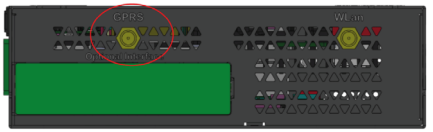
Features	Values
3G/UMTS/HSPA	Bands: 850/AWS/1900/2100 MHz
	42 Mps downlink, 5.76 Mbps uplink
EDGE/GPRS/GSM	Bands: 850/900/1800/1900
	GPRS and EDGE Class 12
SMS	MT/MO PDU/test mode
	SMS over IMS and via SMS-C
Support USB 2.0 interface	
RF interface: 2 Hirose UFL-R_XMT (50 ohm)	

Cellular View



4G Module Configuration

The 4G module is pre-loaded in OS image, you need to go to 4G setting flag to enable it. Refer to Editing the Configuration File *(see page 118)*.

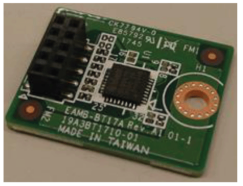
Cyber Security TPM Module Description

Introduction

The HMIYBINLTPM201 is categorized as industrial module. It is compatible with the low pin count module. The Trusted Platform Module (TPM) is an international standard for a secure cryptoprocessor, which is a dedicated microcontroller designed to secure hardware by integrating cryptographic keys into devices.

The mother board and the OS of Box PC IIoT allows you to install the TPM module and activate encryption. Then, storage drives and operating system are encrypted according to password and keys managed within the hardware module.

According to part number, the HMIYBINLTPM201 TPM module can default mounted following the CTO (configured to order) or can be user mounted afterward as an optional accessory module. The encryption can be activated.



Plug the module onto the Box PC IIoT pin header.

TPM Module Installation

Before installing or removing a card, shut down the operating system in an orderly fashion and remove the power from the device.

NOTICE
ELECTROSTATIC DISCHARGE Take the necessary protective measures against electrostatic discharge before attempting to remove the Magelis Industrial PC cover. **Failure to follow these instructions can result in equipment damage.**

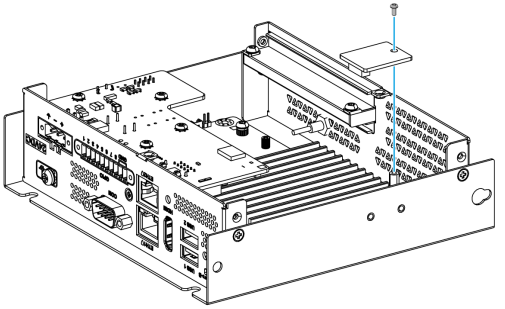
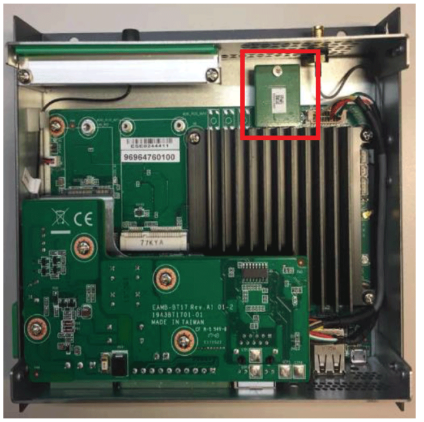
⚠ CAUTION

OVERTORQUE AND LOOSE HARDWARE

- Do not exert more than 0.5 Nm (4.5 lb-in) of torque when tightening the installation fastener, enclosure, accessory, or terminal block screws. Tightening the screws with excessive force can damage the installation fastener.
- When fastening or removing screws, ensure that they do not fall inside the Magelis Industrial PC chassis.

Failure to follow these instructions can result in injury or equipment damage.

The table describes how to install a TPM module:

Step	Action
1	Install TPM card: 
2	Lock screw:  **NOTE:** The recommended torque to tighten these screws is 0.5 Nm (4.5 lb-in).

TPM Module Configuration

The TPM module is pre-loaded in OS image. Refer to Editing the Configuration File *(see page 118)*.

Chapter 7
Configuration Software

This section contains the information required to get started with the Linux Yocto Board Support Packages (BSP).

What Is in This Chapter?

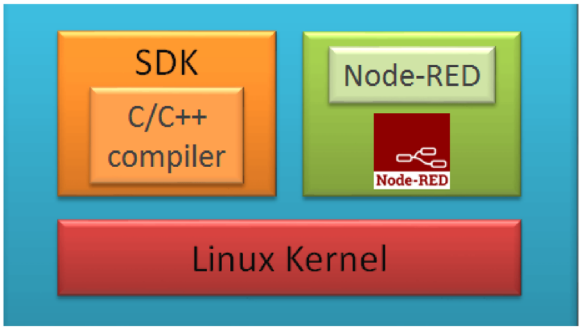
This chapter contains the following topics:

Configuration

Overview

The Box PC IIoT (HMIBSC) has a default software package architecture based on Linux Yocto project (open source embedded Linux Build).

The software stack is built from different layers, they are described in the diagram:



A software development kit (SDK) is typically a set of software development tools that allows the creation of applications for a certain software package, software framework, hardware platform.

In embedded systems, a board support package (BSP) is the layer of software containing hardware-specific drivers and other routines that allow a particular operating system to function in a particular hardware environment

NOTE: C/C++ compiler for customization by skilled developers only. Documentation on request to Customer care center, with limited support.

General Information

The Linux Yocto BSP provide software and recipes necessary to support individual boards. The BSP is a collection of information that defines how to support a particular hardware device, set of devices, or hardware platform.

This manual does not show how to solve every possible programming issue. The Linux Yocto Project has the aim and objective of attempting to improve the user experience of developers of customized Linux systems supporting the ARM CPU architecture. To use this manual, you should be familiar with Linux command and Linux Yocto project.

Software Version

Customization table	Software version or later
Linux Yocto project	Krogoth 2.1
BitBake branch	1.30.0
Linux Kernel	4.4.38
GCC	5.2.1
GNU C library (glibc)	2.23
Node.js	6.10.3
Node.RED	0.17.5

NOTE: The Node-RED is pre-installed. For upgrading Node-RED, refer to *https://nodered.org/*.

Node-RED coming from the OS image has been validated. If you want to change Node-RED, follow the installation procedure on Node-RED website. (*https://nodered.org/docs/getting-started/installation*)

Standard Node-RED provides the standard Node. To know how to use each node and how to make a link, refer to Node-RED *official website*.

Account and Authority Management

⚠ WARNING

UNAUTHORIZED DATA ACCESS

- Immediately change any default passwords to new, secure passwords.
- Do not distribute passwords to unauthorized or otherwise unqualified personnel.
- Limit access-rights to the user Everyone to only those essential to your application needs.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Start to Use Box PC IIoT

There are default passwords for OS login and Node-RED. These default passwords are set on new products or after system restore. User must change the default password for **root account**, **Node-Red account** and **user account** after first long in.

OS Login Password Change

Step	Action
1	Power up the Box PC IIoT at the first time.
2	User is required to change OS login password for root account after first login.
3	Default user name is **root**. The default password is **IIoTB#r8**
4	Follow this password change policy: ● Passwords must have at least 12 characters. ● Passwords cannot contain the username. ● Passwords must include the four available character types: lowercase letters, uppercase letters, numbers, and symbols. Symbols must include any one of [!"#$%&'()*+,./:;<=>?@\^_`{|}~-]. `### ========== ###` `### Please follow the policy to change the password for security ###` `### ========== ###` `- Passwords must have at least 12 characters.` `- Passwords can not contain the username.` `- Passwords must include the four available character types: lowercase letters, uppercase letters, numbers, and symbols.` `### ========== ###` `### ========== ###` `### Change root password ###` `### ========== ###` `You are required to change your password immediately (root enforced)` `Changing password for root` `New password:` `Retype new password:` `passwd: password updated successfully` `### (Info) root password has been changed successfully! ###` **NOTE:** If the password key-in is not met above criteria, system will request again to key in password until meet criteria.

Step	Action
5	After change root password, user is required to change the password of user account: `### ==================== ###` `### Change user password ###` `### ==================== ###` `You are required to change your password immediately (user enforced)` `Changing password for user` `New password:` `Retype new password:` `passwd: password updated successfully` `### (Info) user password has been changed successfully! ###`

Node-RED Password Change

Step	Action
1	After OS login passwords are changed, user is required to change default Node-RED login password. The default username is **NR_account** and password is **NodeRed#0123** `### ======================== ###` `### Change Node-RED password ###` `### ======================== ###` `You are required to change Node-RED login password immediately (root enforced)` `Change password for Node-RED` `Enter current Node-RED password:` `### [info] Node-RED password correct! ###` `Enter New Node-RED password:` `Retype new Node-RED password:` `### [info] Node-RED password matches! ###` `### [info] Node-RED password has been changed successfully! ###` `### [info] Reboot system now ###` `Rebooting.` **NOTE:** Only **Root Account** has right to change Node-RED password.

OS Login

Step	Action
1	Power on Box PC IIoT every time after default OS login password is changed and default Node-RED login password is changed.
2	● If choose root account, user is required to enter root account password. ● If choose user account, user is required to enter user account password.

Node-RED Password Change

Step	Action
1	You have to complete to change Node-RED password before using Node-RED.
2	Only Root Account has right to Node-RED password. User has to change the default password for Root account, Node-RED account and User account after first login.
3	Power up the Box PC IIoT every time after OS login password is changed and Node-RED login password is changed: ● If you choose root account, you are required to enter root account password. ● If you choose user account, you are required to enter user account password.
4	Enter https://<IP address>:1880 (Port number: 1880) from remote site to use. Password is required to enter every time.

