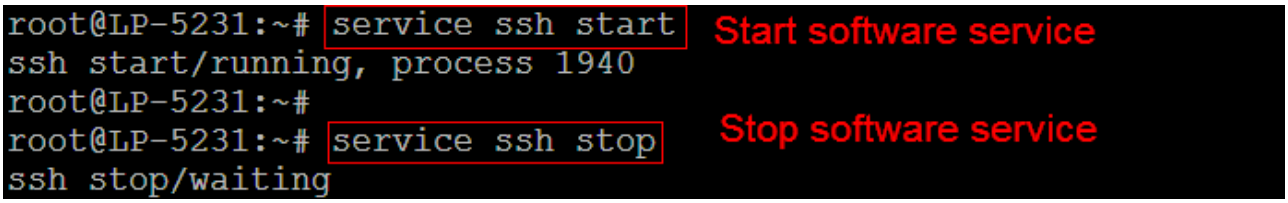


Figure 5.5-1. Start/stop software

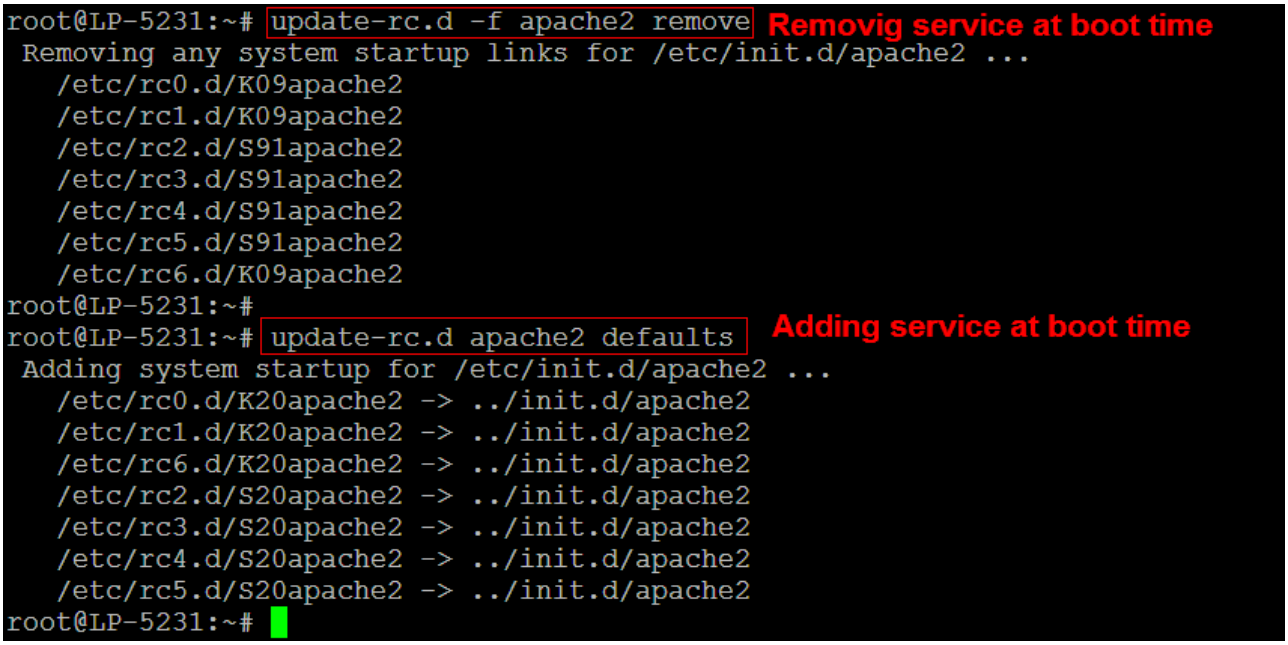


Figure 5.5-2. Enable/Disable software

5.6. Network Support

The LinPAC embedded controller already includes a variety of network functions. Following is an overview of the network functions supported in the LinPAC AM335x PAC.

5.6.1. 2G/3G/4G

LP-5231PM-3GWM support the 2G/3G system and LP-5231PM-4GE/LP-5231PM-4GC support the 2G/3G/4G system. User can use the command '**service pppon start**' to start 2G/3G/4G or the command '**service pppon stop**' to stop 2G/3G/4G. After checking for an IP address from the network provider, look for whether the '**ppp0**' network interface is active. The following is an example of operation in the LP-5231PM-3GWM. (Refer to Figure 5.6.1-1)

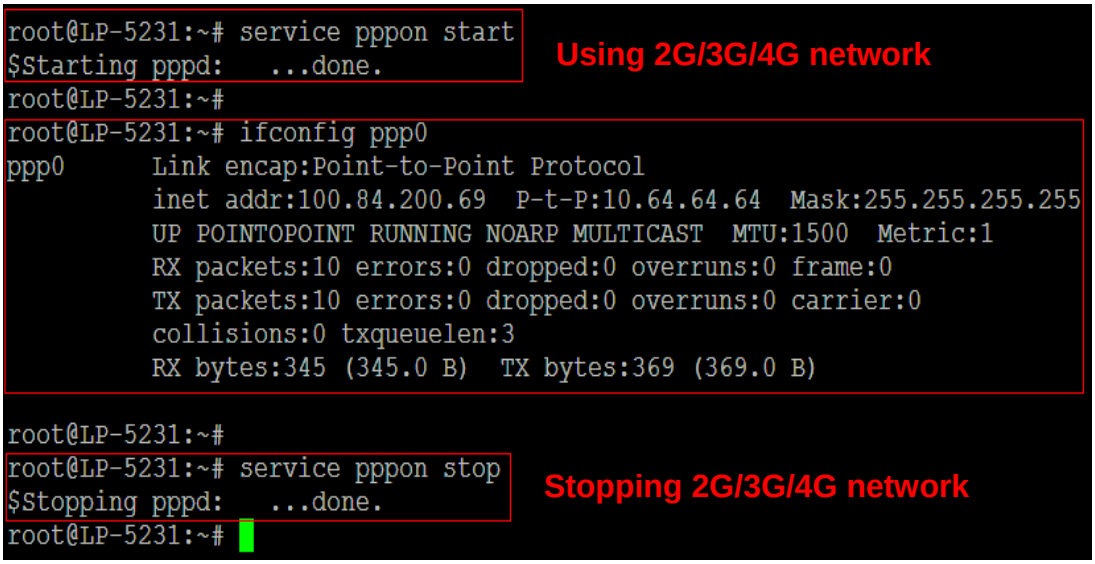


Figure 5.6.1-1. pppon service

5.6.2. SMS(Short Message Service)

The LP-5231PM-3GWA and LP-5231PM-4GE/LP-5231PM-4GC module are equipped with a 3G/4G connection application, meaning that users can download and install software for the purpose of sending short messages via a 2G, 3G, or 4G network. Four SMS tools are available for installation that allows SMS (Short Message Service) applications to be implemented on the LP-5231PM-3GWA and LP-5231PM-4GE/LP-5231PM-4GC module, each of which will be described in more detail below.

However, it is recommended that only one SMS tool be installed on the system at any one time so as to avoid software dependencies. You should ensure that any existing tools have been completely removed from the system before installing another SMS tool.

By default, the name of the 3G device is /dev/ttyUSB3, as illustrated in Figure 5.6.2-1.

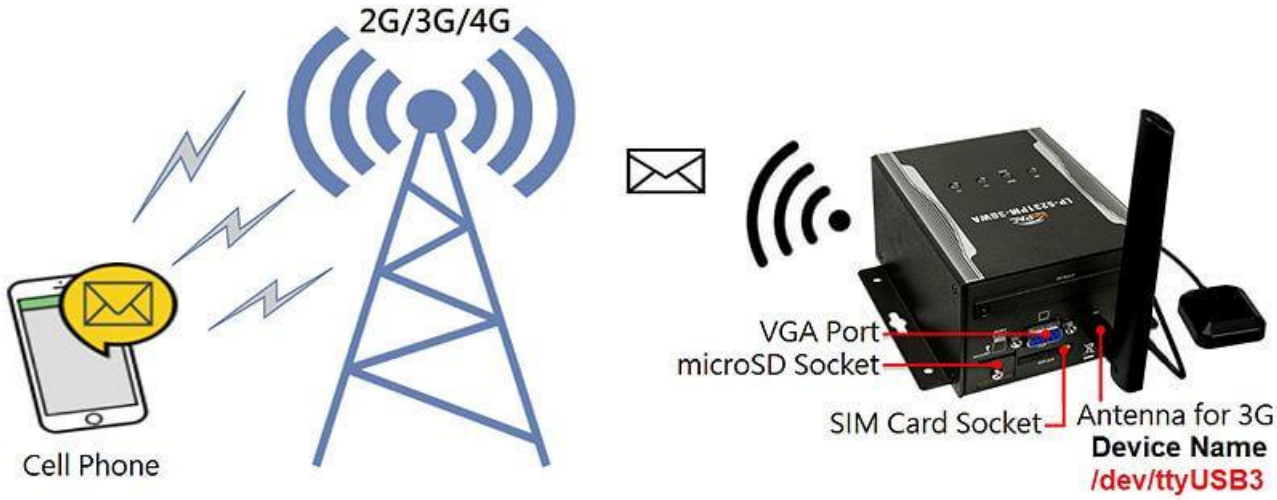


Figure 5.6.2-1. LP-5231PM-3GWA 3G device name

The following is a brief introduction to the four SMS tools available. Any one of which can be installed to send SMS messages, and they are organized as follows:

5.6.2.1. Message in English

Sending an English message, users can try the following SMS tools.

(a) gsm-utils

The gsm-utils binary package provides some simple command line programs that can be used to access a GSM mobile phone via GSM modem. To send an SMS message, using the gsm-utils package, follow the procedure described below.

Step 1: Use the 'apt-get install gsm-utils' command to install the gsm-utils package.

Step 2: Use the following command to send an SMS message.

```
#  gsmsendsms  -d  /dev/ttyUSB3  +8869XXXXXXXX  "gsm : test"
```

(b) Gnokii

Gnokii is a suite of programs for communicating with mobile phones that can be used for reading, writing, sending and receiving SMS messages. To send an SMS message, using Gnokii, follow the procedure described below.

Step 1: Use the 'apt-get install gnokii-cli' command to install the Gnokii package.

Step 2: Open the configuration file found in the '/etc/xdg/gnokii/' folder and adjust the values for the 'port' and 'serial_baudrate' parameters, as illustrated in Figure 5.6.2.1-1.

Figure 5.6.2.1-1. Modifying the Gnokii configuration file

Step 3: Use the 'mkdir gnokii' command to create a folder in the path '/root/.config/', as illustrated in Figure 5.6.2.1-2.

Step 4: Use the 'ln -s /etc/xdg/gnokii/config /root/.config/gnokii' command to create a symbolic link in the '/root/.config/gnokii/' folder, as illustrated in Figure 5.6.2.1-2.

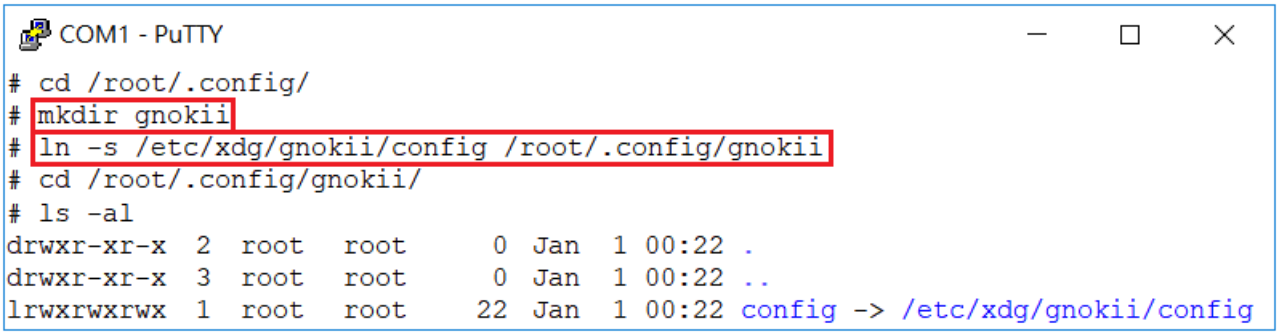


Figure 5.6.2.1-2. Creating the directory and file links

Step 5: Use the 'gnokii –identify' command to verify that the configuration functions as expected, as illustrated in Figure 5.6.2.1-3.

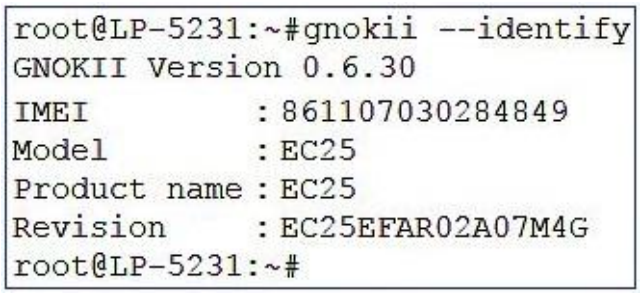


Figure 5.6.2.1-3.

