Symphony IoT – Raspberry Pi Deployment Guide

Digital Output Sensors
Limitations
Testing
Raspberry Pi Device Connection

RS485 Sensors
Limitations
Testing
Check the Sensor with MODSCAN
Raspberry Pi Device Connection

Data Acquisition
Maintenance and Troubleshooting
For MAC and UBUNTU System
For Windows System
Frequently Asked Questions
Finding Product Documentation

Zoho Corp.
Introduction

This document guides you through deploying and configuring the sensors at the remote locations. These sensors fetch the data from the remote location and with the help of the gateway device; they transfer the data to the WebNMS server.

Contacting WebNMS Support

WebNMS support provides assistance with installing the sensors, configuring the data points, answers FAQs, and helps the user to troubleshoot any problems. You can also e-mail your queries related to sensor deployment and configuration to iot-support@webnms.com.

Pre-Requisites

This guide has been written with an assumption that the user has a basic knowledge on the sensors and is well versed with the basics of electronics.

- Relevant sensors as per your requirement.
- WebNMS IoT platform.
IoT, Sensor, and Cloud server

- **Internet of things (IoT)**
- **Sensor deployment**
- **Communication between the sensor and cloud server**

### Internet of Things (IoT)

Internet of things widely known as IoT is a network of physical objects like devices, vehicles, buildings, etc. embedded with electronics, software, sensors, actuators, and network connectivity that allows the exchange of huge volume of data either through wired or wireless means. How this internet of things work is really simple, this is the concept of basically connecting devices to the Internet and let them communicate with other devices, users etc. This IoT structure basically involves sensors, a gateway device, and the cloud server over which the communication takes place. All the devices communicating in IoT network will talk to each other using the same Internet Protocol (IP).

![Figure 1 Communication between sensors, gateway, and cloud server](image-url)
Sensor Deployment

Sensors are deployed at the remote location so that they can detect any changes in the environment and notify the user for the occurred changes. These changes are then captured and pushed to the cloud server. Multi-purpose EdgeX agent bridges the gap between the sensors installed at the remote location and the cloud server. The agent when pushed to the gateway device ensures that the communication establishes between the sensors and the cloud server. The user can then use the cloud server to get the periodical data from the sensors.

Communication between Sensor and Cloud Server

A sensor is fixed at the selected remote location and is connected to the gateway device. Communication establishes through the gateway device in either WAN mode or in the Ethernet mode. WAN mode helps the gateway device to interact directly with the internet whereas; Ethernet mode helps the gateway device to interact with the locally installed devices. Once the communication mode is set and the cloud server is configured, the communication initiates between the sensors and the cloud server. The established communication allows you to monitor the data from any location at your ease.
System Requirements

- **Hardware Requirements**
- **Software Requirements**
  - Windows System
  - UBUNTU system
  - MAC System

Hardware Requirements

The following are the minimum hardware requirements for working with any Raspberry Pi device:

- A Raspberry Pi 2 model B device or Raspberry Pi 3 model B
- 8Gb class 4 SD card
- A USB Micro power supply (like most standard mobile phone chargers)
- A good-quality power supply that can supply at least 2A at 5V for the Model 3B, or 700mA at 5V for the earlier, lower powered models. Low current (~700mA) power supply
- A memory card reader
- Ethernet for connectivity
- HDMI monitor to find the IP address of the R-Pi device
- A R-Pi cover case with cooling fan
- RS485 to USB converter for RS485 sensors
- 40 pin FRC cables or female header pins for connecting DI/DO

Software Requirements

The following are the minimum software requirements for working with any Raspberry Pi device:

- A RPI image with EdgeX agent based on Raspberry Pi models
- IP scanning software
- WebNMS IoT platform
Windows System

Win 32 disk imager software is required to burn the RPI image onto the SD card.

UBUNTU System

Disk software is required to burn the RPI image onto the SD card.

MAC System

Apple Bake off or Burn it software is required to burn the RPI image onto the SD card.
Prepare the Device

- Prepare the SD Card
- Install the SD Card
- Find the IP Address
  - Using HDMI Cable
  - Using IP Scanning Software
    - Windows Platform
    - Linux Platform
      - Using ARP Scan
      - Using Nmap Scan
- Connect the Device to the Network
- Upload the Json File

Prepare the SD Card

E-mail to iot-eval@webnms.com and request the RPI image containing the EdgeX agent. Burn the RPI image containing the EdgeX agent onto the SD card using the below mentioned procedures.

