

**Objective:** In this tutorial, you will learn how to complete an application using **flow-based programming**. To achieve this, you will learn how to use Node-RED, a flow-based development framework (originally developed by IBM) for visual programming with the Internet of Things (IoT). It built connections between sensors, APIs, and online services as part of the IoT.

### **What is Flow-Based Programming (FBP)?**

Flow-based programming (FBP) is a programming model that defines applications using the metaphor of a "data factory" or black boxes. Although this paradigm is not completely new, in the context of IoT it promises to revolutionize the way we build programs (e.g. object orientation or service orientation). In this paradigm, an application is defined as a network of asynchronously executing processes, which communicate by means of streams of data chunks called "information packets". FBP is a form of component-oriented programming paradigm [1].

**Required Setup:** Connect GrovePi+ board to RPi and have all GrovePi+ libraries installed.

### **Parts:**

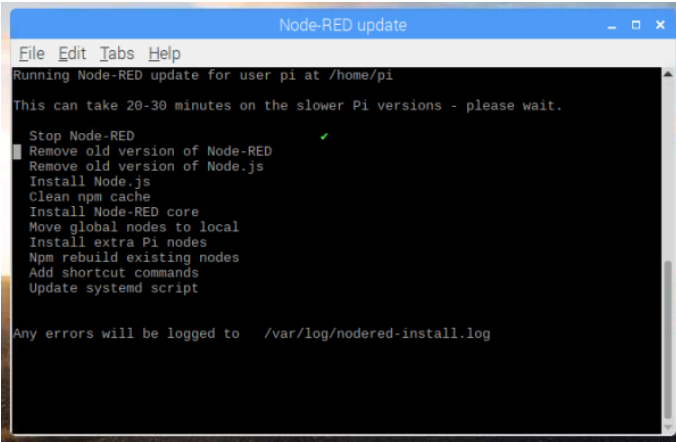
- RPi 3 B+
- GrovePi+ board
- Grove connection wires, and sensors

### **Part A: Install / Upgrade Node-RED on RPi**

**Step 1.** First, Raspbian comes with Node-RED preinstalled. Now we will install/upgrade it. Open the terminal window and enter the following command.

```
bash <(curl -sL https://raw.githubusercontent.com/node-red/raspbian-deb-package/master/resources/update-nodejs-and-nodered)
```

**Step 2.** After entering the above command, the following status will show on the terminal:



```
Node-RED update
File Edit Tabs Help
Running Node-RED update for user pi at /home/pi
This can take 20-30 minutes on the slower Pi versions - please wait.
Stop Node-RED
Remove old version of Node-RED
Remove old version of Node.js
Install Node.js
Clean npm cache
Install Node-RED core
Move global nodes to local
Install extra Pi nodes
Npm rebuild existing nodes
Add shortcut commands
Update systemd script
Any errors will be logged to /var/log/nodered-install.log
```

**Step 3.** The installation might take a while. Wait until it completes. Once completed, it will display the message "All done."

```
Any errors will be logged to /var/log/nodered-install.log
All done.
You can now start Node-RED with the command node-red-start
or using the icon under Menu / Programming / Node-RED
Then point your browser to localhost:1880 or http://{your_pi_ip-address}:1880
Started Fri May 10 13:25:26 PDT 2019 - Finished Fri May 10 13:30:46 PDT 2019
```

**Step 4.** Enter the following command:

```
update-nodejs-and-nodered
```

**Step 5.** Now, let's go to the node-red directory and rebuild node-RED by entering the following commands(one at a time):

```
cd ~/.node-red
npm rebuild
```