Step 6: Use the following command to send an SMS message, as illustrated in Figure 5.6.2.1-4.

```
# echo "gnokii : test" | gnokii --sendsms +8869XXXXXXXX
```

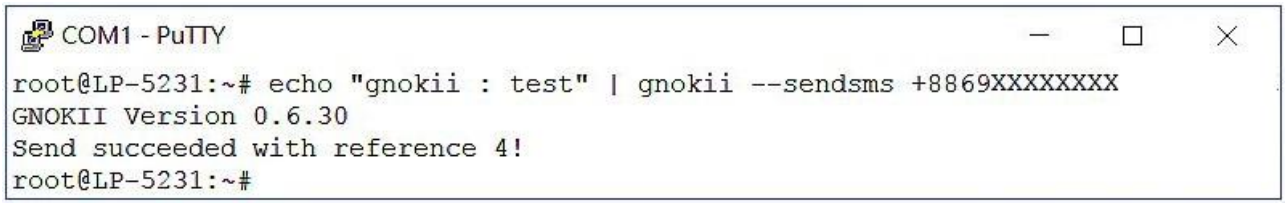


Figure 5.6.2.1-4. Sending an SMS message using Gnokii

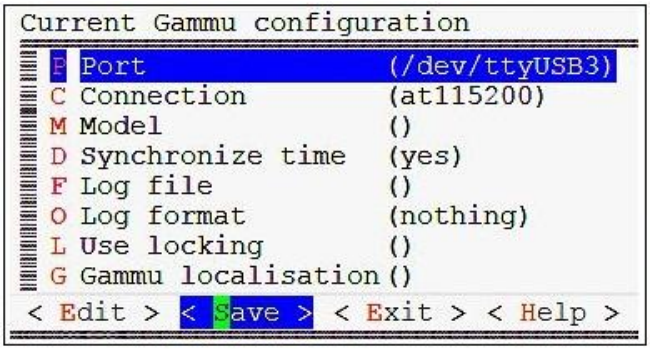
(c) Gammu

Gammu is a command line utility that can be used for reading, writing, sending and receiving SMS messages. To send an SMS message using Gammu, follow the procedure described below.

Step 1: Use the 'apt-get install gammu' command to install the Gammu package.

Step 2: Use the 'gammu-config' command to modify the configuration file.

The Gammu package includes an easy to use tool gammu-config, which allows the user to create and configure the most important options in the Gammu configuration file, as illustrated in Figure 5.6.2.1-5.



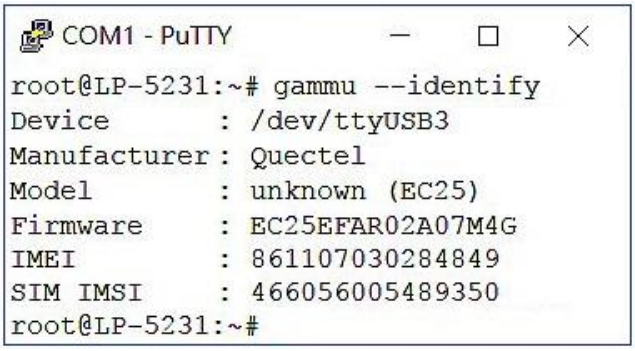


Figure 5.6.2.1-5.

Figure 5.6.2.1-6.

Step 3: Use the 'gammu –identify' command to verify that the configuration is functioning as expected, as illustrated in Figure 5.6.2.1-6.

Step 4: Use the following command to send an SMS message, as illustrated in Figure 5.6.2.1-7.

```
# gammu sendsms TEXT +8869XXXXXXXX -text "gammu : test"
```

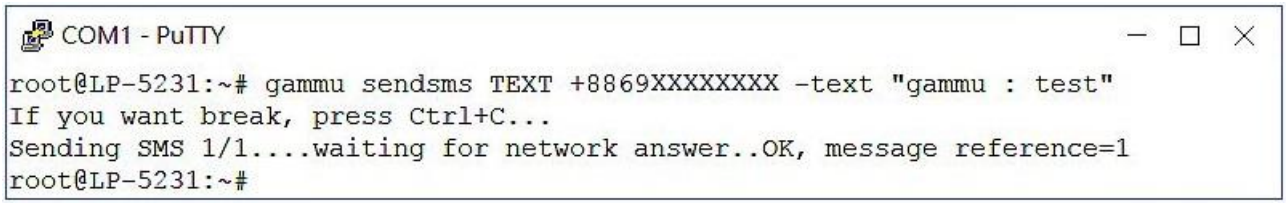


Figure 5.6.2.1-7. Sending an SMS message using Gammu

(d) SMS Server Tools

The SMS Server tools package is an SMS Gateway application which can be used to send and receive SMS messages via GSM modems and mobile phones. To send an SMS message via the SMS Server, follow the procedure described below.

Step 1: Download the SMS Server Tools package.

The latest version of the SMS Server Tools package can be downloaded from:

http://smstools3.kekekasvi.com/index.php?p=packages

Step 2: Compile and install the SMS Server tools package.

Refer to the following website for detailed information about how to compile and install the SMS Server tools package:

http://smstools3.kekekasvi.com/index.php?p=compiling

Step 3: Use the Linux 'vi' command to modify the '/etc/smsd.conf' file and change the 'device' parameters, as illustrated in Figure 5.6.2.1-8.

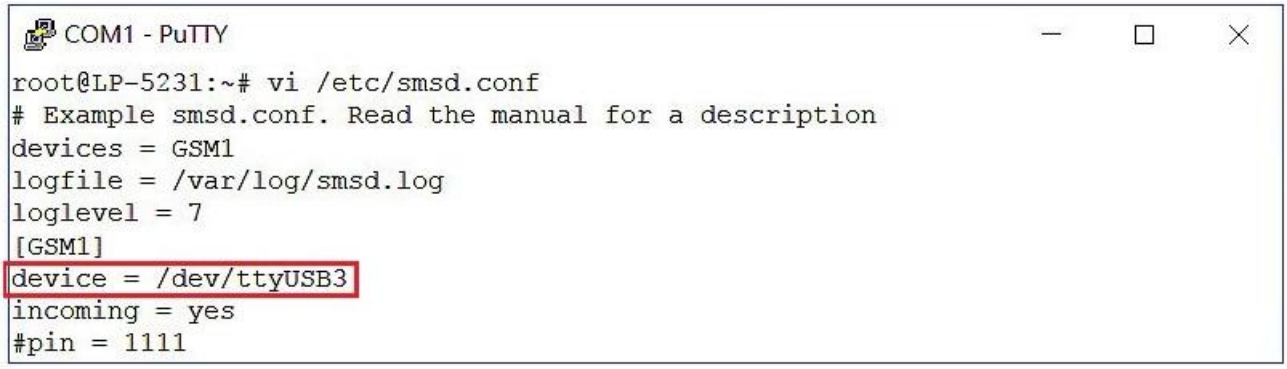


Figure 5.6.2.1-8. Modifying the smsd configuration file

Step 4: Use the '/etc/init.d/sms3 start' command to start the SMSD service in the background, as illustrated in Figure 5.6.2.1-9.

Step 5: Use the following command to send an SMS message, as illustrated in Figure 5.6.2.1-9.

```
# sendsms 09XXXXXXXX 'smstools : test'
```

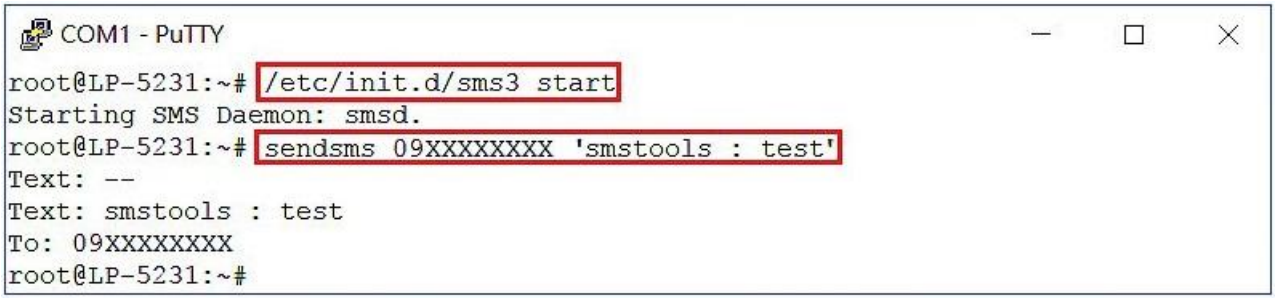


Figure 5.6.2.1-9. Sending an SMS message using SMS Server tools

5.6.2.2. Messages in Traditional Chinese

To send a message in Traditional Chinese, first install the '**SMS Server tools**' package. For more detailed information regarding the installation of this package, refer to Section (d) of Chapter 5.6.2.1. Before you can send a message in Traditional Chinese the locale configuration for the system must be changed to **UTF-8** encoding. Refer to the procedure described below for details of how to accomplish this.

Step 1: Use the following command to create a new system locale and write the 'LC_CTYPE=zh_TW.utf8' environment variables to the '/etc/default/locale' file, as illustrated in Figure 5.6.2.2-1.

(Note: 'zh_TW.utf8' is the UTF-8 code for Traditional Chinese.)

```
# echo LC_CTYPE=zh_TW.utf8 > /etc/default/locale
```

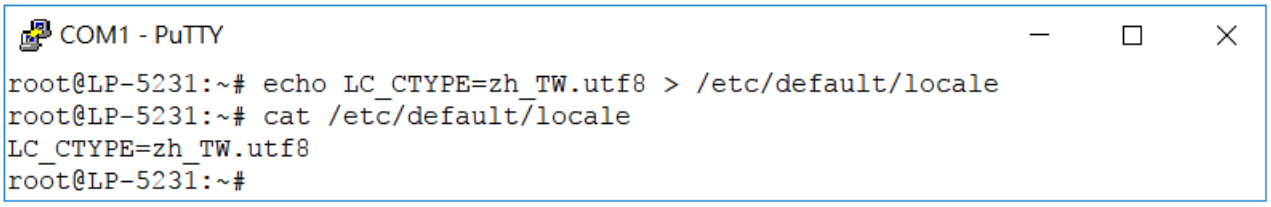


Figure 5.6.2.2-1. Setting the system locale

Step 2: To apply the new configuration, type the 'reboot' command to restart the LinPAC.

Step 3: Use the '/etc/init.d/sms3 start' command to start the SMSD service in the background.

Step 4: Send a message to a mobile device using Traditional Chinese. The results of the execution are as illustrated in Figure 5.6.2.2-2.

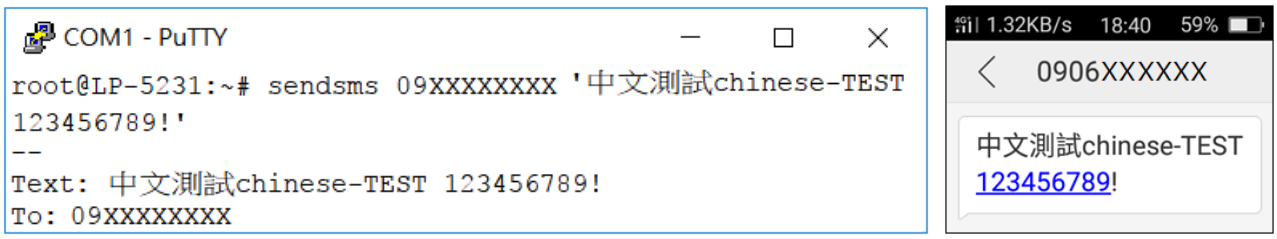


Figure 5.6.2.2-2. Sending an SMS message using Traditional Chinese

6. LinPAC AM335x PAC System Settings

The following is a guide to easily configuration the LinPAC AM335x PAC.

6.1. Using a microSD Card

When using a microSD card, be sure to pay attention to the following items:

1. Unmount the microSD card before removing it.
2. Do not power off or reboot the LinPAC AM335x PAC while data is being written to or read from the microSD card.
3. The microSD card must be formatted with the VFAT/EXT2/EXT3 file system.

To mount a microSD storage devices follow the procedure described below:

(1) Type '**cat /proc/diskstats**' to find the device name of microSD card. (Refer to Figure 6.1-1)

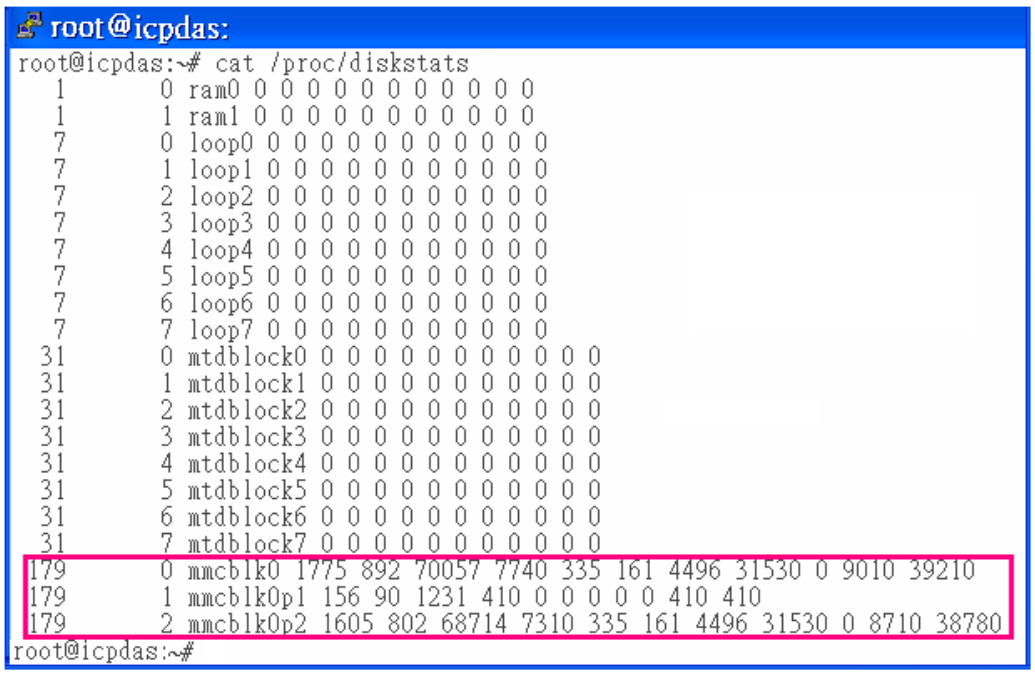


Figure 6.1-1. Find the device name of microSD card

(2) Type '**mkdir /mnt/had**' to create a directory named '**had**'. (Refer to Figure 6.1-2)

(3) Files contained on a mounted microSD card can be accessed from the **/mnt/hda** directory.