Node-RED Quick Start Configuration

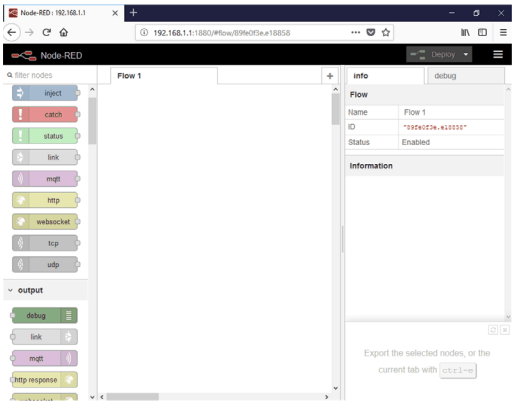
General Information

Node-RED solution is to provide standard Node-RED pre-installed in OS image and Schneider Node which you can install from recovery SD card. Schneider Node also provides sample code and flow sample to help you to use quickly.

Starting Procedure

This procedure gives the information about how to set the Node-RED:

Step	Action	Action
1	Power up the Box.	
2	Open the configuration file.	`$ vi ~/infra_setting.conf`
3	Switch to insert mode by press **i**.	
4	Enable to change Node-RED configuration.	**NODERED_SETTINGS_FLAG=1**
5	Set up network IP address to the Box PC IIoT (HMIBSC).	**LAN_0_SETTINGS_FLAG** (enable to modify the LAN 0 configuration). **LAN_0_ENABLE_STATIC_IP** (switch static IP or DHCP for LAN port 0). **LAN_0_NETMASK_BIT_COUNT** (set the network mask bit count afterwards for LAN port 0).
6	Press **Esc** to leave insert mode.	
7	Type **:wq** to save the setting and exit the editor.	
8	Whenever you modify the configuration file, the settings will be updated after you restart the device.	`$ sync` `$ reboot`
9	Open a browser from a computer in the same network.	
10	Type **https:\\<ip address>:1880**.	
11	After login in the root account, insert the installer SD card.	
12	Type a command to install the package automatically `/run/media/mmcblk1p9/Software/SEnode_Install_package/install.sh`.	
13	After all the install processes are finished, unplug the SD card and the restart the device.	

Step	Action	Action
14	You get Node-RED user interface: 	

Standard Node-RED

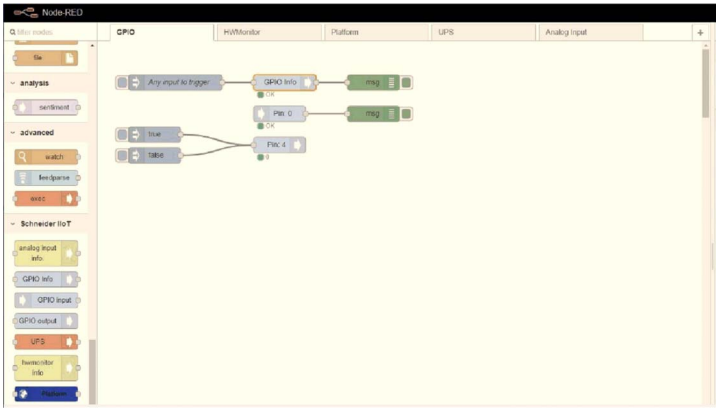
Node-RED is embedded in HMIBSC Operating System image. To up-date the Node-RED version, follow the default installation procedure on Node-RED website. *https://nodered.org/docs/getting-started/installation*

User has to complete the default password change before using Node-RED.

Enter IP address:1880 (port number: 1880) from remote site to use. The password is required to enter every time.

Schneider Electric Node Installation

Node-RED solution is to provide standard Node-RED pre-installed in OS image and Schneider Node which user can install from recovery USB key. Schneider Node also provides sample code and flow sample to help user to use quickly.

Step	Action
1	install Schneider node which are in the recovery SD card/packages folder.
2	After log in the root account, insert the installer SD card.
3	Type a command to install the packages automatically /run/media/mmcblk1p9/Software/SEnode_Install_packages/Install.sh.
4	After all the install processes are finished, unplug the SD card and then restart the device.
5	You see Schneider IIoT Nodes are added in **Node List**: 
6	Scroll down to see Schneider IIoT Node.

NOTE: Although Node-RED have standard Node build in, there is no special Node that can support Schneider-Electric hardware unless you install Schneider-Electric Node. Schneider-Electric Node offers the required.

Start to Use Node-RED

- Node-RED password:
  - Power on the Box iPC for the first time: user is required to change Node-RED password.
  - User has to enter password every time after power-on.
  - Node-RED password has to be changed the first time Node-RED is used.
- User Node-RED:
  - User is required to change password after powering on the Box iPC for the first time (right after the procedure of changing Node-RED password).
  - Enter the Box iPC IP address from remote site. Password is required to enter every time.

Schneider IIoT Node List

- **Platform**
- **UPS**
- **Hardware Monitor**
- **GPIO Set**
- **AI module**

NOTE: You can simply change the value in simple code (flow sample code installer), which can be installed through SD card.

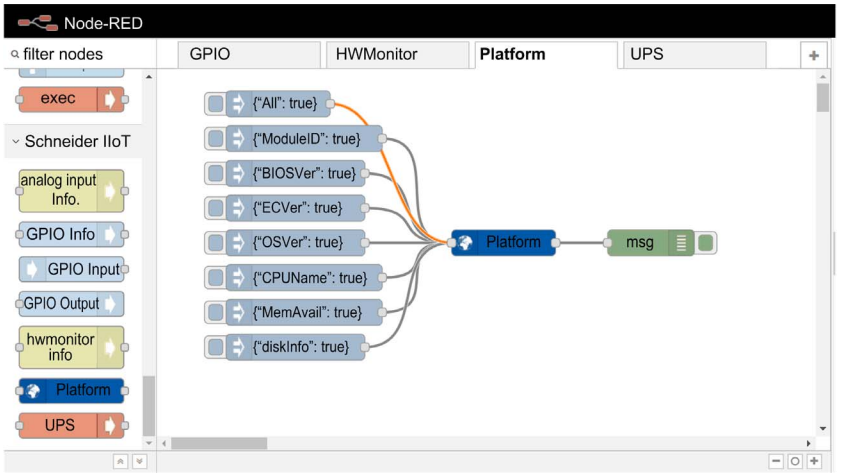
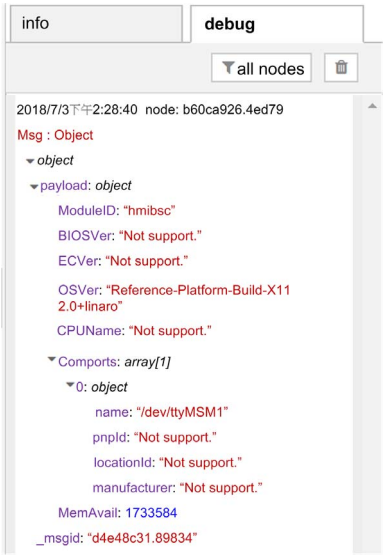
Platform Node

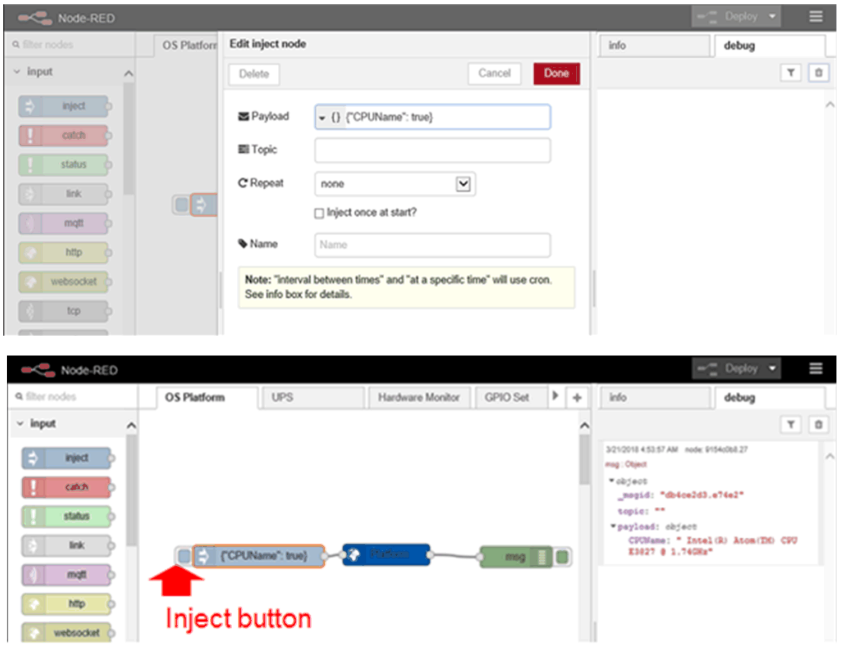
The following information can be obtained from **Platform** node:

Node Name	Information	Description/Value
Platform	**Model Name**	The information from Windows API or supplier SNMP.
	EC version	
	OS version	
	CPU name	
	Memory available	
	Disk information	

Platform node sends the information once at first, and if you want **OSVer** value, input **OSVer** attribute and set it to true to get only **OSVer** value.