Windows System

To burn the image onto the SD card in a Windows system, use Win 32 disk imager software. Follow the steps to burn the image:

1. Double click and open the Win32 disk imager software.
2. Click on the device icon and select the image. Click Write. The image will get burnt onto the SD card.
Figure 2 Win32 Disk Imager

MAC System

Use an Apple Bake off or Burn it to burn the image on the SD card.

UBUNTU System

To burn the image onto the SD card in a UBUNTU system, follow the steps given below:

1. Put the SD card via a card holder to the system.
2. Format the SD card.
3. Click on Restore disk image and a pop-up will appear asking the location of the image. Select the location of the image and click Open. It will start restoring the image onto the SD card.
Figure 3 Disk Imager for UBUNTU
Install the SD Card

- Once the RPI image is burnt onto the SD card, install the SD card in the SD card slot on the Raspberry Pi device.
Switch on the Raspberry Pi device.

Find the IP Address

Using HDMI Cable

1. Connect the keyboard and the monitor to the Raspberry Pi device by the USB and the HDMI cable respectively.
2. Turn on the Raspberry Pi device using the username as 'pi' and password as 'raspberry'.
3. Open the terminal on the monitor and give the command 'ifconfig' and press Enter.
4. IP address of the Raspberry Pi device will get listed on the screen.
Using IP Scanning Software

Windows Platform

1. Double click on the Advanced IP Scanner software icon.
2. Click on Scan in the pop-up window.
3. The Advanced IP Software will show the IP address of all the devices on the network.

Zoho Corp.
LINUX Platform

Using ARP Scan

1. Open the terminal.
2. Give the command `sudo arp-scan --localnet --interface=eth0` and enter the password.
3. Three IP addresses will appear on the screen. The extra IP address is the IP address of the Raspberry Pi device connected.
Using Nmap Scan

1. Open the terminal.
2. Give the command `nmap -sn 192.168.0.0/24` (this is the IP address of the machine) and enter password.
3. Three IP address will appear on the screen. The extra IP address is the IP address of the Raspberry Pi device connected.
Figure 9 Example screen showing Nmap scan running
Connect the Device to the Network

**Raspberry Pi 2 model B** – The device can be directly connected via an Ethernet cable to the RJ45 socket in the device.

**Raspberry Pi 3 model B** – The device can be connected either via an Ethernet cable to the RJ45 socket or via Wi-Fi network. To connect the device to a Wi-Fi network, connect the R-Pi device to a monitor and a keyboard. Follow the steps as given below.

- Open the terminal in the system.
- Login to the Raspberry Pi device remotely by giving the command `ssh pi@IP address of the device` and press Enter.
- Enter the password as ‘raspberry’ to continue.
- Give the command `sudo iwlist wlan0 scan` and press Enter. All the Wi-Fi networks available in the area where the device is available will be visible on the screen.
- Check the Wi-Fi network to which the Raspberry Pi device is to be connected. The Wi-Fi network for the device will be displayed next to ‘Essid’.
- If the network is available, open the `wpa-supplicant` configuration file in nano text editor. Type `sudo nano /etc/wpa_supplicant/wpa_supplicant.conf`, to open the Wi-Fi configuration page and go to the bottom of the page.
- Give the command,
  
  ```
  network=
  ssid="The_ESSID_from_earlier"
  psk="Your_wifi_password"
  }
  ```
- In case if the above command does not respond, give the command as `sudo nano /etc/network/interfaces`.
- Press Ctrl+X and save the file.
- Reboot the Wi-Fi module on the Raspberry Pi device by giving the below two commands.
  - Give the command `sudo ifdown wlan0` to switch off the Wi-Fi module.
  - Give the command `sudo ifup wlan0` to switch on the Wi-Fi module.
- After rebooting the Wi-Fi module of the Raspberry Pi device, check the IP address of the Wi-Fi network to which the device is connected. Give the command `ifconfig` and the IP address will be listed against `wlan0`.
- Turn on the Raspberry Pi device for all modes.
**Note:** Reboot the entire system by giving the command, `sudo reboot` and check if the device is connected on startup.
Upload the Json file

1. Download the Json file after doing the initial configuration in the WebNMS server. To know more on how to do the initial configuration, refer to User_Admin Guide.

2. Open the browser in the system and type \texttt{http://IPaddressofRPI:5000} in the URL. Page to upload the Json file will appear on the screen.