(Refer to Figure 6.1-2)

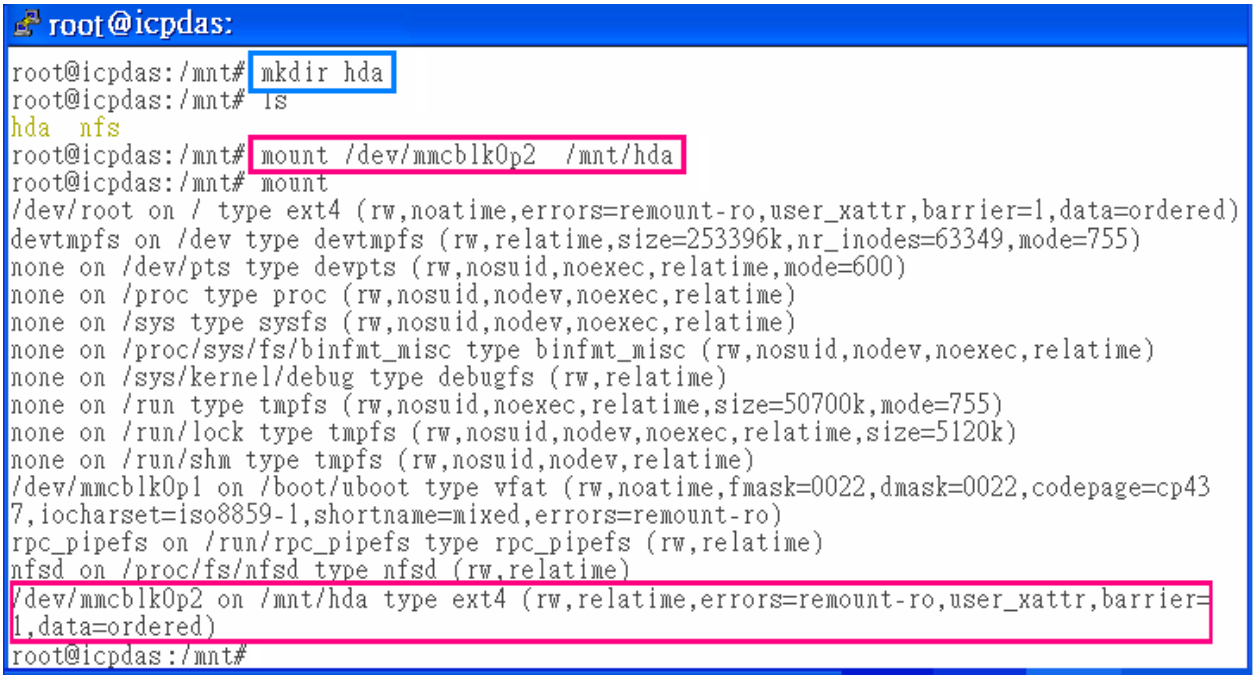


Figure 6.1-2. Create and mount a directory named 'had'

6.1.1. Mounting a microSD Card

To use a microSD card, insert the microSD card into the socket on the LinPAC AM335x PAC, and it will be automatically mounted when the LinPAC AM335x PAC is booted. The files of SD card can then be accessed from the **/boot/uboot** directory. (Refer to Figure 6.1.1-1)

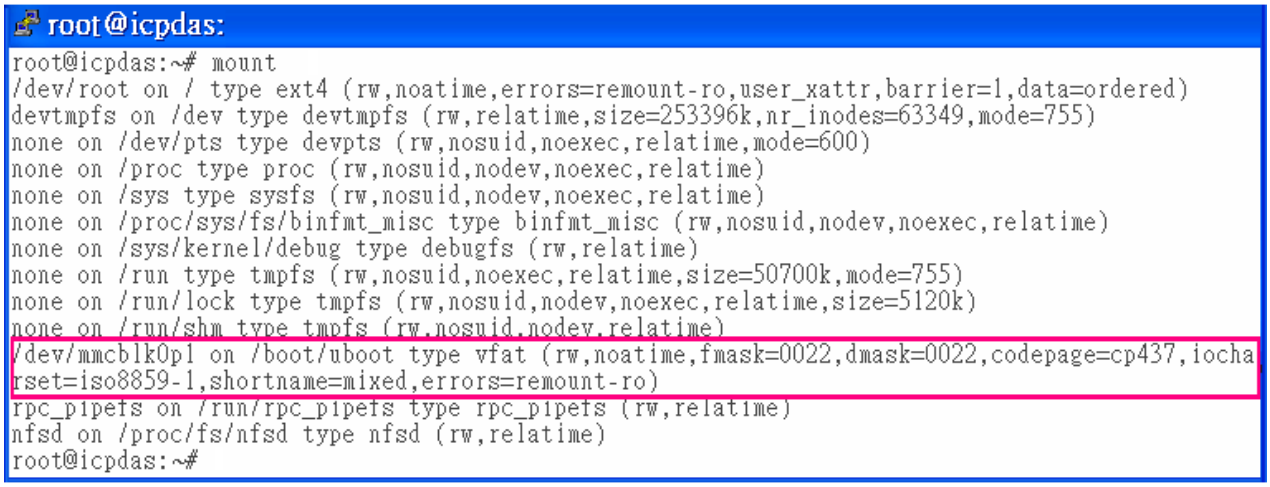


Figure 6.1.1-1. The files of SD card can be accessed from the /boot/uboot directory

6.1.2. Unmounting the microSD Card

Before removing the microSD card from the LinPAC AM335x PAC, unmount the card by entering the command:

```
#   umount   /boot/uboot
```

The microSD card can then be safely removed to prevent damage to the card. (Refer to Figure 6.1.2-1)

Figure 6.1.2-1. Remove the microSD card

6.1.3. Scanning and repairing a microSD Card

After the LinPAC AM335x PAC is booted, the microSD card will be named '**/dev/mmcblk0p1**'. It is recommended that the microSD card is unmounted first before attempting to perform a scan or repair.

- **blockdev**: this command is used to call block device ioctls from the command line.

Parameter	Description	Example
--report	print a report for device	blockdev --report /dev/mmcblk0p1
-v --getra --getbz	get readhead and blocksize	blockdev -v --getra –getbz /dev/mmcblk0p1

- **fsck.minix:** this command is used to perform a consistency check for the Linux MINIX filesystem.

Parameter	Description	Example
-r	performs interactive repairs	fsck.minix -r /dev/mmcblk0p1
-s	outputs super-block information	fsck.minix -s /dev/mmcblk0p1

- **fsck.vfat**: this command is used to check and repair MS-DOS file systems.

Parameter	Description	Example
-a	automatically repair the file system	fsck.vfat -a /dev/mmcblk0p1
-l	list path names of files being processed	fsck.vfat -l /dev/mmcblk0p1

- **mkfs:** this command is used to build a Linux file system on a device, usually a hard disk partition.

Parameter	Description	Example
-t	specifies the type of file system to be built	mkfs -t vfat /dev/mmcblk0p1
-c	check the device for bad blocks before building the file system	mkfs -c vfat /dev/mmcblk0p1

❑ **mkfs.minix**: this command is used to make a MINIX filesystem

Parameter	Description	Example
	create a Linux MINIX file-system	mkfs.minix /dev/mmcblk0p1
-c	check the device for bad blocks before building the file system	mkfs.minix -c /dev/mmcblk0p1

❑ **mkfs.vfat**: this command is used to make an MS-DOS filesystem

Parameter	Description	Example
-A	use Atari variation of the MS-DOS filesystem	mkfs.vfat -A /dev/mmcblk0p1
-v	verbose execution	mkfs.vfat -v /dev/mmcblk0p1

6.2. Using a USB Storage Device

USB storage devices are not automatically mounted to the LinPAC AM335x PAC, set it must be manually mounted before attempting to access the USB storage device. (Refer to Figure 6.2-1)

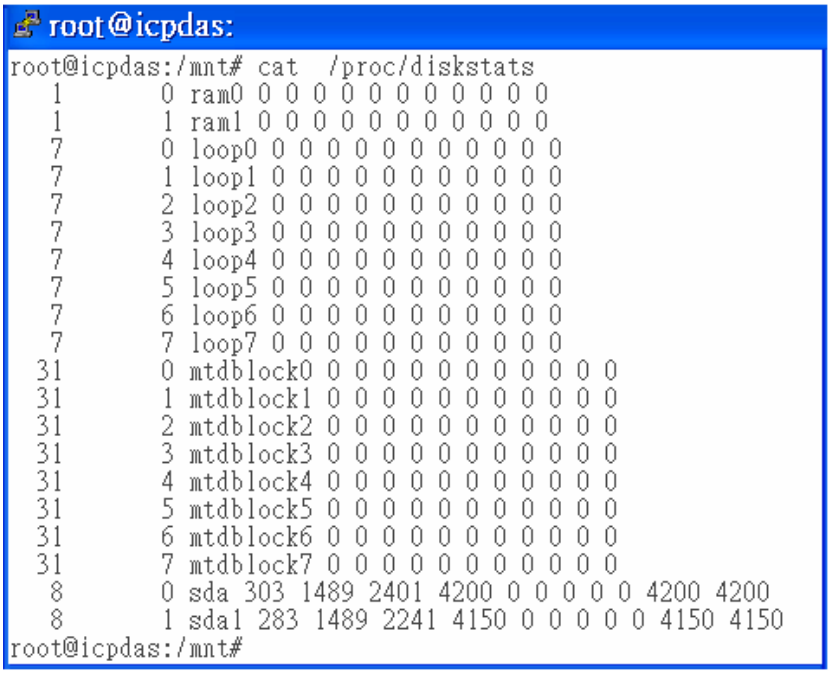


Figure 6.2-1. Checking that the USB storage device is on disk

6.2.1. Mounting a USB Storage Device

To mount a USB storage devices follow the procedure described below:

(1) Type '**mkdir /mnt/usb**' to create a directory named 'usb'.

(2) Type '**mount /dev/sda1 /mnt/usb**' to mount the USB storage device to the usb directory and then type '**ls /mnt/usb**' to view the contents of the USB storage device. (Refer to Figure 6.2.1-1)

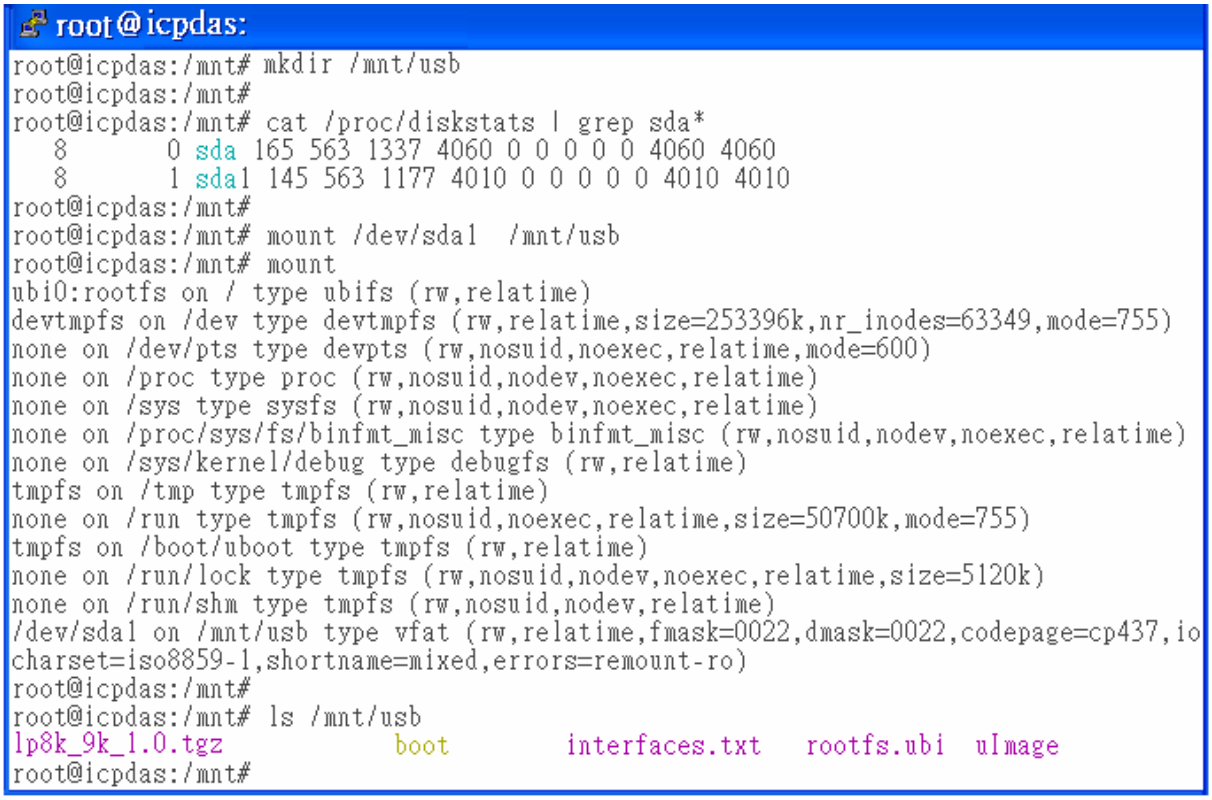


Figure 6.2.1-1. Mounting a USB Storage Device

6.2.2. Unmounting the USB Storage Device

Before removing the USB storage device from the LinPAC AM335x PAC, the device must be unmounted to prevent any damage to the device. To unmount the device, type the '**umount/mnt/ usb**' command and then remove the USB storage device.

6.3. WDT

6.3.1. WDT for LP-8x2x and LP-9x2x

Use the '**wdt**' command to enable and configure the WDT. There are three steps to this process, which are described below.

(1) Enable the WDT.

❒ Enable the WDT. The default response time is 10 seconds.

```
# wdt -e
```

❒ Enable the WDT and set the response time (the response time ranges between 2 and 510 seconds).

```
# wdt -e -s4                                    // Set the response time to 4 seconds
```

(2) Refresh the WDT source.

```
# wdt -r
```

(3) Disable the WDT.

```
# wdt -d
```

6.3.2. WDT for LP-22xx and LP-52xx

To **Enable WDT** working status, there are two steps to this process, which are described below.

(1) Refresh WDT source.

```
# echo timer > /sys/class/leds/beaglebone::wdt/trigger  //Refresh WDT
```

(2) Enable WDT.

```
# echo 0 > /proc/hmistat/radiopower                     //Enable WDT
```

To **Disable WDT** working status, there are two steps to this process, which are described below.