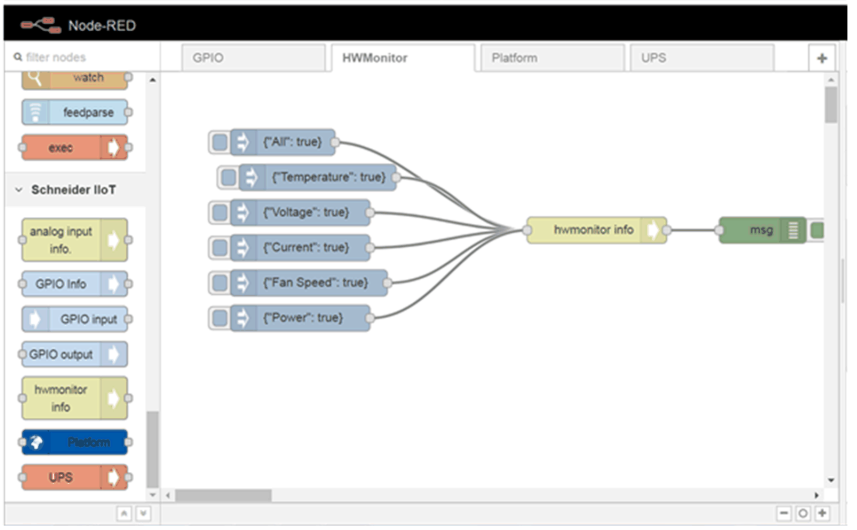
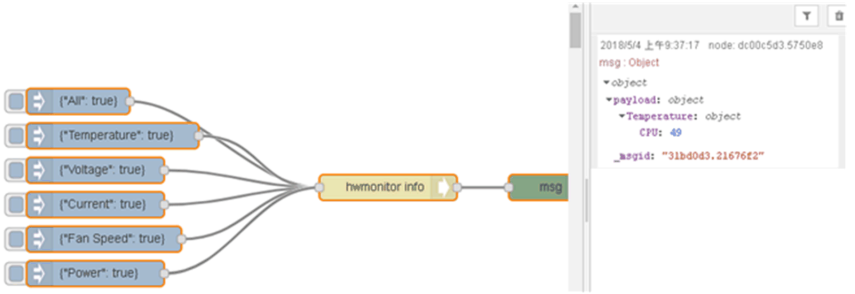
If you want the program to do setting instead of using Node, here is sample code for your reference.

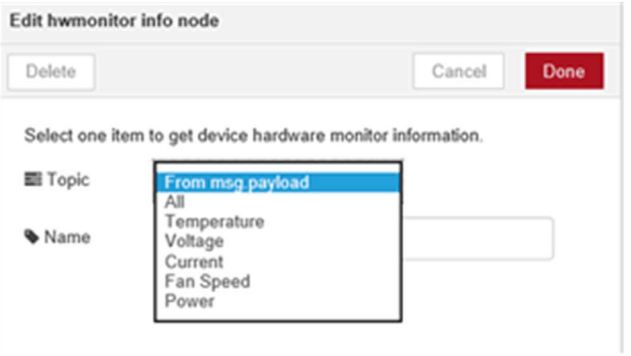
Step	Action
1	Select **Platform** page.
2	Double-click **Platform** node: 
3	Click **Deploy** button to get all information from debug area. 

Step	Action
4	If you want the specific information (for example, CPU name): ● Click **Platform** node to change OSVer in the payload column to CPUName. ● Click **Done** to close the window of **Edit** inject node. ● Click **Deploy** and then click **Inject** button to verify the result in debug window.
5	Click **Deploy** button and then click **Inject** button to verify the result in debug window: 
6	Sample flow reference. User can get all up-to-date sample flow from bellow link: */usr/lib/node_modules/node-red-contrib-seplatform*.

Hardware Monitor Node

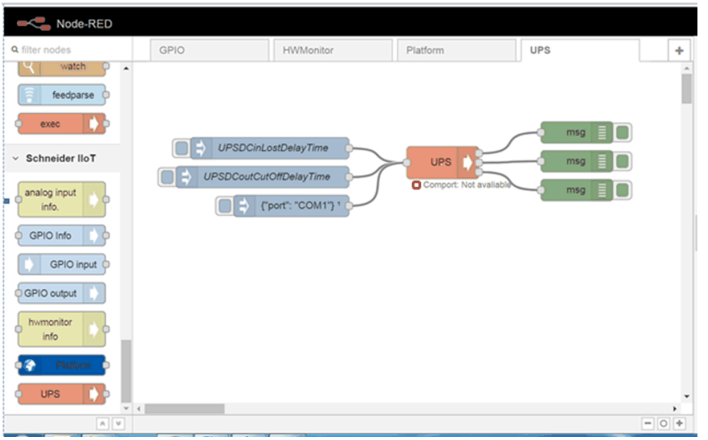
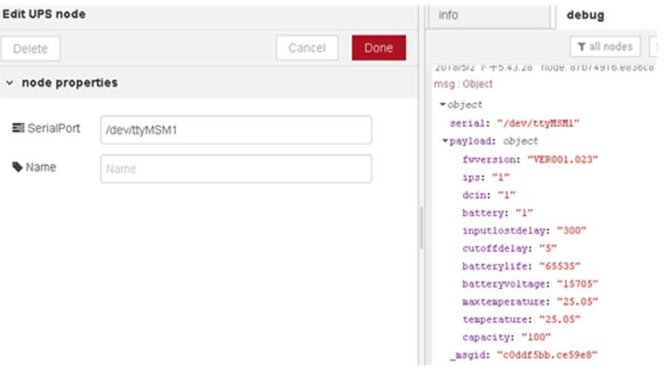
Node Name	Information	Description/Value
Hardware Monitor	Temperature	All current information from embedded control.
	Voltage	
	Current	

Step	Action
1	Select **Hardware Monitor** page.
2	Double-click **Hardware Monitor** node: 
3	Click **Deploy** button to get all information from debug area: 

Step	Action
4	If you want the specific information (for example, Voltage): ● Click **hwmonitor info** node to change all in the topic column to voltage. ● Click **Done** to close the window of **Edit hwmonitor info** node. ● Click **Deploy** and then click **Inject** button to verify the result in debug window. 
5	Sample flow reference. User can get all up-to-date sample flow from bellow link: */usr/lib/node_modules/node-red-contrib-selmsensor*.

UPS Node

Node Name	Information	Description/Value
UPS	Emergency Output	• DC-IN is lost. • Battery over temperature. • Battery gauge is lost connection. • EEPROM accesses fail. • DC-IN is over voltage. • DC-Out cut-off trigger. • Restores power to IPS-AE DC-IN.
	Status output	• fwversion: device firmware version. • ips: the status of device. 1 is ready and 0 is not ready. • dcin: the status of DC-IN. 1 is ready and 0 is not ready. • battery: the status of battery. 1 is ready and 0 is not ready. • inputlostdelay: the DC Input Lost detection duration(sec). • utoffdelay: the DC-OUT cut-off delay time(minutes). • batterylife: battery life (minutes) at the present rate of discharge. "65535" is battery charged. • temperature: battery. temperature (Celsius). • maxtemperature: It is the max temperature (Celsius) of battery from the system started. • batteryvoltage: It is the battery voltage (mV). • capacity: battery capacity (%).
	Response output	Describe the input result.

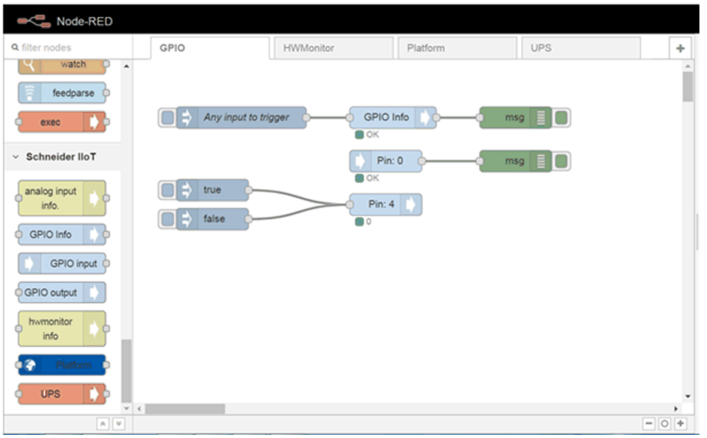
Step	Action
1	Select **UPS** page.
2	Double-click **UPS** node: 
3	Sample code: ● The inputs must be `msg.payload.UPSInputLostDelay` and `msg.payload.UPSCutOffDelay` which are numeric. ● `msg.payload.UPSInputLostDelay` is a number which is the DC Input Lost detection duration(sec). ● `msg.payload.UPSCutOffDelay` is a number which is the DC-OUT cut-off delay time(minutes). ● Another input `msg.payload.port` is COM port name which is used to connect with UPS. 

Step	Action
4	Sample code: `var ups;` `try {` `  ups = require('./bin/binding/' + process.platform + '-' + process.arch + '/ipsae');` `} catch (e) {` `  console.error(e);` `  process.exit();` `}` `function emerency(msg)` `{` `  console.log("[emerency] : " + msg);` `}` `function infomation(msg)` `{` `  console.log("[infomation] : " + msg);` `}` `// The first argument may be COMn or /deb/tty*n` `ups.start("COM1", emerency, infomation);` `process.on('SIGINT', function() {` `  ups.bye();` `  process.exit();` `});`

Step	Action
5	Sample code: `// Check if USP is connected` `console.log('UPS status: ' + ups.getSerialStatus());` `// Set DC_IN lost delay time (3 ~ 360s)` `var dcInLostDelayTime = 0;` `console.log('Set DC_IN lost delay time to ' + dcInLostDelayTime + 's: ' + ups.setDCinLostDelayTime(dcInLostDelayTime));` `dcInLostDelayTime = 300;` `console.log('Set DC_IN lost delay time to ' + dcInLostDelayTime + 's: ' + ups.setDCinLostDelayTime(dcInLostDelayTime));` `// Set DC_OUT cut off delay time (1 ~ 10s)` `var dcOutCutOffDelayTime = 0;` `console.log('Set DC_OUT cut off delay time to ' + dcOutCutOffDelayTime + 's: ' + ups.setDCoutCutOffDelayTime(dcOutCutOffDelayTime));` `dcOutCutOffDelayTime = 5;` `console.log('Set DC_OUT cut off delay time to ' + dcOutCutOffDelayTime + 's: ' + ups.setDCoutCutOffDelayTime(dcOutCutOffDelayTime));`

GPIO Set Node

Node Name	Information	Description/Value
GPIO Set	**GPIO input**	Set the selected GPIO pin to input and then read the value from it.
	GPIO output	Set the selected GPIO pin to output and then write a value to it.
	GPIO info	Pin number, mode (In/Out), value (high/low) of each GPIO pin.