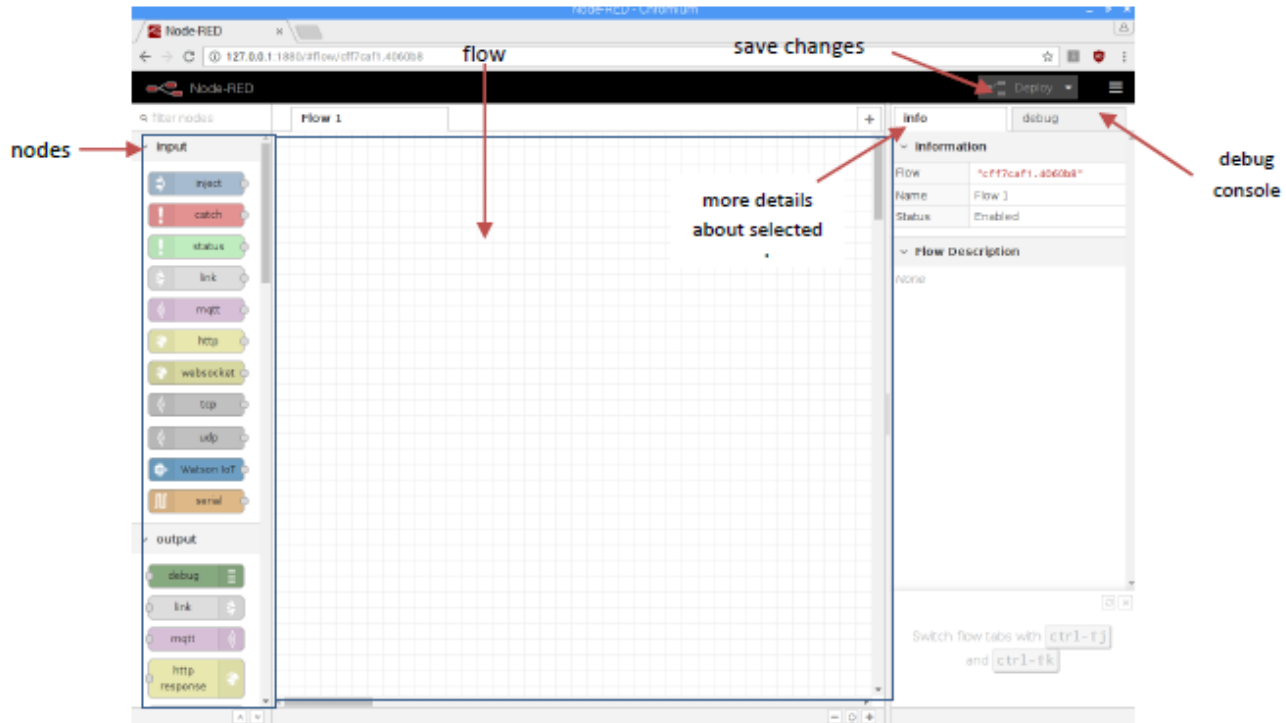
**Step 6.** Congrats! Node-red is installed/updated. Now we can open it from Start menu -> Programming.

A Terminal window will appear, and it will then initialization of Node-RED will take place. Once you see "Started flows", this indicates that Node-RED is now ready to use. To access Node-RED, we will need to use the browser. To access Node-RED via the browser, in the Terminal window, the program will provide the address where it can be accessed. Usually (if default is not modified) is **http://127.0.0.1:1880**

```
your credentials.
You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----
10 May 13:44:30 - [info] Server now running at http://127.0.0.1:1880/
10 May 13:44:30 - [info] Starting flows
10 May 13:44:30 - [info] Started flows
```


**Step 7.** Open the browser and enter the address to launch Node-RED dashboard.

Now you should see the following page. (Next page ->)

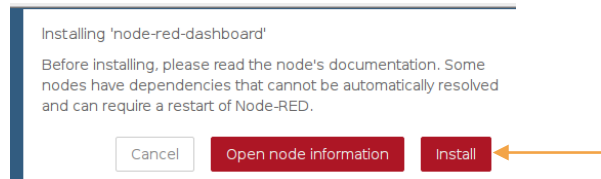


The following description is based on the official documentation of Node-RED (<https://nodered.org/>)