(1) Disable WDT.

```
# echo 1 > /proc/hmistat/radiopower                     //Disable WDT
```

(2) Clear WDT refresh source.

```
# echo none > /sys/class/leds/beaglebone::wdt/trigger   //Clear WDT Refresh Source
```


E-mail: service@icpdas.com

6.4. EEPROM

To **Enable EEPROM** working status, there are four steps to this process, which are described below.

(1) Startup EEPROM GPIO function.

```
#  echo  64  >  /sys/class/gpio/export
```

(2) The EEPROM is write protected by default, the user needs to modify default value of EEPROM.

```
#  echo  out  >  /sys/class/gpio/gpio64/direction
```

(3) Change to writable of EEPROM.

```
#  echo  0  >  /sys/class/gpio/gpio64/value
```

(4) To write a data to EEPROM.

```
#  echo  hello  >  /sys/bus/i2c/devices/1-0050/eeprom
```

More detailed information, please refer to the demo code:

C:\cygwin\LinPAC_am335x_SDK\examples\common\eeprom.c or

root@LinuxPC-ICPDAS:/icpdas/linpac_am335x_sdk/i8k/examples/common/eeprom.c

6.5. LED

Following is the control method of the LinPAC AM335x series PAC LED indicators.

6.5.1. LED Indicators for LP-22xx series

The LP-22xx series modules have 6 LED indicators, as illustrated in Figure 6.5.1-1.

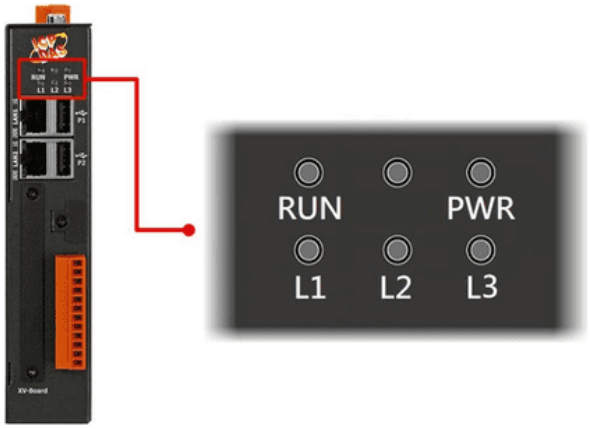


Figure 6.5.1-1. LED indicators for LP-22xx series

LED Indicator	Color	Meaning
PWR	**Red**	Power is on.
RUN	**Green**	Power on and OS is running.
L1	**Green**	User programmable LED indicator.
L2	**Orange**	
L3	**Red**	

The user can use 'led' command to control the LP-22xx LED indicator.

Parameter:

LED Indicator	L1		L2		L3	
Status	**ON**	**OFF**	**ON**	**OFF**	**ON**	**OFF**
Parameter	**-g1**	**-g0**	**-y1**	**-y0**	**-r1**	**-r0**

6.5.2. LED Indicators for LP-52xx series

The LP-52xx series modules have 3/4 LED indicators, as illustrated in Figure 6.5.2-1.

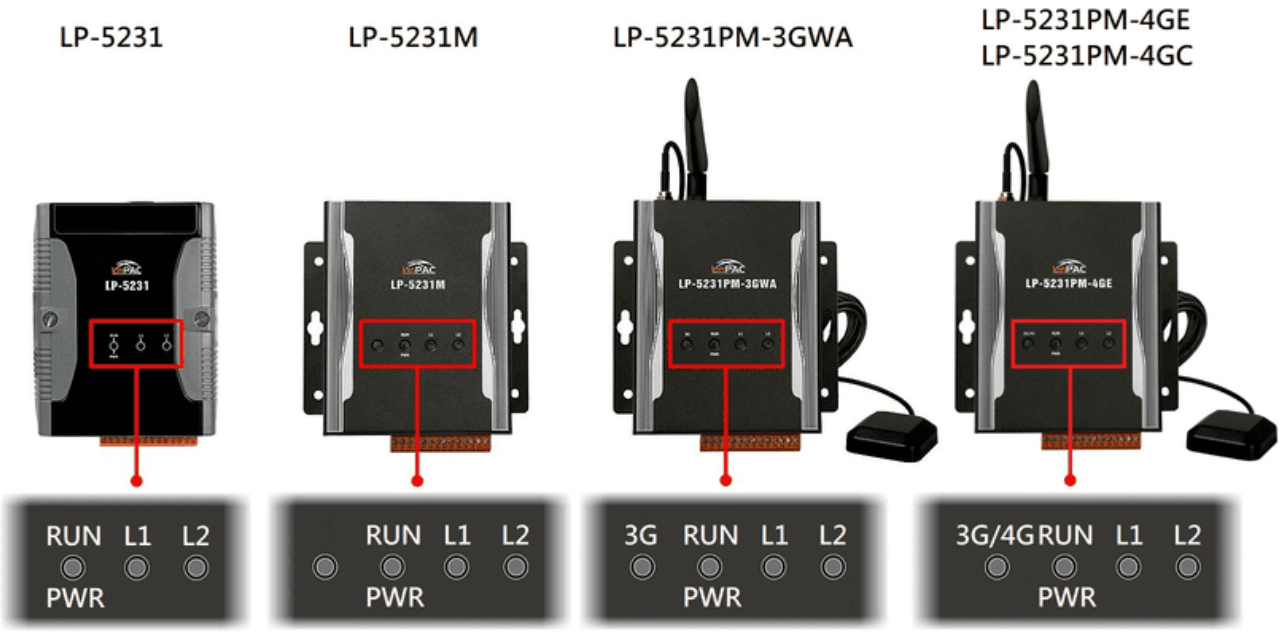


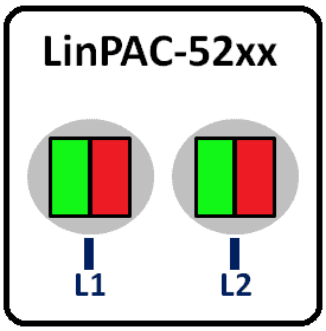
Figure 6.5.2-1. LED indicators for LP-52xx series

LED Indicator	Color	Meaning
3G/4G	**Green**	The 3G/4G LED indicates that the antenna is connected to 3G/4G network.
PWR	**Red**	Power is on.
RUN	**Green**	OS is running.
L1	**Green/Red**	User programmable LED indicator.
L2	**Green/Red**	

The led_52xx.c demo program illustrates control method of the LP-52xx LED indicator, user can be found demo code in the path C:\cygwin\LinPAC_am335x_SDK\examples\common\led-52xx.c or root@LinuxPC-ICPDAS:/icpdas/linpac_am335x_sdk/i8k/examples/common/led-52xx.c

Parameter:

LED Indicator		Parameter
L1		-l1
L2		-l2
LED color		**Parameter**
Green	ON	-g1
	OFF	-g0
Red	ON	-r1
	OFF	-r0



6.5.3. LED Indicators for LP-8x2x series

The LP-8x2x series modules have 2 LED indicators, as illustrated in Figure 6.5.3-1.

Figure 6.5.3-1. LED indicators for LP-8x2x series

LED Indicator	Color	Meaning
PWR	**Green**	Power 1 is on.
RUN	**Red**	OS is running.
	Red	User programmable LED indicator.

The led_8x2x.c demo program illustrates control method of the LP-8x2x LED indicator, user can be found demo code in the path C:\cygwin\LinPAC_am335x_SDK\examples\common\led-8x2x.c or root@LinuxPC-ICPDAS:/icpdas/linpac_am335x_sdk/i8k/examples/common/led-8x2x.c

Parameter:

[RUN LED] **1** : Turn on the RUN LED

0 : Turn off the RUN LED

6.5.4. LED Indicators for LP-9x2x series

The LP-9x2x series modules have 5 LED indicators, as illustrated in Figure 6.5.4-1.

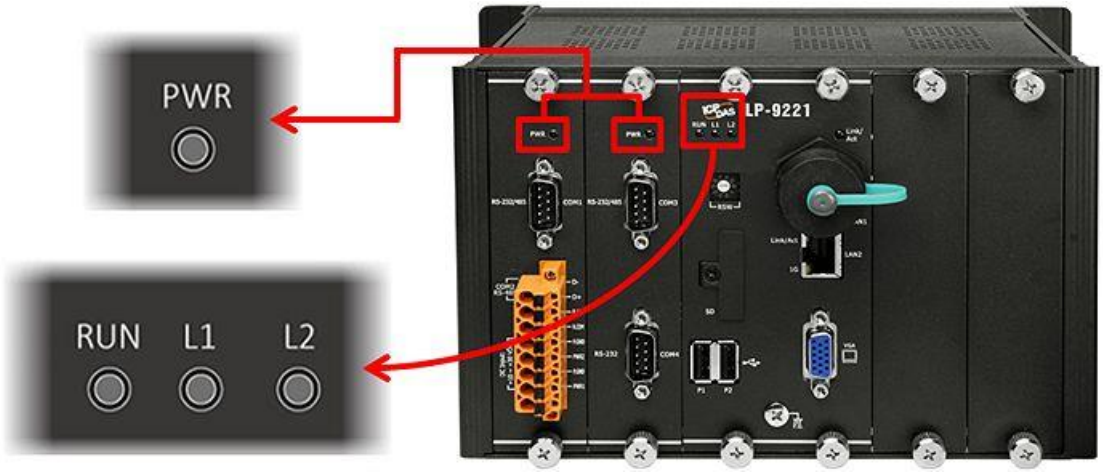


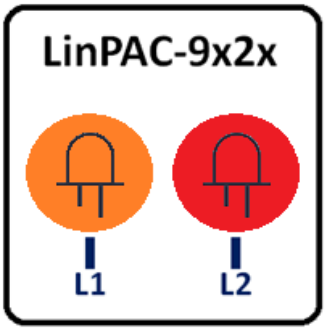
Figure 6.5.4-1. LED indicators for LP-9x2x series

LED Indicator	Color	Meaning
PWR	**Red**	Power is on.
RUN	**Green**	OS is running.
L1	**Orange**	User programmable LED indicator.
L2	**Red**	

The led_9x2x.c demo program illustrates control method of the LP-9x2x LED indicator, user can be found demo code in the path C:\cygwin\LinPAC_am335x_SDK\examples\common\led-9x2x.c or root@LinuxPC-ICPDAS:/icpdas/linpac_am335x_sdk/i8k/examples/common/led-9x2x.c

Parameter:

Options	LED Status	
	L1	L2
1	**ON**	**OFF**
2	**OFF**	**ON**
3	**ON**	**ON**
4	**OFF**	**OFF**



7. Additional Support

This chapter provides additional information related to the modules supported, together with instructions that can be used to enhance the functionality and efficiency of the LinPAC AM335x PAC module.

7.1. Support for N-Port Modules

N-port communication modules provide **two** or **four serial ports** and can be inserted into the slot of an LP-8x2x/9x2x embedded controller. In this way, additional serial ports can be used on the LP-8x2x/9x2x embedded controller, meaning that the maximum number of serial ports available on the LP-8x2x/9x2x will be expanded to **thirty-four**.

■ LP-8x2x

The LP-8x2x embedded controller is a multi-tasking uint, meaning that all the serial ports can be controlled simultaneously. **The number** of **each serial port on the** I-8114W and I-8112iW modules are presented in Figures 7.1-1 and 7.1-2. The information illustrated in Figure 7.1-5 is for the LP-8121 only and is **fixed** based on their slot position.

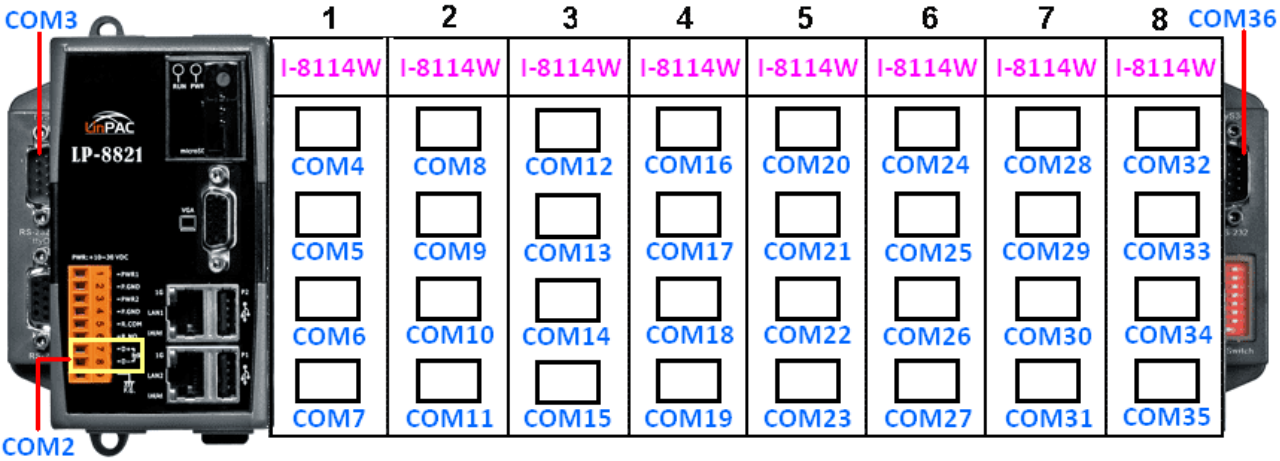


Figure 7.1-1. The number of each serial port on the I-8114W modules are presented

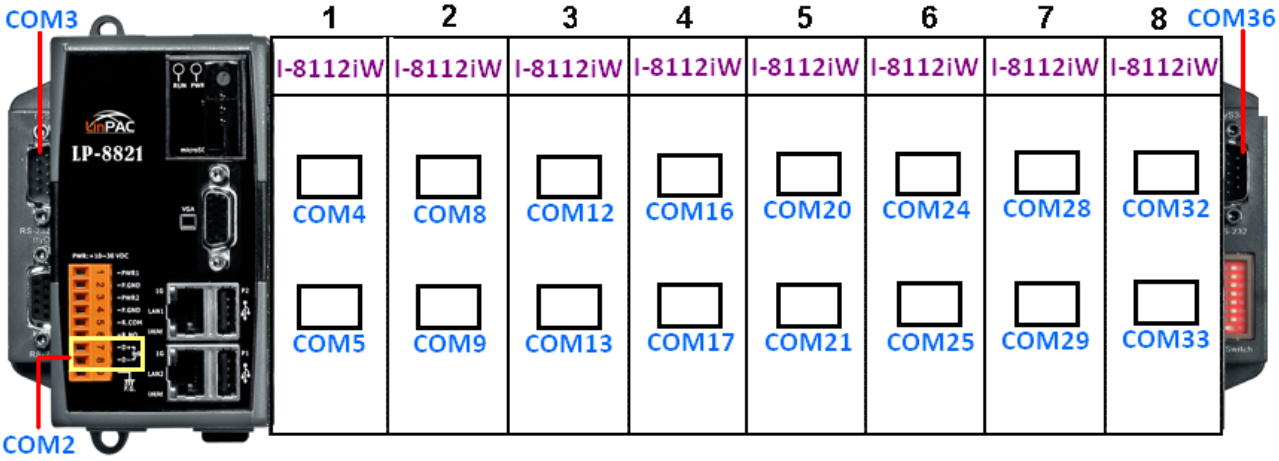


Figure 7.1-2. The number of each serial port on the I-8112iW modules are presented

Figures 7.1-3 and 7.1-4 illustrated the serial port numbers that correspond to the **device name** on the LP-8x2x.