Step	Action
1	Select **GPIO** page.
2	Double-click **GPIO** node: 
3	Double-click **GPIO in** settings: 
4	Double-click **GPIO out** settings: 

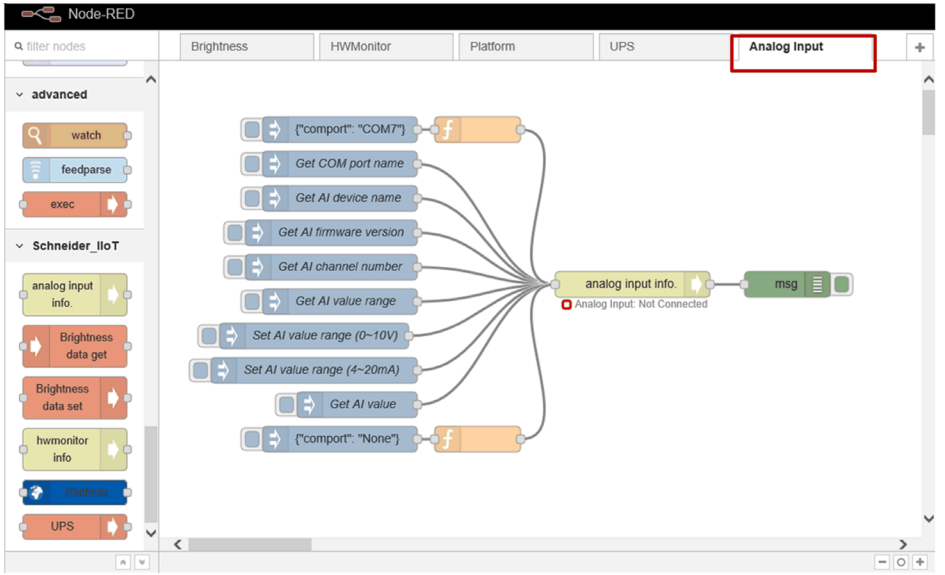
Step	Action
5	Double-click **GPIO info** settings: 
6	Sample flow reference. User can get all up-to-date sample flow from bellow link: */usr/lib/node_modules/node-red-contrib-segpio*.

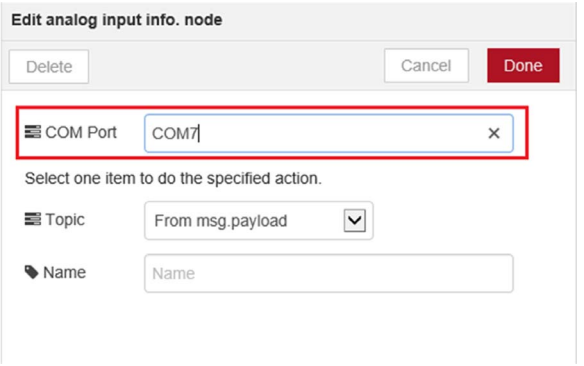
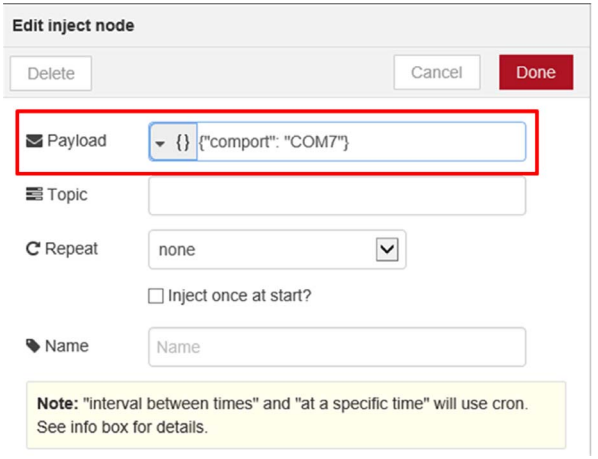
Analog Input Module Node

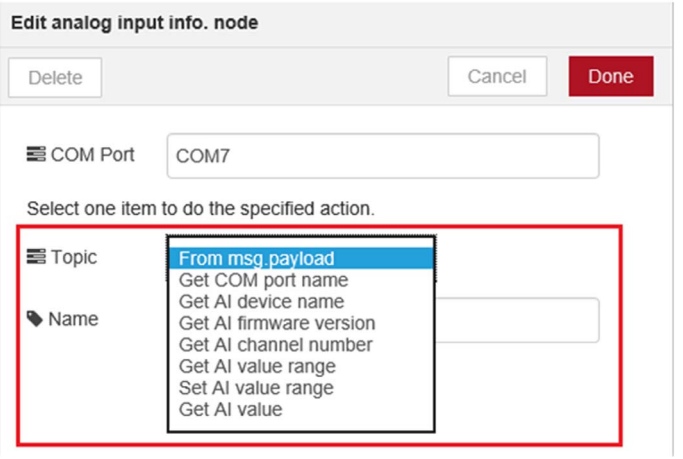
Node Name	Information	Description/Value
AI Module	**Get COM port name**	COM port name (used by this analog input module).
	Get AI device name	Analog input module name.
	Get AI firmware version	Analog input firmware version.
	Get AI channel number	Analog input channel number.
	Get AI value range	Analog input value range.
	Set AI value range	Analog input value range setting.
	Get AI value	Analog input value.

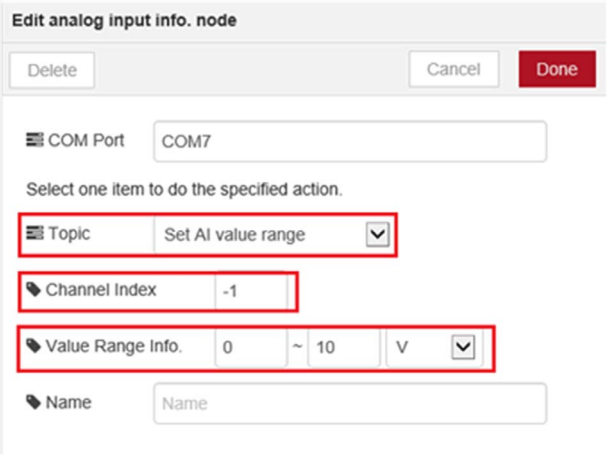
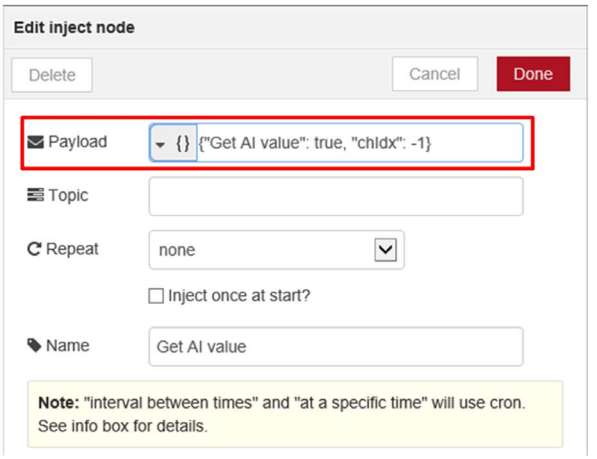
Sample Flow

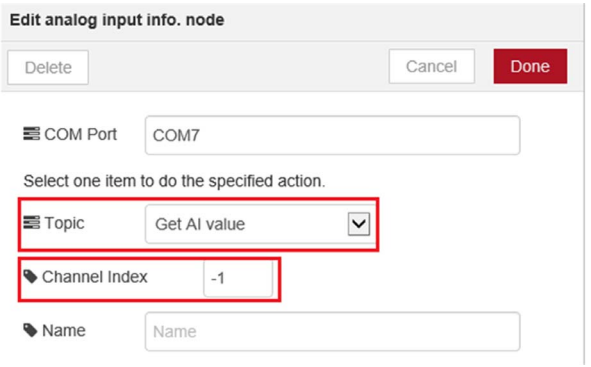
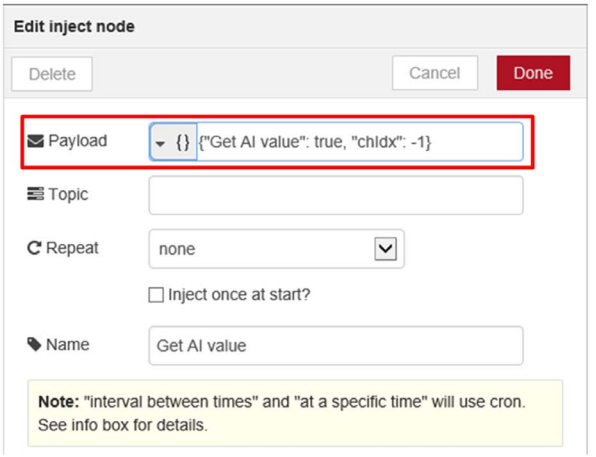
You can create your own analog input module flow or you can select the **Analog Input** tab to get default analog input sample flow and the sample flow is as below:

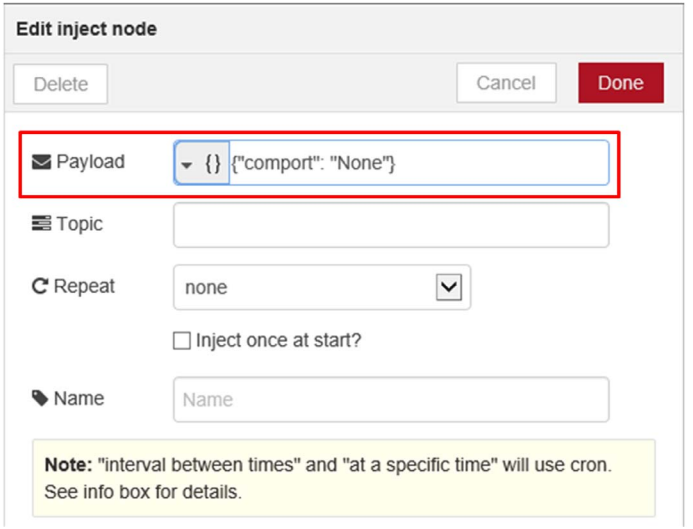


Step	Action
1	At first, COM port path setting is required to make analog input module connect to host. The other functions cannot be used before finishing analog input module connection step. Set a COM port item in an analog **input info node**. (COMx: X = number, for example, COM7, COM number depends on the host.)  **NOTE:** It can also be set by Input {"comport": "COMx"} to analog input info. node. (COMx: x=number, for example, COM7, COM number depends on the host.) For example, if you want to set COM7, set msg.payload to {"comport": "COM7"} and send this message to this node. 