<i>node</i>	<p>A <b>node</b> is a building-block function that enables you to perform a specific task. Some nodes are specific in terms of the tasks they can perform while others can be programmed or altered in different ways (provides more flexibility). A node can act as (a) an <b>input</b>, (b) an <b>output</b> or (c) a <b>bridge</b>. Node-RED provides a set of basic collections of nodes on the left-hand side called Palette. However, you can always extend the palette collection by installing new nodes built by other developers. You can find a list of available nodes or collections of nodes at the Node-RED website: <a href="https://flows.nodered.org">https://flows.nodered.org</a>. You can add nodes to the</p>
<i>flow</i>	<p>A <b>flow</b> is a collection of connected nodes. These flows represent the program flow that you can execute without an actual script. This is essentially your program that consists of the flow connecting multiple nodes for performing tasks. A workspace can have a number of flows (e.g. programs or scripts). You can always edit the properties of a flow by double-clicking on its tab in the top bar.</p> <p><b>Import/export flows:</b> you can always import or export flows using their JSON format. To import a flow, open the import dialog and then paste the flow json and click Import. To export a flow, open the export diagram and it will allow you to export the entire flow in json format to clipboard.</p>
<i>debug console</i>	<p>The debug console provides a textual interface to debug nodes in a flow (i.e. log messages from the runtime). It is used to output logs from nodes or flows. It mainly used for debugging when executing flows. The console is located on the right-hand side in the sidebar section.</p>
<i>info</i>	<p>The info is located on the sidebar section and provides additional information or properties about Node-RED elements (e.g. nodes, workspace, etc.).</p>
<i>palette</i>	<p>A palette contains all of the existing nodes that are currently installed and available for integration into the workspace. A palette is organized by categories with inputs, output and functions at the top. You can expand or collapse categories of nodes. The palette manager enables you to install or remove nodes.</p>
<i>workspace</i>	<p>This is the area where the main flows are created by dragging nodes from the palette and wiring them together to create flows. The workspace is entire editor window of Node-RED.</p>
<i>sidebar</i>	<p>The sidebar contains panels for useful tools such as info, debug, configuration nodes and context data.</p>

**Step 8.** From the top menu, expand the menu by clicking the icon . Then select **"Manage Palette"**.

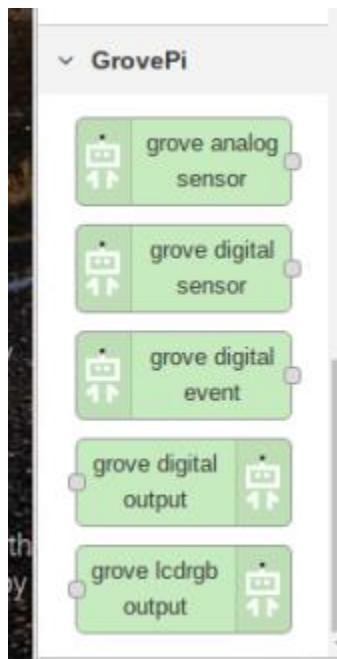
**Step 9.** Click on the **"Install"** tab and then enter the following keyword in the search area **"dashboard"**. Select the one titled **"node-red-dashboard"** (you may need to scroll down) and click install (to the right). A pop-up window will appear titled "Installing 'node-red-dashboard'". Click install.



**Step 10.** Once installed, it will list the nodes that have been added into Node-RED. The Dashboard enables you to create HTML objects easily on a webpage as well as dashboard elements such as gauge, slider, picker, etc.

**Step 11.** Again, open the **Manage Palette** window and enter the following keyword in the search area **"node-red-contrib-grovepi(v 0.1.8)"**. Click on "install" and select "install" from the pop-up window. Give it a while to install.

**Step 12.** Close Manage Palette. Check the palette section on the left, scroll down, you should see a section titled GrovePi and as shown below:



**Congratulations!** You have now completed installing the Node-RED and configured the GrovePi module properly. You are now ready to complete the first demo.

**References:**

1. J. Paul Morrison, Flow-Based Programming, 2nd Edition: A New Approach to Application Development, CreateSpace, 2010.
2. Node-RED (<https://nodered.org/>)