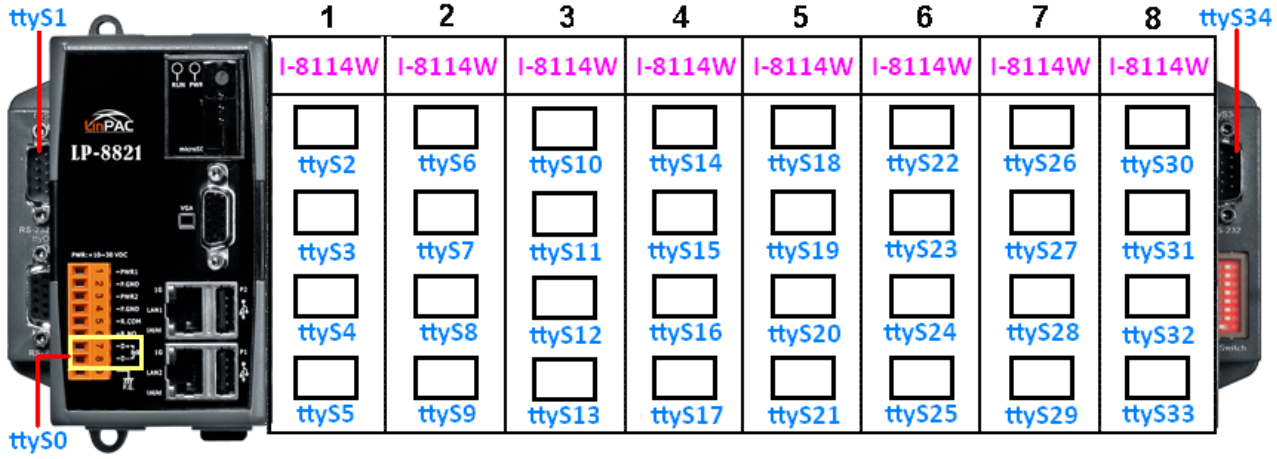


Figure 7.1-3. Device node of I-8114W

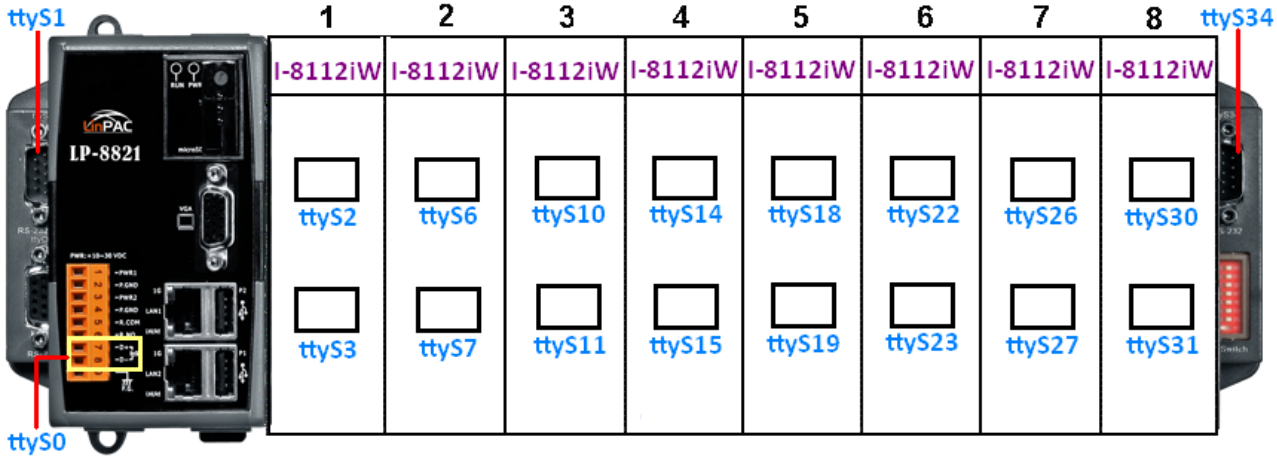


Figure 7.1-4. Device node of I-8112iW

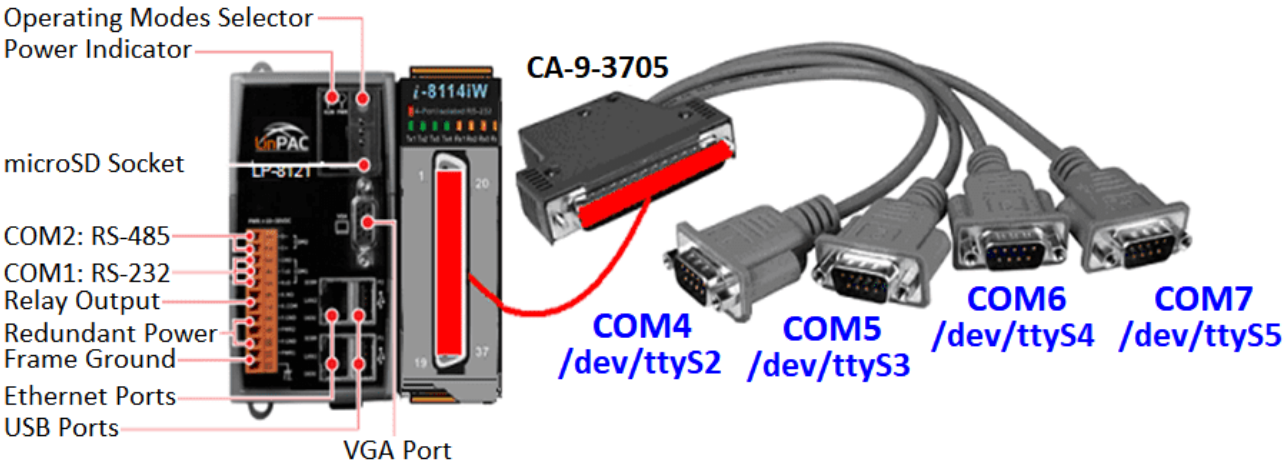


Figure 7.1-5. Device slot information for the LP-8121 only

The user can type 'getlist' to check the module is installed or not, as illustrated in Figure 7.1-6.

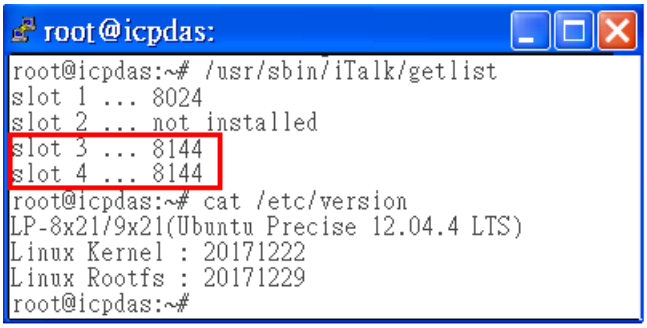


Figure 7.1-6. Lists the names of all modules inserted

Selection guide for Hight-profile I-8K modules:

Module	Interface	Ports	Max. Channels	Max. Speed (Kbps)	Isolation (V)
I-8112iW	RS-232	2	16	115.2	2500
I-8114W	RS-232	4	32	115.2	–
I-8114iW	RS-232	4	32	115.2	2500
I-8142iW	RS-422/RS-485	2	16	115.2	2500
I-8144iW	RS-422/RS-485	4	32	115.2	2500

For more information relating to these modules, refer to:

http://www.icpdas.com/products/Remote_IO/i-8ke/selection_rs232_i8k.htm

LP-9x2x

The LP-9x2x embedded controller is a multi-tasking uint, meaning that all the serial ports can be controlled simultaneously. **The number** of **each serial port on the** I-9114 and I-9144 modules are presented in Figure 7.1-7 and is **fixed** based on their slot position.

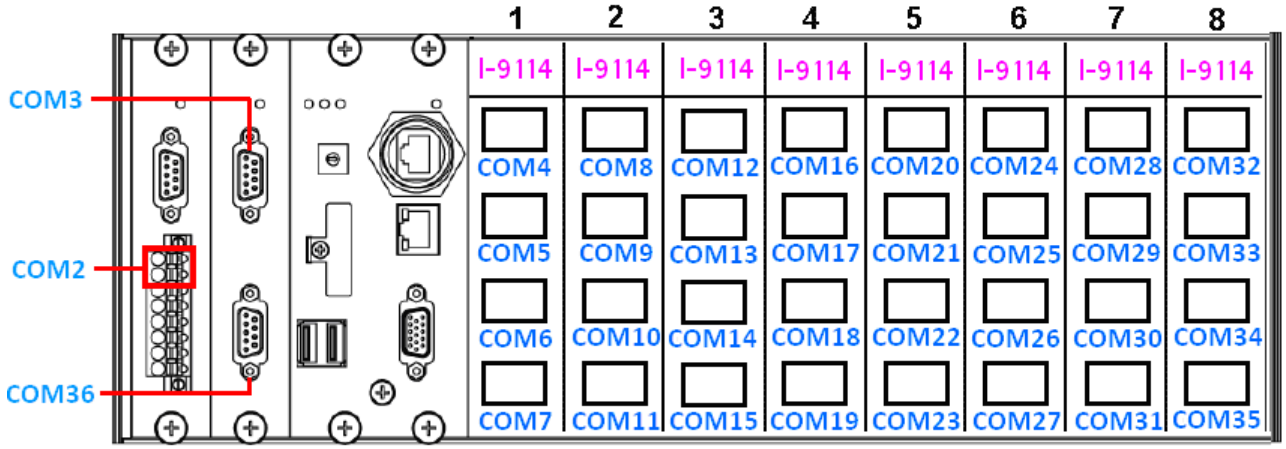


Figure 7.1-7. The number of each serial port on the I-9114 modules are presented

Figure 7.1-8 illustrated the serial port numbers that correspond to the **device name** on the LP-9x2x.

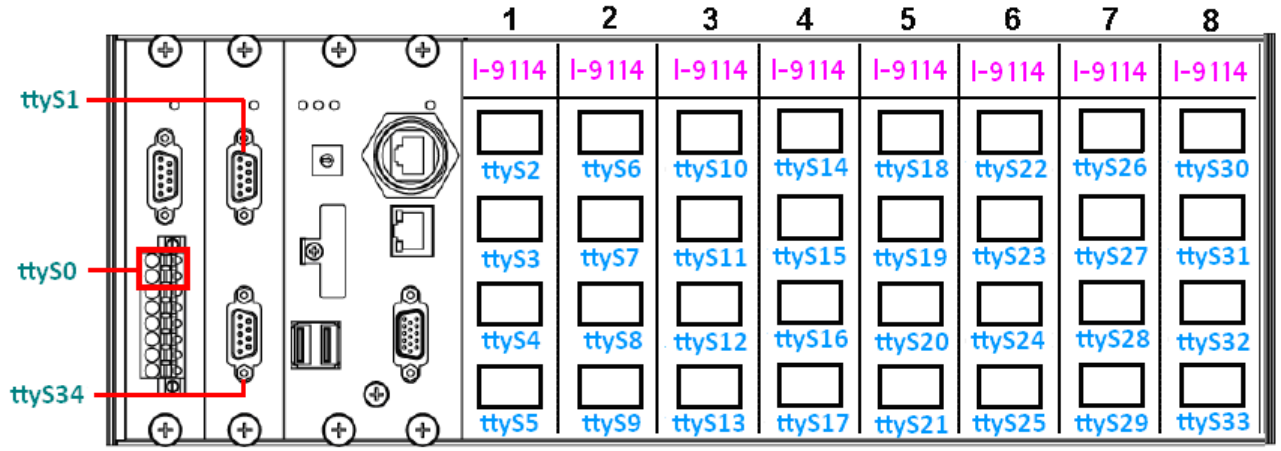


Figure 7.1-8. Device node of I-9114

Check the module is installed or not, and then check the status of the serial port, as illustrated in Figure 7.1-9.