Step	Action
2	Select an item which you want to do in analog input info. node from **Topic** list. 

Step	Action
3	In **analog input info node**, select **Set AI value** from **Topic** list and set **Channel Index** and **Value Range Info** field. **NOTE:** f you want to target all the channels, you can set -1 in **Channel Index** field.  **NOTE:** It can also be set by Input {"Set AI value range": true, "valRangeInfo": {"minValue":0, "maxValue":10, "unit": "V"}, "chIdx": -1} in msg.payload to analog input info. node. For example, if you want to get analog input value, set msg.payload to {"Get AI value": true, "chIdx": -1} and send this message to analog input info. node. For example, if you want to target all the channels, you can set "chIdx": -1. For example, if you want to target channel 2, you can set "chIdx": 2. 

Step	Action
4	In **analog input info node**, select **Get AI value** from **Topic** list and set **Channel Index** field. **NOTE:** f you want to target all the channels, you can set -1 in **Channel Index** field.  **NOTE:** It can also be set by Input {"attribute name": true} in msg.payload to analog input info. node. For example, if you want to get analog input value, set msg.payload to {"Get AI value": true, "chIdx": -1} and send this message to analog input info. node. If you want to target all the channels, you can set "chIdx": -1. If you want to target channel 2, you can set "chIdx": 2. 

Step	Action
5	If you do not need analog input module, you can set input {"comport": "None"} to disconnect the communication between host and analog input module. The disconnected step ends after the state of the node change from connected to disconnected. 
6	Sample flow reference. User can get all up-to-date sample flow from the link: /usr/lib/node_modules/node-red-contrib-seai.

Software Configuration

General Information

This manual provides information about using the configuration files to initialize the device and utilities to change the settings on the device.

This manual does not show you how to solve every possible programming issue. To use this manual, you should already be familiar with Linux shell command. If you need to build your own OS image, customizing your OS image or using SDK to compile the application, contact your local Schneider to get further information and resources for better support.

This user guide is divided into the following sections:

- Using the **Configuration File** on the **Target Device**.
- Using the **Utility** on the **Target Device**.

This section gives the information about how many configurations can be set and how to use them on the target device. Whenever you modify the configuration file, the settings are updated after you restart the device.

Editing The Configuration File

The **vi** editor is a screen-oriented text editor. Use **vi** editor to modify the configuration file,as follows:

Step	Action	Action
1	Open the configuration file.	`$ vi ~/infra_setting.conf`
2	Switch to insert mode by pressing **i**.	
3	Modify the flags. `#### ---------- #### #### ---------- Flags ---------- #### #### ---------- ####` `### 1. Enable to change the COM port configuration` `COM_SETTINGS_FLAG=0` `### 2. Enable to change the GPIO configuration` `GPIO_SETTINGS_FLAG=0` `### 3. Enable to change LAN 0 configuration` `LAN_0_SETTINGS_FLAG=0` `### 4. Enable to change LAN 1 configuration` `LAN_1_SETTINGS_FLAG=0` `### 5. Enable to change WiFi configuration and connect to WiFi` `WIFI_SETTINGS_FLAG=0` `### 6. Enable to change BT configuration` `BT_SETTINGS_FLAG=0` `### 7. Enable to change Node-RED configuration` `NODERED_SETTINGS_FLAG=1` `### Plus 1. Enable to change 4G configuration` `W4G_SETTINGS_FLAG=0` `### Plus 2. Enable to initialize the TPM 2.0 module` `TPM_INIT_FLAG=0`	

Step	Action	Action
4	Modify the parameters and press **Esc** to leave insert mode. `#### ---------- #### #### ---------- Parameters ---------- #### #### ---------- ####` `### 1. The COM Port Configuration ###` `## 1-1. Set the mode of RS-232/422/485 transceiver` `## COM_MODE=1 -> RS-232` `## COM_MODE=2 -> RS-422` `## COM_MODE=3 -> RS-485` `COM_MODE="1"` `## 1-2. Set the COM baudrate` `COM_BAUDRATE="9600"` `## 1-3. COM Parity` `## COM_PARITY=1 -> odd` `## COM_PARITY=0 -> even` `COM_PARITY="1"` `### ---------- ###` `### 2. The GPIO configuration ###` `## 2-1. Load the GPIO configuration file (File: ~/utility/gpio/gpio_config.conf)` `## GPIO_LOAD_CONFIG=1 -> Load the GPIO configuration file` `## GPIO_LOAD_CONFIG=0 -> Use the default value` `## PS. The GPIO default values are all input for safety` `GPIO_LOAD_CONFIG="1"` `### ---------- ###` `### 3. LAN 0 Configuration ###` `## 3-1. Switch static IP or DHCP` `## LAN_x_ENABLE_STATIC_IP=1 -> static IP` `## LAN_x_ENABLE_STATIC_IP=0 -> DHCP` `LAN_0_ENABLE_STATIC_IP="0"`	
5	Type **:wq** to save the setting and exit the editor. **NOTE:** The following common commands are also available: **:clear** - clear the screen **ZZ** - Save and exit **:q!** - Discard the changes, since the last save, and exit **:w** - Save file **:wq** - Save and exit	
6	Whenever you modify the configuration file, the settings will be updated after you restart the device.	`$ sync` `$ reboot`

Configuration List

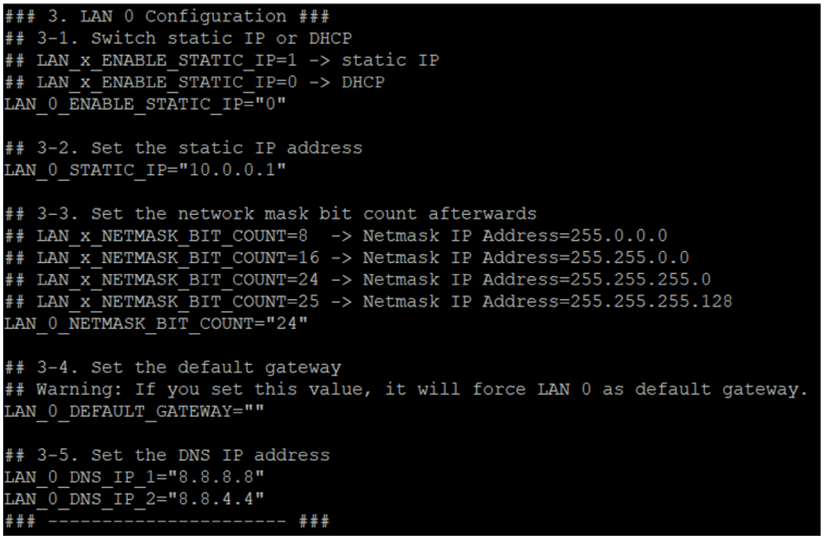
<table>
<tr><th>Item</th><th>Variable</th><th>Description</th></tr>
<tr><td rowspan="5">COM</td><td>COM_SETTINGS_FLAG</td><td>Enable to modify the COM port configuration.</td></tr>
<tr><td>COM_MODE</td><td>Set the COM mode of RS-232/422/485 transceiver.
COM_Mode=0→ RS-232
COM_Mode=1→ RS-422
COM_Mode=2→ RS-485</td></tr>
<tr><td>COM_BAUDRATE</td><td>Set the COM baudrate.</td></tr>
<tr><td>COM_PARITY</td><td>Set the COM parity.
COM_PARITY=1→ odd
COM_PARITY=0→ even</td></tr>
<tr><td colspan="2"><pre>
1. The COM Port Configuration
1-1. Set the mode of RS-232/422/485 transceiver
COM_MODE=1 -> RS-232
COM_MODE=2 -> RS-422
COM_MODE=3 -> RS-485
COM_MODE="1"

1-2. Set the COM baudrate
COM_BAUDRATE="9600"

1-3. COM parity
COM_PARITY=1 -> odd
COM_PARITY=0 -> even
COM_PARITY="1"
------------------------------###
</pre></td></tr>
<tr><td rowspan="3">GPIO</td><td>GPIO_SETTINGS_FLAG</td><td>Enable to modify the GPIO configuration.</td></tr>
<tr><td>GPIO_LOAD_CONFIG</td><td>Load the GPIO configuration file (~/utility/gpio/gpio_config.conf).
GPIO_LOAD_CONFIG=1 → Load the GPIO configuration file
GPIO_LOAD_CONFIG=0 → Use the default value
NOTE: GPIO values are set as default inputs for security reasons.</td></tr>
<tr><td colspan="2"><pre>
2. The GPIO configuration
2-1. Load the GPIO configuration file (File: ~/utility/gpio/gpio_config.conf)
GPIO_LOAD_CONFIG=1 -> Load the GPIO configuration file
GPIO_LOAD_CONFIG=0 -> Use the default value
PS. The GPIO default values are all input for safety
GPIO_LOAD_CONFIG="1"