3. Click \textbf{Browse} to go to location of the downloaded Json file.

4. Select the file from the location and click \textbf{Upload}.

![Figure 10 Example screen for agent upload](image-url)
Pin Configuration

Figure 11 Pin configurations on Raspberry Pi device
Compatible Sensors

- **Digital Sensors**
  - Digital Input Sensors
  - Digital Output Sensors
- **RS485 sensors**

Digital Sensors

Digital sensors are the sensors in which the signal is directly converted into the digital signal output. The signal is usually measured in two states in digital sensors i.e. on state and off state. Typical example of a digital sensor can be a door sensor, which gives the output in two states either normally closed (NC) when the door is open or normally open (NO) when the door is closed. Some examples of the digital sensors are the door sensor (magnetic), the LED lights, alarms, the push button switches etc.

![Digital Sensors Configuration](image)

Figure 12 Digital sensors configuration
Digital Input Sensors

The compatible digital input sensors are the ones with the voltage limit between 0V to 3.3V.

Digital Output Sensors

The compatible digital output sensors are the ones with the maximum voltage at 3.3V.

RS485 Sensors

RS485 sensors are the sensors that transmit signal over two lines rather than on single ended with a voltage referenced to the ground. The transmission is done via twisted pair of cables. A common configuration for the RS485 sensors is the bus network i.e. multiple sensors can be attached to a single cable over a long range of distance. Each sensor is given an individual id. This means that multiple sensors will only use up one port on the data logger, saving considerably on cable and data logging costs. At the moment, the compatible RS485 sensors with the Raspberry Pi device are the ones which can follow the MODBUS protocol. Examples of RS485 sensors are the energy meters, the humidity sensors, and the temperature sensors etc.
Protocols Supported

- MODBUS
  - MODBUS RTU
  - MODBUS TCP

MODBUS

At the moment, the only protocol supported by the Raspberry Pi device is MODBUS protocol. MODBUS protocol follows the master-slave arrangement in which each device connected will communicate using a unique address. Being said that, one node which is assigned as the master node will initiate communication to one of the slave nodes and will have the unique address of that slave node. Only the intended slave node will respond back to the communication initiated by the master node. Other nodes might receive the data but they won’t acknowledge the received data.

MODBUS has different types of the variants like MODBUS RTU and MODBUS TCP.

MODBUS RTU - MODBUS RTU is an open, serial protocol derived from the Master/Slave architecture. It is a widely accepted protocol due to its ease of use and reliability.

MODBUS TCP - MODBUS TCP/IP (also MODBUS-TCP) is simply the MODBUS RTU protocol with a TCP interface that runs on Ethernet. The MODBUS messaging structure is the application protocol that defines the rules for organizing and interpreting the data independent of the data transmission medium.
Digital Input Sensors

- **Limitations**
- **Testing**
- **Raspberry Pi Device Connection**
  - **Pull Up State**
  - **Pull Down State**

**Limitations**

The only limitation with the digital input sensors is the voltage limits of 0V to 3.3V.

**Testing**

The digital input sensor is to be tested before connecting it to the Raspberry Pi device. To test the digital input sensor, measure the voltage using a multi-meter.

- If the multi-meter shows 0V or max voltage as suggested by the manufacturer of the sensor, it can be connected to the Raspberry Pi device.
- If the multi-meter does not show any reading, the digital input sensor is damaged and it must be replaced.

**Raspberry Pi Device Connection**

**Pull Up State**

In pull up state of the digital input sensors, one wire from the sensor will be connected to one of the default pull up pins i.e. 2, 3, 4, 5, 6, 7, and 8 and the other wire will be connected to one of the ground pin i.e. 9, 6, 14, 20, 25, 30, 34, and 3.

**Pull Down State**

In pull down state of the digital input sensors, one wire from the sensor will be connected to one of the default pull down pins i.e. 9, 10, 11, 12, 13, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, and 27 and the other wire will be connected to the pins with 3.3V i.e. **pin 1 and pin 17.**
Digital Output Sensors

- Limitations
- Testing
- Raspberry Pi Device Connection

Limitations

While using the digital output sensors with values of 3.3V or an AC voltage, relays should be used.

Testing

A digital output sensor is to be tested before connecting to the Raspberry Pi device. To test the digital output sensor, inject a 3.3V power supply to the sensor.

If you get the output value, the sensor can be connected to the Raspberry Pi device.

If you don't get any output value, the digital output sensor is damaged and should be replaced.