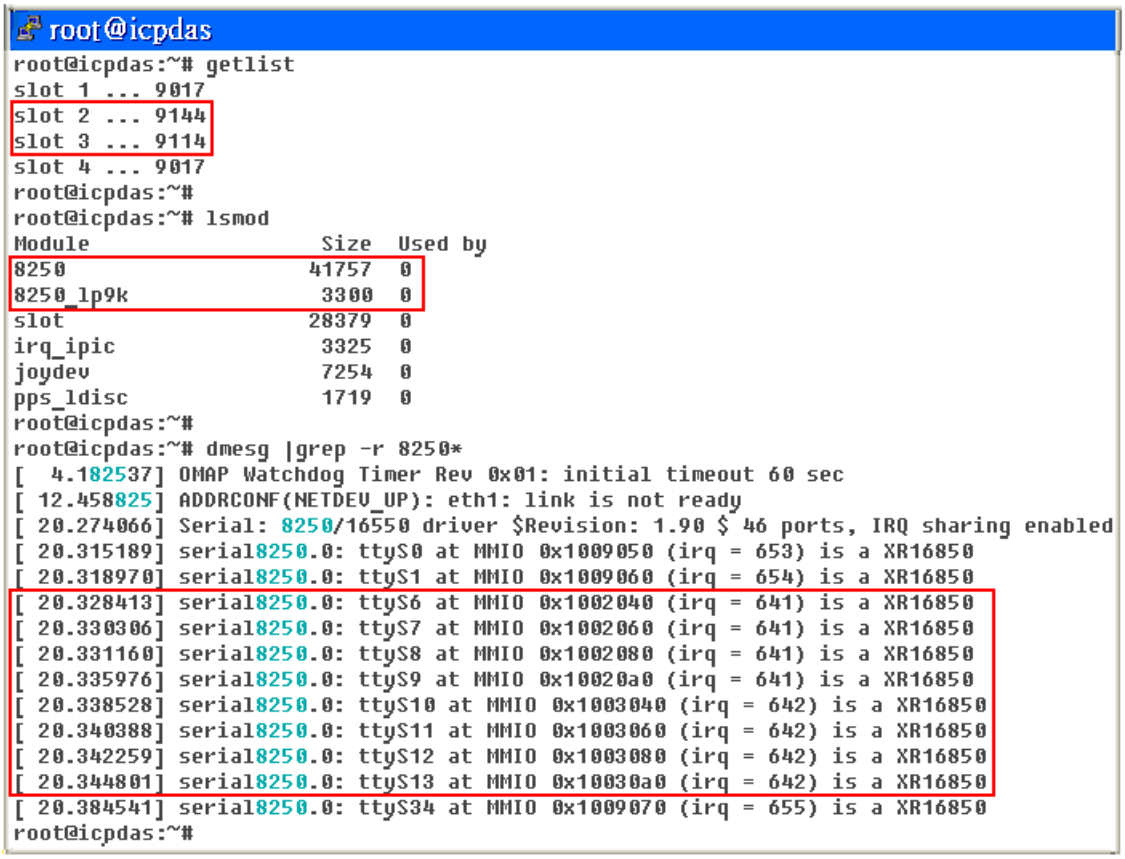


Figure 7.1-9. Lists the names of all modules inserted

Selection guide for Hight-profile I-9K modules:

Module	Interface	Ports	Max. Speed (Kbps)	Isolation (Vrms)
I-9114	RS-232	4	115.2	2500
I-9144	RS-422/RS-485	4	115.2	2500

For more information relating to these modules, refer to:

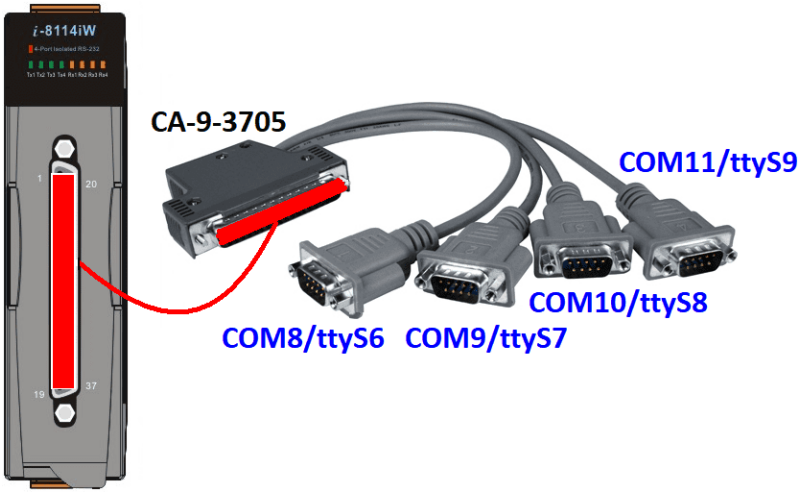
http://www.icpdas.com/root/product/solutions/remote_io/i-9k_i-97k/i-9k_i-97k_selection.html

7.1.1. Application for N-Port Module

The **i7kdio_8114.c** demo program illustrates how to use an I-8114W module that is inserted into an LP-8x21 embedded controller. In this demo program, the I-7044 module (8 DO and 4 DI channels) is controlled through the second serial port on the I-8114W module that is inserted into the slot 2 on the LP-8x21, which, in turn, is connected to an RS-485 network. The address of the I-7044 module is 02 and the baud rate is 115200 bps. Figure 7.1.1-1 provides an illustration of the control diagram.

Figure 7.1.1-1. Connection diagram of the device

The result of executing this demo program is that the state of the DO channels can be controlled, and the program returns the state of the DI channels. The source code for the demo program is as follows:




```
#include<stdio.h>
#include<stdlib.h>
#include "msw.h"

char szSend[80], szReceive[80], ans;
WORD wBuf[12];
float fBuf[12];
int main()

{
    int wRetVal, j=0;
```

```
char i[10];

// Check Open_Com9 on the I-8114W
wRetVal = Open_Com(COM9, 115200, Data8Bit, NonParity, OneStopBit);

if (wRetVal > 0) {
      printf("Failed to open port. \n");
      return (-1);
}

//   *****   7044 DO & DI Parameters *******
wBuf[0] = 9;                    // COM Port
wBuf[1] = 0x02;                 // Address
wBuf[2] = 0x7044;               // ID
wBuf[3] = 0;                    // Checksum disable
wBuf[4] = 100;                  // Timeout, 100 milliseconds
wBuf[6] = 0;                    // Debug string

// 7044 DO
while(j!=113) {
      printf("Enter the DO value, or press 'q' to quit -> ");
      scanf("%s",i);

      if (i[0]=='q') {
            wBuf[5] = 0;              // All DO Channels OFF
            wRetVal = DigitalOut(wBuf, fBuf, szSend, szReceive);
            break;
      }

      j=atoi(i);
      if (j>=0 & j<=255)
            wBuf[5] = j;              // DO Channels ON
      else if (j>255)
            wBuf[5] = 255;

      wRetVal = DigitalOut(wBuf, fBuf, szSend, szReceive);
      if (wRetVal)                  // There was an error with the Digital Output on the I-7044
            printf("Digital Output of 7044 is error, Error Code=%d\n", wRetVal);
      else
            printf("The DO value of 7044 is: %u \n", wBuf[5]);
```

```
        // 7044 DI
        DigitalIn(wBuf, fBuf, szSend, szReceive);
        printf("The DI of 7044 : %u \n", wBuf[5]);
    }
    Close_Com(COM9);
    return 0;
}
```

Figure 7.1.1-2 below illustrates the result of the execution.

Figure 7.1.1-2. Results of the demo

7.2. Configuration of multiple spanning tree protocol interface setting with dual LAN

The LP-22xx/8x2x/9x2x series modules include support for the MSTPD daemon, which is the Spanning Tree Protocol (STP) that is always recommended in layer 2 topologies that run on bridges and switches. An extension of STP is the MSTP (Multiple Spanning Tree Protocol), which can be used to prevent loops and broadcast storms on a network, and also provide redundant links for automatic failover if an active link fails. The bridge function allows a virtual interface to be configured so that it can be included in the network. The virtual interface behaves similarly to a physical interface and ensures that LinPAC networking communication is always active, as illustrated in Figure 7.2-1.

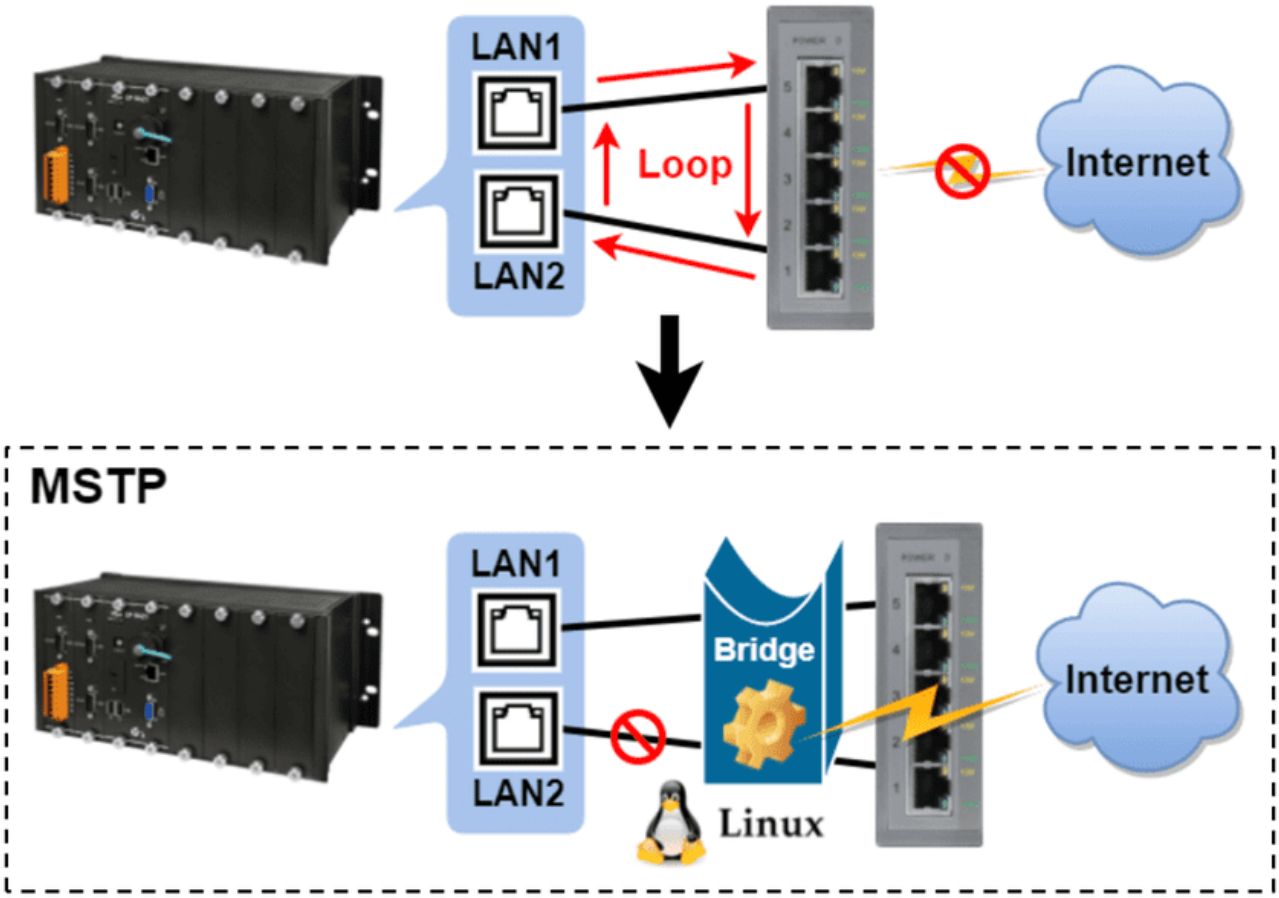


Figure 7.2-1. Using MSTP technology to solve loop problems

The procedure described below will help you to quickly and accurately complete the configuration tasks.

Step 1: Ensure that the bridge-utils package is installed. If not, user can type 'sudo apt-get update' and 'apt-get install bridge-utils' command to install bridge-utils package.

Step 2: Before proceeding, back up the interfaces file found in the '/etc/network/' directory.

Step 3: Locate the interfaces-mstpd file in the '/etc/network/' directory and rename the file as 'interfaces'.

Step 4: Type 'vi /etc/network/interfaces' to edit the file, and enable either a DHCP or a Static IP address by configuring the network settings, as illustrated in Figure 7.2-2 below.

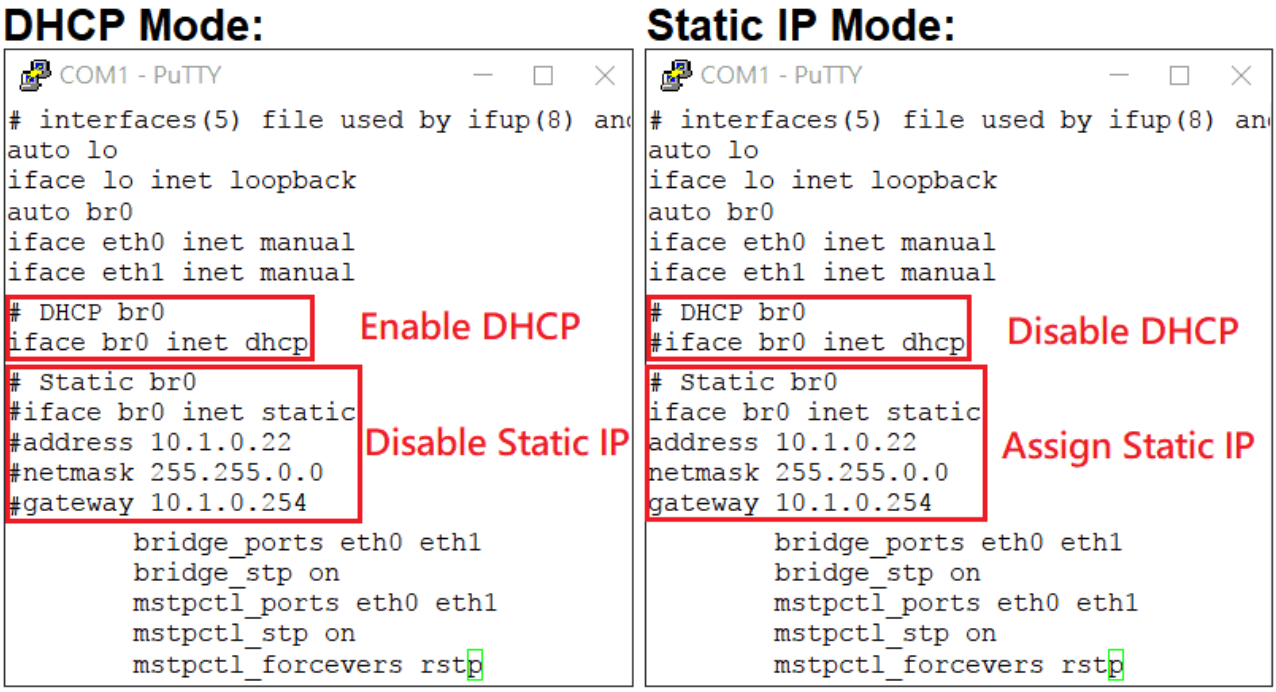


Figure 7.2-2. Configuration for a Dual LAN network

Step 5: Reset the network settings by entering the command 'reboot' or '/etc/init.d/networking restart'.