</pre></td></tr>
</table>

Item	Variable	Description
LAN 0	LAN_0_DNS_IP_1	Set the DNS IP address 1.
	LAN_0_DNS_IP_2	Set the DNS IP address 2.
	LAN_0_SETTINGS_FLAG	Enable to modify the LAN 0 configuration.
	LAN_0_ENABLE_STATIC_IP	Switch static IP or DHCP for LAN port 0. `LAN_x_ENABLE_STATIC_IP=1` → static IP `LAN_x_ENABLE_STATIC_IP=0` → DHCP
	LAN_0_STATIC_IP	Set the static IP address for LAN port 0.
	LAN_0_NETMASK_BIT_COUNT	Set the network mask bit count afterwards for LAN port 0. `LAN_x_NETMASK_BIT_COUNT=8` → Netmask IP Address=255.0.0.0 `LAN_x_NETMASK_BIT_COUNT=16` → Netmask IP Address=255.255.0.0 `LAN_x_NETMASK_BIT_COUNT=24` → Netmask IP Address=255.255.255.0 `LAN_x_NETMASK_BIT_COUNT=25` → Netmask IP Address=255.255.255.128
	LAN_0_DEFAULT_GATEWAY	Set a default Gateway for LAN port 0.



```
### 3. LAN 0 Configuration ###
## 3-1. Switch static IP or DHCP
## LAN_x_ENABLE_STATIC_IP=1 -> static IP
## LAN_x_ENABLE_STATIC_IP=0 -> DHCP
LAN_0_ENABLE_STATIC_IP="0"

## 3-2. Set the static IP address
LAN_0_STATIC_IP="10.0.0.1"

## 3-3. Set the network mask bit count afterwards
## LAN_x_NETMASK_BIT_COUNT=8  -> Netmask IP Address=255.0.0.0
## LAN_x_NETMASK_BIT_COUNT=16 -> Netmask IP Address=255.255.0.0
## LAN_x_NETMASK_BIT_COUNT=24 -> Netmask IP Address=255.255.255.0
## LAN_x_NETMASK_BIT_COUNT=25 -> Netmask IP Address=255.255.255.128
LAN_0_NETMASK_BIT_COUNT="24"

## 3-4. Set the default gateway
## Warning: If you set this value, it will force LAN 0 as default gateway.
LAN_0_DEFAULT_GATEWAY=""

## 3-5. Set the DNS IP address
LAN_0_DNS_IP_1="8.8.8.8"
LAN_0_DNS_IP_2="8.8.4.4"
### ---------------------- ###
```

Item	Variable	Description
LAN 1	LAN_1_DNS_IP_1	Set the DNS IP address 1.
	LAN_1_DNS_IP_2	Set the DNS IP address 2.
	LAN_1_SETTINGS_FLAG	Enable to modify LAN 1 configuration.
	LAN_1_ENABLE_STATIC_IP	Switch static IP or DHCP for LAN port 1. `LAN_x_ENABLE_STATIC_IP=1` → static IP `LAN_x_ENABLE_STATIC_IP=0` → DHCP
	LAN_1_STATIC_IP	Set the static IP address for LAN port 1.
	LAN_1_NETMASK_BIT_COUNT	Set the network mask bit count afterwards for LAN port 1. `LAN_x_NETMASK_BIT_COUNT=8` → Netmask IP Address=255.0.0.0 `LAN_x_NETMASK_BIT_COUNT=16` → Netmask IP Address=255.255.0.0 `LAN_x_NETMASK_BIT_COUNT=24` → Netmask IP Address=255.255.255.0 `LAN_x_NETMASK_BIT_COUNT=25` → Netmask IP Address=255.255.255.128
	LAN_1_DEFAULT_GATEWAY	Set a default Gateway for LAN port 1.

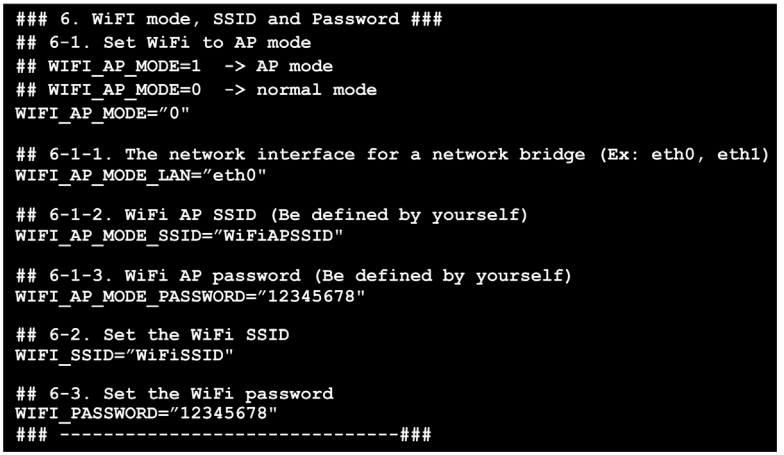
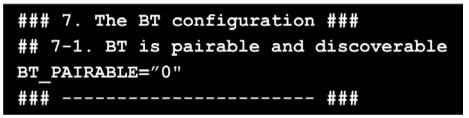
```
### 4. LAN 1 Configuration ###
## 4-1. Switch static IP or DHCP
## LAN_x_ENABLE_STATIC_IP=1 -> static IP
## LAN_x_ENABLE_STATIC_IP=0 -> DHCP
LAN_1_ENABLE_STATIC_IP="0"

## 4-2. Set the static IP address
LAN_1_STATIC_IP="10.0.1.1"

## 4-3. Set the network mask bit count afterwards
## LAN_x_NETMASK_BIT_COUNT=8  -> Netmask IP Address=255.0.0.0
## LAN_x_NETMASK_BIT_COUNT=16 -> Netmask IP Address=255.255.0.0
## LAN_x_NETMASK_BIT_COUNT=24 -> Netmask IP Address=255.255.255.0
## LAN_x_NETMASK_BIT_COUNT=25 -> Netmask IP Address=255.255.255.128
LAN_1_NETMASK_BIT_COUNT="24"

## 4-4. Set the default gateway
## Warning: If you set this value, it will force LAN 1 as default gateway.
LAN_1_DEFAULT_GATEWAY=""

## 4-5. Set the DNS IP address
LAN_1_DNS_IP_1="8.8.8.8"
LAN_1_DNS_IP_2="8.8.4.4"
### ---------------------- ###
```

Item	Variable	Description
Wi-Fi	WIFI_SETITINGS_FLAG	Enable to modify Wi-Fi configuration and connect Wi-Fi.
	WIFI_AP_MODE	Enable Wi-Fi AP mode. `WIFI_AP_MODE=1` → AP mode `WIFI_AP_MODE=0` → Normal mode
	WIFI_AP_MODE_LAN	The network interface for a network bridge (For example, eth0, eth1).
	WIFI_AP_MODE_SSID	Wi-Fi AP SSID.
	WIFI_AP_MODE_PASSWORD	Wi-Fi AP password.
	WIFI_SSID	Set the Wi-Fi SSID.
	WIFI_PASSWORD	Set the Wi-Fi password.
		
Bluetooth	BT_SETTINGS_FLAG	Enable to modify BT configuration.
	BT_PAIRABLE	BT is pairable and discoverable.
		

```
### 6. WiFI mode, SSID and Password ###
## 6-1. Set WiFi to AP mode
## WIFI_AP_MODE=1  -> AP mode
## WIFI_AP_MODE=0  -> normal mode
WIFI_AP_MODE="0"

## 6-1-1. The network interface for a network bridge (Ex: eth0, eth1)
WIFI_AP_MODE_LAN="eth0"

## 6-1-2. WiFi AP SSID (Be defined by yourself)
WIFI_AP_MODE_SSID="WiFiAPSSID"

## 6-1-3. WiFi AP password (Be defined by yourself)
WIFI_AP_MODE_PASSWORD="12345678"

## 6-2. Set the WiFi SSID
WIFI_SSID="WiFiSSID"

## 6-3. Set the WiFi password
WIFI_PASSWORD="12345678"
### ------------------------------###
```

```
### 7. The BT configuration ###
## 7-1. BT is pairable and discoverable
BT_PAIRABLE="0"
### ---------------------- ###
```

Item	Variable	Description
Node-RED	NODERED_SETTINGS_FLAG	Enable to modify Node-RED configuration.
	NODE_RED_AUTORUN	Run Node-RED automatically.
	`` ### 8. Node-RED Configuration ### ## 8-1. Run Node-RED automatically NODE_RED_AUTORUN="1" ### ------------------------ ### ```	
TPM	TPM_INIT_FLAG	Enable to initialize the TPM 2.0 module.
4 G	W4G_SETTINGS_FLAG	Enable to modify 4 G configuration.
	W4G_SIM_PIN	The SIM pin code.
	W4G_APN	The access point name.
	W4G_USERNAME	The user name for carrier.
	W4G_PASSWORD	The password for carrier.
	`` ### Plus 1. The 4G configuration ### ## Plus 1-1. The SIM PIN code (optional) W4G_SIM_PIN="0000" ## Plus 1-2. The Access Point Name (optional) W4G_APN="`  ## Plus 1-3. The username for carrier (optional) W4G_USERNAME="` ## Plus 1-4. The password for carrier (optional) W4G_PASSWORD="" ### ------------------------------### ```	

Using the Utility on the Target Device

General Information

This section gives information about how to use utility on the target device. The peripheral settings can be changed during the runtime.