Raspberry Pi Device Connection

Any pin can be configured as digital output in the Raspberry Pi device. One wire will be connected to a relay and the other wire will be connected to any one of the ground pins i.e. 9, 6, 14, 20, 25, 30, 34, and 39.
RS485 Sensors

- **Limitations**
- **Testing**
  - Check the Sensor with MODSCAN
  - Raspberry Pi Device Connection

**Limitations**

The only limitation with RS485 sensors is the range. It can be used up to a range of 1200m only.

**Testing**

A RS485 sensor is to be tested before connecting to the Raspberry Pi device. To test a RS485 sensor, use a USB to RS485 converter. Once the connection is established, check the sensor with a MODSCAN.

**Check the Sensor with MODSCAN**

To check the sensor with MODSCAN you need to know the COM port number of the USB converter. For checking the COM port number of the USB converter, follow the procedure as given below.

- Click on the Start button and then click on the Control Panel.
- Click on the Device Manager Button. A new window will appear listing all the devices connected to your system.
Double click on the Ports (COM & LPT). A list of available serial and parallel port devices will appear on the screen. Your USB COM port should appear on the screen with the number assigned to it.

Open MODSCAN and click on the Connection tab. A new window will appear on the screen where you will have to fill details like the COM port number, the baud rate, the word length, the parity, and the stop bits and click OK.
Figure 15 MODSCAN Connection details

- Click on the Connect button to establish the connection. Once the connection is established it should show you a true value as given by the manufacturer.
- If it displays the true value, the RS485 sensor can be connected to the Raspberry Pi device.
- If it does not display the true value, the RS485 sensor is damaged and should be replaced.

Raspberry Pi Device Connection

While connecting the RS485 sensor to the Raspberry Pi device, +ve of the sensor will be connected to the D+ve of the USB converter and -ve of the sensor will be connected to the D-ve of the USB converter. The USB converter is then connected to one of the USB slots on the Raspberry Pi device.
Data Acquisition

Sensor to server communication is established once the configuration file is installed to the gateway device. The configuration of the devices and the data points are some of the parameters that will be visible on the WebNMS server dashboard. The communication between the gateway device and the cloud server is established with the multipurpose agent EdgeX that bridges the gap between the sensor, the gateway and the cloud server.

Once the communication is established with the WebNMS server, the data for any change in the environment of the sensors will be sensed by the sensors and then transferred to the WebNMS server, where it's then aggregated and analyzed. You can access the information gathered by the WebNMS server on your smartphone or web browser. Below given is the illustrative example of how the WebNMS dashboard will look like when the data transmission starts.

![WebNMS server dashboard for data acquisition](image)

Figure 16 WebNMS server dashboard for data acquisition

Zoho Corp.
Maintenance and Troubleshooting

- For MAC and UBUNTU System
- For Windows System

WebNMS support provides complete assistance for troubleshooting any errors that occurs during any phase of data acquisition. For eradicating the errors completely, WebNMS support requires the log reports which can be generated by the user and then mailed to the iot-support@webnms.com.

For MAC and UBUNTU System

Follow the procedure given below to take the logs from any MAC and UBUNTU system.

1. Connect the Raspberry Pi device one to one with the system.
2. Open the terminal.
3. Give the command `sftp pi@IP address of the device`.
4. If you are connected to the terminal, the system will ask for password. Give the default password as 'raspberry'. If you have changed the password previously, use the changed password.
5. Give the command `mget - r/mnt/M2M/logs`.
6. A folder will appear with all the log files.
7. Zip the entire folder and mail it to iot-support@webnms.com.

For Windows System

Follow the procedure given below to take the logs from any Windows system.

1. Connect the Raspberry Pi device one to one with the system.
2. Open the sftp or ftp application in the system.
3. Browse and transfer the file `/mnt/M2M/logs`.
4. A folder will appear with all the log files.
5. Zip the entire folder and mail it to iot-support@webnms.com.
Frequently Asked Questions

- **Which sensors are compatible with the device?**
  Refer to the [Compatible sensors](#) with the Raspberry Pi device for details on this section.

- **How to change the pin configuration?**
  The pin configuration on the WebNMS cannot be changed dynamically. The old mapping has to be deleted and the new mapping has to be done. It is not possible to change the pin or device name after initial configuration.

- **How to identify if the sensor is working or not?**
  Refer to testing of the sensors under each sensor topic.

- **What are the specifications of the connection cables?**
  There is no particular specification for the cables. It is preferred and convenient to use two/three core coaxial cables.
Finding Product Documentation

- Finding Product Documentation
- Feedback

Feedback

You can send your valuable feedback to iot-support@webnms.com.