Step 6: Use the 'ifconfig' command to display the current configuration information for the network interface, as illustrated in Figure 7.2-3.

```
root@icpdas:~# ifconfig
br0     Link encap:Ethernet  HWaddr 68:c9:0b:b4:ba:95
        inet addr:10.1.0.22  Bcast:10.1.255.255  Mask:255.255.0.0
        inet6 addr: fe80::6ac9:bff:feb4:ba95/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:129988 errors:0 dropped:0 overruns:0 frame:0
        TX packets:181 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:18450942 (18.4 MB)  TX bytes:15518 (15.5 KB)
eth0    Link encap:Ethernet  HWaddr 68:c9:0b:b4:ba:95
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:188673 errors:0 dropped:128 overruns:0 frame:0
        TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:36222282 (36.2 MB)  TX bytes:60 (60.0 B)
eth1    Link encap:Ethernet  HWaddr 68:c9:0b:b4:ba:97
        inet6 addr: fe80::6ac9:bff:feb4:ba97/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:188330 errors:0 dropped:128 overruns:0 frame:0
        TX packets:365 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:36198068 (36.1 MB)  TX bytes:26692 (26.6 KB)
```

Figure 7.2-3. Checking the IP address that has been assigned to the br0 interface

Step 7: Use the 'mstpctl showbridge br0' command to verify that the 'force protocol version' is set to RSTP mode, as illustrated in Figure 7.2-4.

```
root@icpdas:/# mstpctl showbridge br0
br0 CIST info
  enabled         yes
  bridge id       8.000.68:C9:0B:B4:BA:95
  designated root 8.000.00:0C:DB:CE:E0:69
  regional root   8.000.68:C9:0B:B4:BA:95
  root port       eth1 (#1)
  path cost     200019     internal path cost   0
  max age       20         bridge max age       20
  forward delay 15         bridge forward delay 15
  tx hold count 6          max hops             20
  hello time    2          ageing time          300
  force protocol version     rstp
  time since topology change 2084
  topology change count      5
  topology change            no
  topology change port       eth0
  last topology change port  eth1
```

Figure 7.2-4. Check that the MSTPD service is correctly configured

Step 8: Use the 'mstpctl showport br0' command to check the connection status of the two LANs and monitor any changes in the status when inserting or removing the Ethernet cable, as illustrated in Figure 7.2-5.

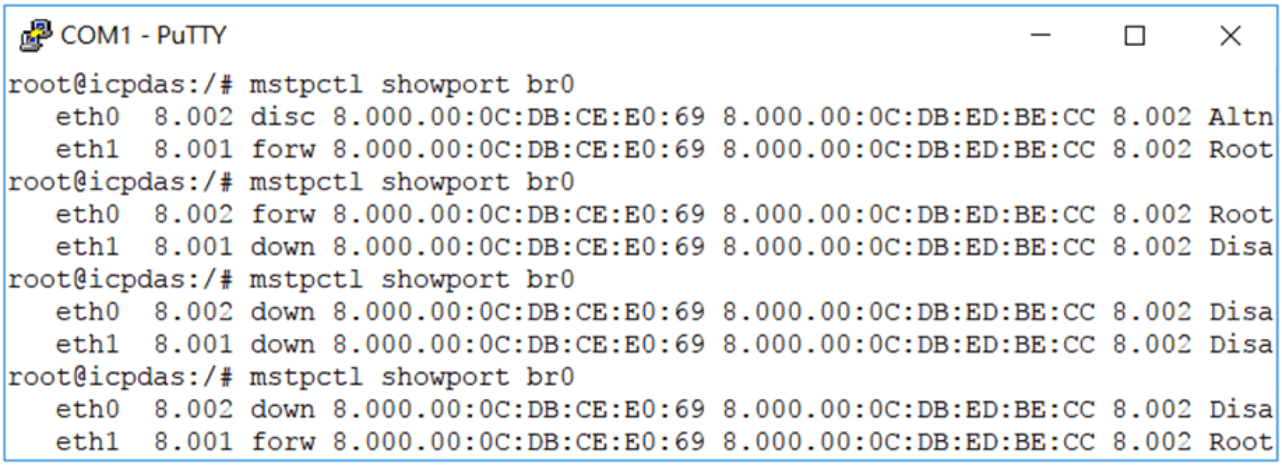


Figure 7.2-5. Monitoring the status of the MSTPD bridge port

7.3. Building a sample MQTT application using LinPAC

MQTT (Message Queuing Telemetry Transport) is an ISO standard publish-subscribe based messaging protocol. It is a lightweight connectivity protocol for M2M (machine to machine) communication that works on top of the TCP/IP protocol. The protocol uses a publish-subscribe model for message exchange between machines, based on the efficient distribution of messages to one or many receivers through a broker. It is useful for mobile applications because of its small size, low power usage and minimized data packets.

The LinPAC controller provides message transfer functionality for MQTT applications. Let's look at an example where we wish to communicate between an MQTT client and an MQTT server (broker). The following is a demonstration of the configuration and processes involved, as illustrated in Figure 7.3-1.

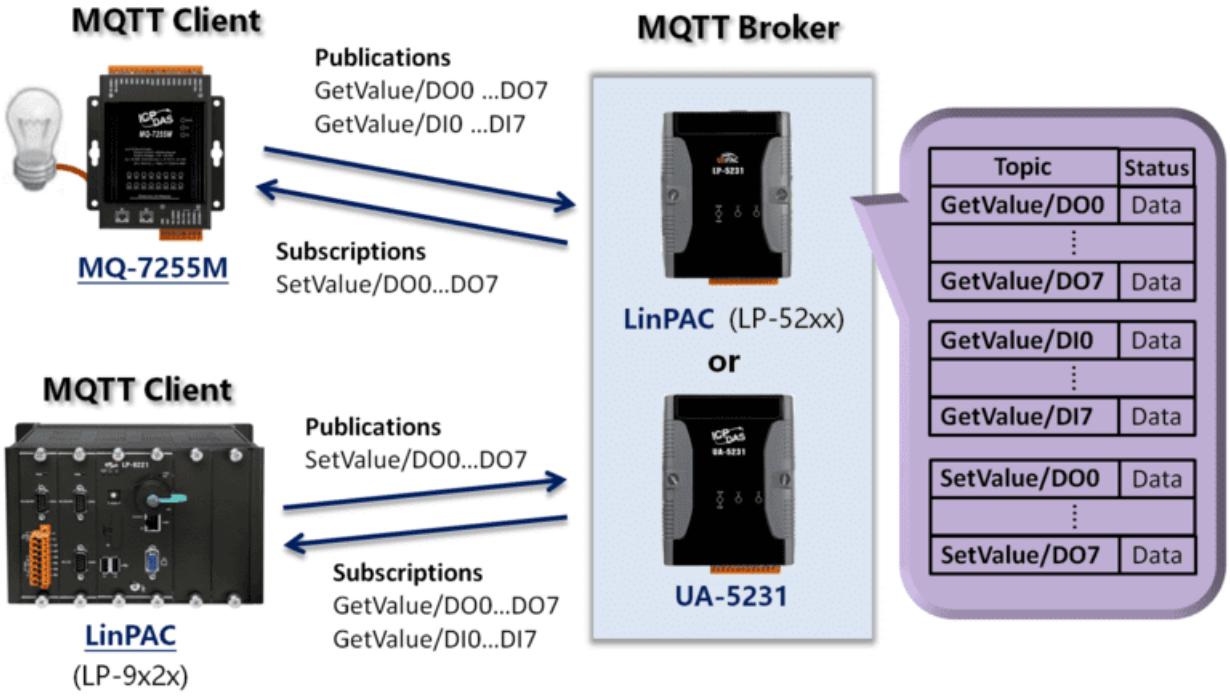


Figure 7.3-1. MQTT architecture for LinPAC

Refer to the following websites for more detailed information.

UA-5200 series:

http://www.icpdas.com/root/product/solutions/industrial_communication/m2m_iiot_server/opc_ua.html

MQ-7200M series:

http://www.icpdas.com/root/product/solutions/remote_io/mqtt_io/mq-7200m_introduction.html

We use LinPAC series devices as both the MQTT client and the MQTT broker, and communicate through the broker to control the MQ-7255M module, as you can see from the architecture diagram in Figure 7.3-1, the LP-9x2x module is used as the MQTT client, and the LP-52xx module is used as the MQTT broker.

You must install the relevant packages discussed below for testing purposes on LinPAC series modules that are either an MQTT broker or an MQTT client. Before installing the packages, it is recommended that you use the command '**sudo apt-get update**' to upgrade the package lists.

<MQTT broker>

You can deploy the MQTT broker on either a UA-5200 or a LinPAC series module (LP-52xx). If you need to deploy an MQTT broker on a LinPAC series module, follow the procedure described below.

Step 1: Use the following command to install the Mosquitto package on the LP-52xx module.

```
# sudo apt-get install mosquitto
```

Step 2: Use the following command to verify that the MQTT broker/server is active, as illustrated in Figure 7.3-2.

```
# netstat -tnl | grep 1883
```

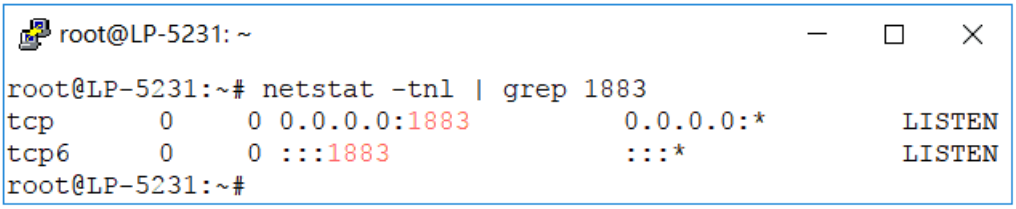


Figure 7.3-2. Confirm that the MQTT broker is listening on Port 1883

<MQTT client>

Use the following command to install the Mosquitto-client package on the LinPAC series module (LP-9x2x).

```
# sudo apt-get install mosquitto-clients
```

As an MQTT client, the MQ-7200M series module is able to publish messages that related to the status of the Digital I/O to a broker and subscribe to topics from a broker that are used to control the DO lines. In a similar way, other MQTT clients can obtain the status information from the Digital I/O by subscribing to a topic from the broker and then publishing messages to the broker that then control the DO lines, as illustrated in Figure 7.3-3. An example application is demonstrated below.

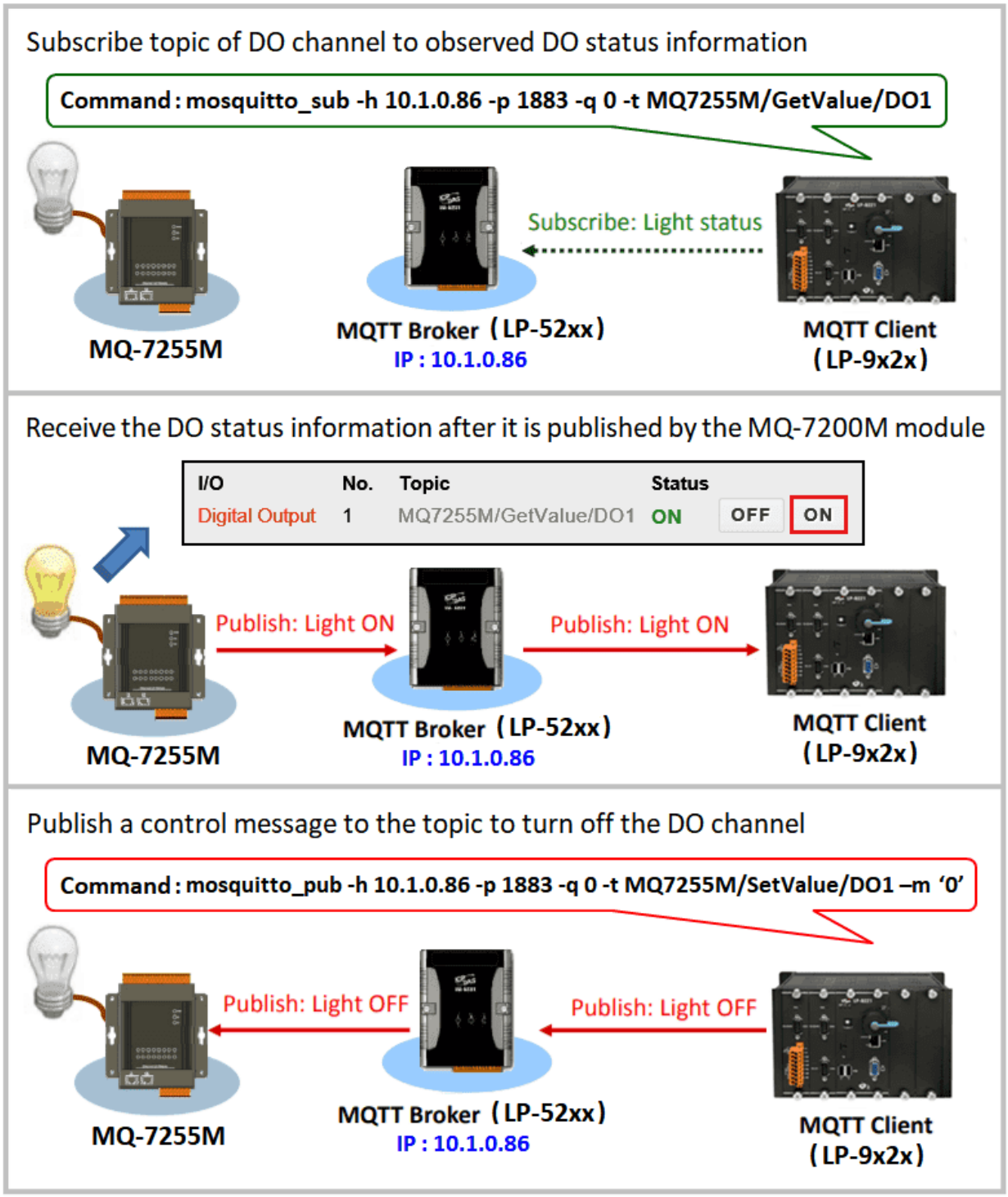


Figure 7.3-3. Controlling the device via the MQTT broker

Step 1: Connecting an MQ-7255M module to a LinPAC MQTT broker.