Utility List

The following table describes the Utility functions:

Item	File name	Description	Path examples
COM	`com_mode_change.sh`	Change COM mode.	`~/utility/com/com_mode_change.sh`
Bluetooth	`bt_setup.sh`	Initialize the BT module and pair to a specific device.	`~/utility/bt/bt_setup.sh`
	`bt_send.sh`	Send a file to a specific remote BT device.	`~/utility/bt/bt_send.sh`
TPM	`rsa_encrypt_files.sh`	Encrypt a file with **RSA** key.	`~/utility/tpm/rsa_encrypt_files.sh`
	`rsa_decrypt_files.sh`	Decrypt a file with **RSA** key.	`~/utility/tpm/rsa_decrypt_files.sh`
4G	`w4g_setup.sh`	Initialize the 4G module and connect the 4G module to the base station.	`~/utility/w4g/w4g_setup.sh <simPin> <apn> <username> <password>`

Bluetooth Utility

The usage and examples for utilities are as follows:

Utilities	Usage	Examples
bt_setup.sh	`~/utility/bt/bt_setup.sh start|stop|list|paired|pair <macAddress>`	Let BT device is pairable and discoverable: `~/utility/bt/bt_setup.sh start` List the discovered remote BT device: `~/utility/bt/bt_setup.sh list` List the paired remote BT device: `~/utility/bt/bt_setup.sh paired` Pair to a specific remote BT device: `~/utility/bt/bt_setup.sh pair 01:02:03:04:05:06` Disable BT device: `~/utility/bt/bt_setup.sh stop`
	Parameter: `<macAddress>`: The remote BT MAC address	
bt_send.sh	`~/utility/bt/bt_send.sh <macAddress> <filePath>`	Send a file to a specific BT device: `~/utility/bt/bt_send.sh 01:02:03:04:05:06 ~/utility/bt/README.txt`
	Parameter: • `<macAddress>`: The remote BT MAC address • `<filePath>`: The file path	

TPM Utility

The usage and examples for utilities are as follows:

Utilities	Usage	Examples
rsa_encrypt_files.sh	`~/utility/tpm/rsa_encrypt_files.sh <inFile> <outFile>`	Encrypt a file: `~/utility/tpm/rsa_encrypt_files.sh test.txt` Encrypt a file with specific output name: `~/utility/tpm/rsa_encrypt_files.sh test.txt en_test.txt`
	Parameter: • `<inFile>`: Input file path, the original data • `<outFile>`: Output file path, the encrypted data (optional)	
rsa_decrypt_files.sh	`~/utility/tpm/rsa_decrypt_files.sh <inFile> <outFile>`	Decrypt a file: `~/utility/tpm/rsa_decrypt_files.sh en_test.txt` Decrypt a file with specific output name: `~/utility/tpm/rsa_decrypt_files.sh en_test.txt de_en_test.txt`
	Parameter: • `<inFile>`: Input file path, the encrypted data • `<outFile>`: Output file path, the original data (optional)	

4G Utility

The usage and examples for utilities are as follows:

Utilities	Usage	Examples
w4g_setup.sh	`~/utility/w4g/w4g_setup.sh <simPin> <apn> <username> <password>`	Unlock SIM pin, create PDP context with APN, and connect to the base station: `~/utility/w4g/w4g_setup.sh "0000" "internet"` Unlock SIM pin, create PDP context with APN, username, password and connect to the base station: `~/utility/w4g/w4g_setup.sh "0000" "lte-d.ocn.ne.jp" "mobileid@ocn" "mobile"` Overwrite APN, username, password and reconnect to the base station: `~/utility/w4g/w4g_setup.sh "" "lte-d.ocn.ne.jp" "mobileid@ocn" "mobile"`
	Parameter: • `<simPin>`: The SIM pin code (optional) • `<apn>`: The access point name (optional) • `<username>`: The username for carrier(optional) • `<password>`: The password for carrier (optional)	

COM Utility

The usage and examples for utilities are as follows:

Utilities	Usage	Examples
com_mode_change.sh	`~/utility/com/com_mode_change.sh <mode>` `<mode>`: The COM mode Parameter: ● 1: RS-232 ● 2: RS-422 ● 3: RS-485	Change COM mode to RS-232: `~/utility/com/com_mode_change.sh 1` Change COM mode to RS-422: `~/utility/com/com_mode_change.sh 2` Change COM mode to RS-485: `~/utility/com/com_mode_change.sh 3`

Chapter 8
IIoT and Cyber Security

Subject of This Chapter

This chapter describes the IIoT and Cyber Security features of the Box iPC.

What Is in This Chapter?

This chapter contains the following topics:

Cyber Security

Overview

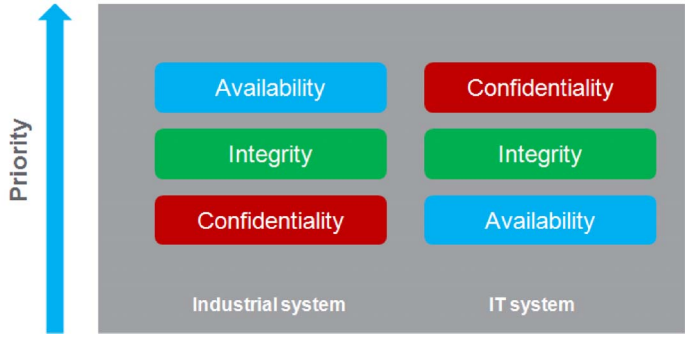
It is a fact that Industrial and control systems are more and more vulnerable to cyber attacks due to their modern design:

- They use commercial technologies.
- They are more and more connected.
- They can be remotely accessible.
- Their strategic location in the industrial processes is a point of interest for hackers.

Industrial systems have also different cyber security objectives compared to typical IT systems.To secure properly the industrial installation, it is important to understand these differences. Three fundamental characteristics have to be considered:

- Availability of the system: how to ensure that the system remains operational?
- Integrity of the data: how to maintain the integrity of information?
- Confidentiality: how to avoid information disclosure?

The priorities between an industrial system and a typical IT system are not the same as described on the following diagrams:



A good recommendation to address these security objectives is to adopt a defense-in-depth approach matching these priorities.

The IIoT Box provides a defense-in-depth approach by default, thanks to the different security mechanisms it contains.

The Magelis Box iPC enhanced cyber security to access, communicate, and store information:

IoT Box Defense-in-depth approach

To keep the system as secured as possible, it is necessary to secure the environment where the Box is installed by following the standard recommendations described below.

Cybersecurity Support Portal: *http://www.schneider-electric.com/b2b/en/support/cybersecurity/overview.jsp*

General Practices

Unauthorized persons may gain access to the Magelis Industrial PC and IIoT Box as well as to other devices on the network/fieldbus of the machine and connected networks via insufficiently secure access to software and networks.

To avoid unauthorized access to the Magelis Industrial PC and IIoT Box, users are advised to:

- Perform a hazard and risk analysis that considers all hazards resulting from access to (and operation on) the network/fieldbus, and develop a cyber security plan so.
- Verify that the hardware and software infrastructure that the Magelis Industrial PC and IIoT Box is integrated into (along with all organizational measures and rules covering access to the infrastructure) consider the results of the hazard and risk analysis, and are implemented according to best practices and standards such as ISA/IEC 62443.
- Verify the effectiveness of the IT security and cyber security systems using appropriate, proven methods.
- Keep your system up to date (security patches).
- Keep your antivirus up to date.
- Define properly the security of the Box: access rights, user's accounts. Ensure that the minimum access rights are given to users to avoid illegal access or too much privilege given to the user.
- Enable data encryption (available by default or as option depending on Part Numbers).
- Limit the access to the only needed information and users.
- Follow the recommendations to secure the Network infrastructure (see **General Practices** chapter in the document **How Can I Reduce Vulnerability to Cyber Attacks in PlantStruxure Architectures?** (*http://www.schneider-electric.com/b2b/en/support/cybersecurity/resources.jsp?*)

Cyber Security Features Available

Cyber security features available on Magelis Industrial PC and IIoT Box:

1. IIoT Box architecture is based on the operating system.
2. Hardware can include a TPM module used for security enforcement.
3. Integrity of the operating system is also checked by RISC (Reduced Instruction Set Computer) mechanism that ensures that the OS is the official one.

NOTE: Taking into account the large number of various configurations and applications, convenient and efficient out of the box settings for the Box PC IIoT cannot be provided. It belongs to authorized person in charge of commissioning and configuration to enable or disable functions and interfaces according to cyber security requirements for the applications.

Recommendations For Node-RED

Node-RED can be configured from several channels:

1. Using a connection to IIoT Box Node-RED server from another computer in the network.
2. By importing a JSON file in the IIoT Box using a media or network access.
3. Using Web services from the Node-RED server from an application.

NOTE: What ever the scenario, the user must be sure that the computer used to access the IIoT Box is safe: OS up to date, security patches up to date, antivirus up to date, no malware on the PC.

When importing a JSON file using removable media like USB key must be done very carefully to avoid importation of corrupted JSON files or malware on the IIoT Box. The operation should be reserved to people authorized to modify the configuration of the IIoT Box.

NOTE: A configuration of the IIoT Box has a deep impact on the overall security architecture. All modification done in the box configuration can lead to device access or cloud access by unauthorized users.

The configuration of the IIoT Box is done thanks to Node-RED configuration with the Node-RED server. The system is provided with an existing set of nodes.

However, for specific needs (specific device access, specific cloud access, specific data management) the user may need new functionalities. This is given by the ability to create new Nodes.

NOTE: Creation of new nodes also implies the increase of the attack surface that could lead to an unsecure system.

A Node-RED designer should be aware of the following recommendations to keep the security of the system at the expected level:

- Recommendation 1: Node-RED designers should apply well-known good practices of software engineering to ensure a good quality level and avoid typical mistakes like buffer overflow, bad exception management.
- Recommendation 2: All data coming/going from the devices and more generally all data injected in Node-RED modules should be checked and validated to avoid typical errors like buffer overflow, data injection (see OWASP recommendations for typical errors). Communication errors with devices should also be handled properly to avoid deny of services of the system.
- Recommendation 3: All data coming/going from IT services (like cloud for instance) should be properly checked and validated to avoid information disclosure, deny of services and typical security issues.

IIoT and Node-RED

Overview

The Industrial Internet of Things (IIoT) is the use of Internet of Things (IoT) technologies in manufacturing. The IoT is a network of intelligent computers, devices, and objects that collect and share huge amounts of data. The collected data is sent to Cloud-based service where it is shared with users in a helpful way.

The IIoT works not only at the machine or process level, but from the device itself to be seamlessly wired to the business systems and Internet data levels. It is a parallel application model, connecting edge to cloud computing: Collecting data from agent.enabled edge devices, connected to field devices, and improving operations and asset performance with cloud applications.

The IIoT runs analytics in the agents, preferably the field device itself, or an edge device connected to the field devices, interfacing with the automation application. The analytics are built and deployed over time without the need to modify or even shut down the existing control system.

The IIoT consolidates analytics across a fleet of heterogeneous assets, in disparate geographies. It aggregates data and seamlessly provides analytics at the cloud level, building the digitalized smart factories and improving responsiveness.

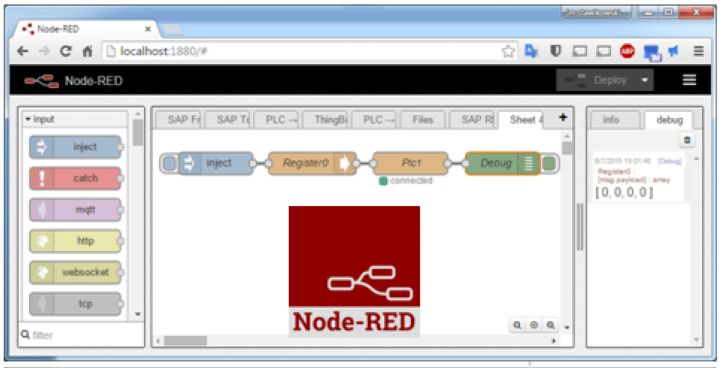
Node-RED

Node-RED leverages IT/OT convergence. It is the new software technology to wire the **things** from the field to the Internet IT and cloud applications without the need to modify existing systems. It is the quick path to the IIoT. Node-RED is light, open source, and simple to use. An existing transparent Ethernet TCP/IP network is used with Node-RED.

Node-RED is composed of an editor tool and an engine to make easily and run the connections between the IIoT applications. Any **things** can be connected with Node-RED over the IIoT, including all automation devices with processing capabilities and Ethernet TCP/IP connections. Even the smallest field devices without such capabilities can be wired with Node-RED thanks to intermediary edge devices that collect data.

Node-RED is the visual tool for wiring the Internet of Things. The Box iPC Nodes are delivered with IIoT package. Any nodes from the Node-RED community can also be used, to “wire” together hardware devices, APIs, and online services in new ways, leveraging Internet of Things and Enterprise 4.0 approaches. It builds the infrastructure for new digitalized services.

Node-RED editor is accessible with Web browser:



The Box iPC can be upgraded with an IIoT featuring Node-RED. Nodes to monitor and control devices are delivered with the package (iPC internal temperatures, storage disk status, power supply status, SMS/email alerts, device recovery, and so on). Open, any of the thousands of nodes available from the Node-RED community can also be added to **[wire]** together hardware devices, APIs, and online services.

Cybersecurity for the IIoT

Cybersecurity has become a challenge to implementing the IIoT. Using standard network means benefitting from the security measures already provided by your IT system, such as firewalls, VPNs, and safe zones.

NOTE: The devices with Node-RED can be set to make only **[output]** communication. The cloud applications have no **[input]** communication request to the Node-RED devices. Node-RED devices push data to the cloud. So communications to the machine and plant levels are not necessary and should be avoided to guard against attacks.

NOTE: Schneider Electric adheres to industry best practices in the development and implementation of control systems. This includes a "Defense-in-Depth" approach to secure an Industrial Control System. This approach places the controllers behind one or more firewalls to restrict access to authorized personnel and protocols only.

⚠ WARNING

UNAUTHENTICATED ACCESS AND SUBSEQUENT UNAUTHORIZED MACHINE OPERATION

- Evaluate whether your environment or your machines are connected to your critical infrastructure and, if so, take appropriate steps in terms of prevention, based on Defense-in-Depth, before connecting the automation system to any network.
- Limit the number of devices connected to a network to the minimum necessary.
- Isolate your industrial network from other networks inside your company.
- Protect any network against unintended access by using firewalls, VPN, or other, proven security measures.
- Monitor activities within your systems.
- Prevent subject devices from direct access or direct link by unauthorized parties or unauthenticated actions.
- Prepare a recovery plan including backup of your system and process information.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Chapter 9
Maintenance

Subject of this Chapter

This chapter covers maintenance of the Box iPC.

What Is in This Chapter?

This chapter contains the following topics:

Reinstallation Procedure

Introduction

In certain cases, it may be necessary to reinstall the operating system.

Precautions to take:

- Keep static-producing materials (plastic, upholstery, carpeting) out of the immediate workpace.
- Do not remove ESD-sensitive components from their anti-static bags until you are ready to install them.
- When handling static-sensitive components, wear a properly grounded wrist strap (or equivalent).
- Avoid contact with exposed conductors and component leads.

Before Reinstallation

Hardware required:

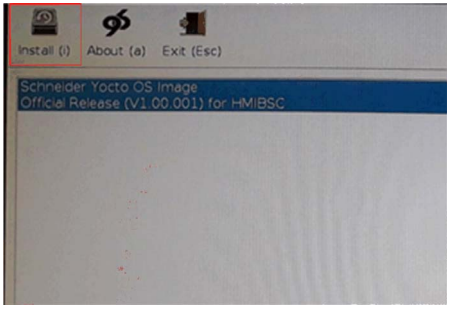
- Recovery media.

Setting up the hardware:

- Shut down the operating system in an orderly fashion and remove all power from the device.
- Disconnect all external peripherals.

NOTE: Save all main data onto a hard drive or a memory card. The reinstallation process returns the computer to its factory settings and erases all data.

Installing OS Image from SD Card

Step	Action
1	Plug in the recovery SD card into the board and restart it.
2	Choose the displayed Operating system (Yocto Linux) and click **Install**. This flashes the OS on the board eMMC: 
3	Once you see the programming successful dialog, unplug the power cord.
4	Remove the SD card and then plug in the power cord. The system restart into your chosen Operating System.

Regular Cleaning and Maintenance

Introduction

Inspect the Box iPC periodically to determine its general condition. For example:

- Are all power cords and cables connected properly? Have any become loose?
- Are all installation screws holding the unit securely?
- Is the ambient temperature within the specified range?

The following sections describe maintenance procedures for the Box iPC, which can be carried out by a trained, qualified user.

⚠⚠ DANGER

HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH

- Remove all power from the device before removing any covers or elements of the system, and prior to installing or removing any accessories, hardware, or cables.
- Unplug the power cable from both the Magelis Industrial PC and the power supply.
- Always use a properly rated voltage sensing device to confirm that power is off.
- Replace and secure all covers or elements of the system before applying power to the unit.
- Use only the specified voltage when operating the Magelis Industrial PC. The AC unit is designed to use 100...240 Vac input. The DC unit is designed to use 24 Vdc input. Always check whether your device is AC or DC powered before applying power.

Failure to follow these instructions will result in death or serious injury.

During operation, the surface temperature of the heat sink may exceed 70 °C (158 °F).

⚠ WARNING

RISK OF BURNS

Do not touch the surface of the heat sink during operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Cleaning Solutions

⚠ CAUTION

HARMFUL CLEANING SOLUTIONS

- Do not clean the unit or any component of the unit with paint thinner, organic solvents, or strong acids.
- Use only a mild soap or detergent that will not harm the poly carbonate material of the screen.

Failure to follow these instructions can result in injury or equipment damage.

Appendices

Appendix A
Accessories

Accessories for the Box iPC

Available Accessories

Accessories are available as options. The table shows the list of accessories available for the Box iPC:

Reference	Description
Interfaces	
HMIYBIN2AIM21	Interface M.2 2 x analog input 0-10 V/4-20 mA
HMIYMIN8AI1	Interface mini PCIe 8 x analog input 0-10 V
HMIYMIN4GEU1	Cellular 4G EU/Asia
HMIYMIN4GUS1	Cellular 4G US
HMIYMINWIFI2	Interface WiFi access point and 2 x antennas
HMIYCABWIFIAN511	Antenna WiFi/Bluetooth
HMIYBINLTPM201	Module TPM
Drives	
HMIYSD016C1	SD Card industrial grade 16 GB
HMIYSD064C1	SD Card industrial grade 64 GB
Accessories	
HMIYMMAC1	AC power supply module 100 W
HMIYPSOMAC1	AC power supply module 60 W
HMIYMUPSKT1	UPS battery
HMIYCABUPS31	UPS 3 m (9.84 ft) cable
HMIYBMKTBSC1	Maintenance kit
HMIYADBMODIN11	DIN rail adaptor
HMIYCAB4GAN51	5 m Cable for 4G card

Index