On the configuration page for the MQ-7255M module, first click the 'MQTT' option. Modify the configuration for the built-in web browser on the MQ-7255M module, as illustrated in Figure 7.3-4.

For more detailed information, refer to Chapter 4.3 of the MQ-7200M user manual, which you can download from:

http://ftp.icpdas.com/pub/cd/mq-7200m/document/mq-7200m_user_manual_english_v100.pdf

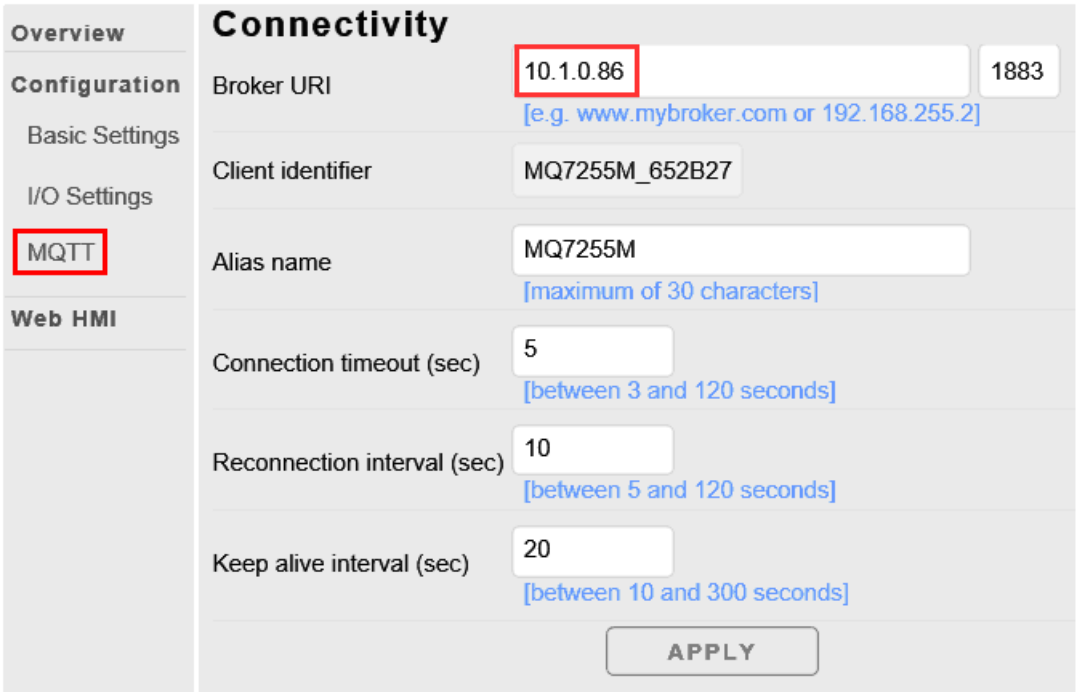


Figure 7.3-4. Connectivity information

Step 2: Managing topics and subscriptions.

On the LP-9x2x module, use the following command to subscribe to a topic from the MQ7255M via the MQTT Broker.

```
# mosquitto_sub -h <broker IP> -p 1883 -q 0 -t MQ7255M/GetValue/DO1
```

In this example, the topic being subscribed to is 'MQ7255M/GetValue/DO1', as illustrated in Figure 7.3-5. The MQ-7255M module will then continuously publish the status information from the DO1 channel to the topic 'MQ7255M/GetValue/DO1'. Messages will also be published in cases where an I/O event causes the status to change, meaning that we can quickly monitor the status of the DO1 channel from the module.

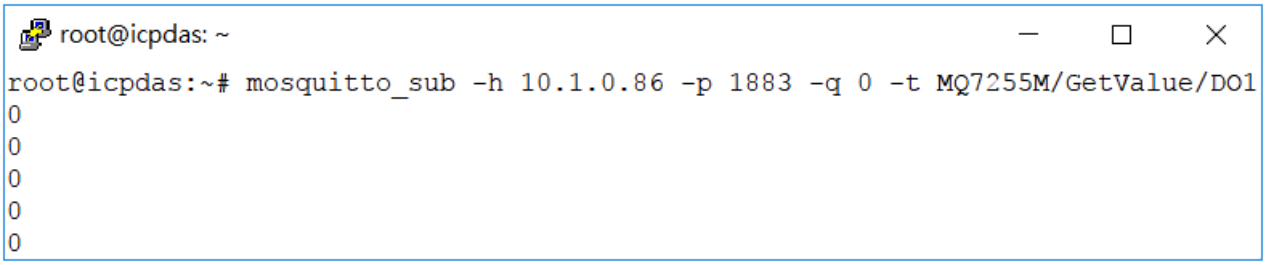


Figure 7.3-5. Subscribing to a topic

Step 3: Managing topics and publishing.

On the LP-9x2x module, use the following command to publish a message from the LP-9x2x module to the MQ-7255M module via the MQTT Broker.

```
#   mosquitto_pub -h <broker IP> -p 1883 -q 0 -t MQ7255M/SetValue/DO1 –m ‘1’
```

In this example, the topic being published is ‘MQ7255M/SetValue/DO1’. We can use the command to turn the device OFF or ON by publishing the status value ‘0’ or ‘1’ for the DO1 channel to the topic ‘MQ7255M/SetValue/DO1’. In a similar way, we can also publish changes to the I/O value to the relevant topic, which then allows us to control the device.

Step 4: Verify that the value of the subscription topic ‘MQ7255M/GetValue/DO1’ has changed.

You can determine if there are any variations in the DO1 value by monitoring the changes in the value published to the ‘MQ7255M/GetValue/DO1’ topic, as illustrated in Figure 7.3-6 below.

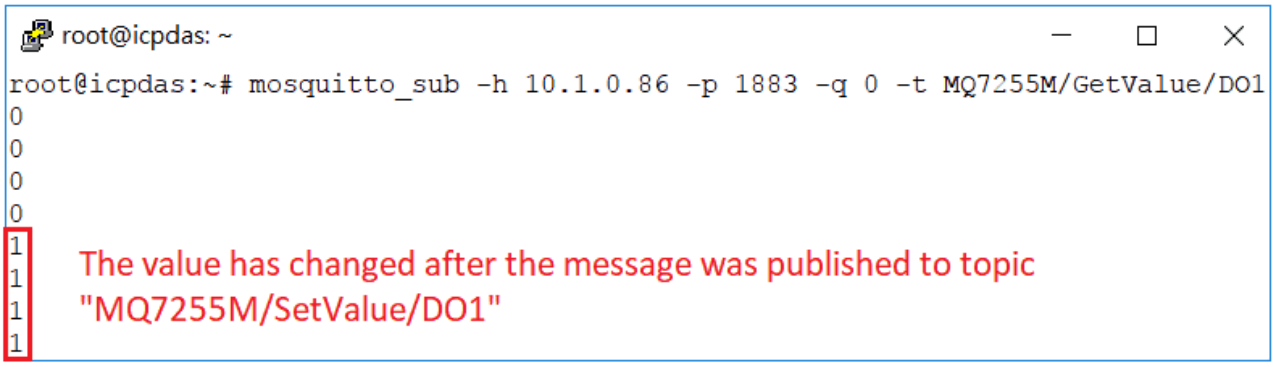


Figure 7.3-6. Monitor the variation in the DO1 value

7.4. Power-on Value Settings

This section will discuss the Power-on Value functions that are adopted in the Linux PAC. When the Power-on Value function is active, the DO or AO output will be restored to the preconfigured value if the power supply has an on/off switch. The following is an operational example using an I-87024W module inserted in an LP-8x21 controller.

Communicate with the I-87K Series module using the setexdo.c and getsendreceive.c programs. Sample code can be found in the C:\cygwin\LinPAC_am335x_SDK\examples\common or root@LinuxPC-ICPDAS:/icpdas/linpac_am335x_sdk/i8k/examples/common folder. The following is the procedure for configuring the Power-on Value:

Step 1: Configure the channel value for the I/O module

Set the Analog Output value to **2 V** for **channel 0** on the I-87024W module and use a multimeter to measure the output voltage of channel 0, as illustrated in Figure 7.4-1.

```
#   ./setexao   1   1   2   0        // Command syntax: setexao <slot> 1 <value> <channel>
```

Step 2: Set the Power-on Value for the specified output channel

Set the Analog Output value for channel 0 to the Power-on Value on the I-87024W module, as illustrated in Figure 7.4-1.

```
#   getsendreceive   1   1   1   '$0040'   115200
// DCON command: '$AA4N'. Refer to the Notes below for additional information.
```

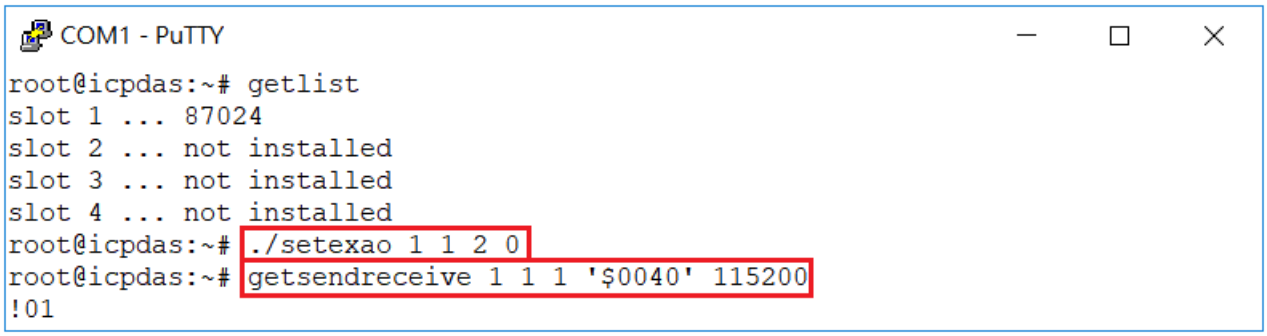


Figure 7.4-1. Power-on Value settings

Step 3: Restart the LinPAC

Switch off the LinPAC and then restart it. Channel 0 on the I-87024W module will output 2 V.

Step 4: Confirm that the Power-on Value is functioning correctly

Use the multimeter to measure the channel output voltage on the I-87024W module and verify that the voltage is the same as before the LinPAC was restarted.

Notes:

1. Use the DCON protocol to configure the I-87024W module. For more information about the command set for the I-87024W module, refer to:
   http://ftp.icpdas.com/pub/cd/8000cd/napdos/dcon/io_module/87k/aio/87024w_rw/dcon_87k/87024_dcon_87k.htm
2. The **address** and **baudrate** for the LP-8x2x are **00** and **115200** respectively. They are fixed-parameters in the library and cannot be modified.

Appendix

A. I-8K Modules and I-87K Modules

This chapter provides a brief overview of the differences between the I-8K and I-87K series modules.

I-8K and I-87K modules provide the option to expand the local I/O to expansion I/O slots and the bus type for the modules can be either parallel (high profile I-8K series) or serial (high profile I-87K series).

The differences between the I-8K and I-87K series modules are as follows.

Item	I-8K Series	I-87K Series
Microprocessor	No	Yes (8051)
Communication Interface	Parallel Bus	Serial Bus
Communication Speed	Fast	Slow
Latched DI Function	No	Yes
Counter Input (for digital input modules)	No	Yes (100 Hz)
Power-on Value	No	Yes
Safe Value	No	Yes
Programmable Slew-Rate for AO modules	No	Yes

For full details of specifications which can be found at:

http://www.icpdas.com/products/PAC/winpac/io_support_list.htm

B. I-9K Modules and I-97K Modules

This chapter provides a brief overview of the differences between the I-9K and I-97K series modules.

There are two types of I/O modules provided for supporting LP-92xx. One is high communication speed I-9K series modules with parallel interface; the other is I-97K series modules with serial interface. The differences between the two series are listed as follows:

The differences between the I-9K and I-97K series I/O modules are as follows.

Item	I-9K Series	I-97K Series
Communication Interface	Parallel Bus	Serial Bus
Protocol	-	DCON
Communication Speed	Fast	Slow
DI with latched function	-	Y
DI with counter input	-	Y (100 Hz)
Power on value	-	Y
Safe Value	-	Y
Programmable slew-rate for AO module	-	Y

For full details of specifications which can be found at:
http://www.icpdas.com/root/product/solutions/remote_io/i-9k_i-97k/i-9k_i-97k_selection.html

C. XV-Board Modules

The XV-board series are for LP-22xx/52xx and WP-2x41/52xx-CE7. One PAC can only plug only one XV-board. The XV-board series have following common specification:

- DI channel is dry contact, sink type.
- DO channel is open collector, sink type.

For more detailed information about these support modules, please refer to

http://www.icpdas.com/root/product/solutions/hmi_touch_monitor/touchpad/xv-board_selection.html

D. Revision History

This chapter provides revision history information to this document.

The table below shows the revision history.

Revision	Date	Description
V1.0.0	July 2018	Initial issue
V2.0.0	Jan 2019	1. Add the product introduction for LP-22xx. 2. Add the product introduction for LP-5231PM-4GE. 3. Add the product introduction for LP-5231PM-4GC. 4. Add the Power-on Value Settings.

E-mail: service@icpdas.com