

CEC

Continuing
Education
Center

DesignNews

Hands-On with Industry 4.0 using the Raspberry Pi and the Arduino Platforms

DAY 4: Build a Raspberry Pi Node-RED Dashboard: Part 1-DC Motor Controller

Sponsored by



Webinar Logistics

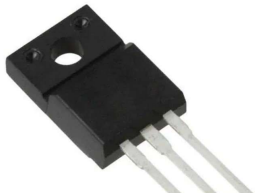
- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Attendee Chat’ by maximizing the chat widget in your dock.



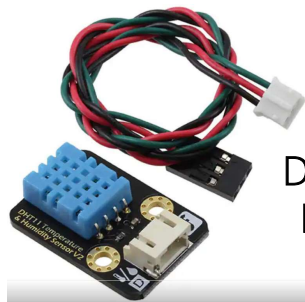
Dr. Don Wilcher

Visit 'Lecturer Profile' in your console for more details.

IRFS630A N-Channel PMOSFET



Axial DC Fan, 5VDC



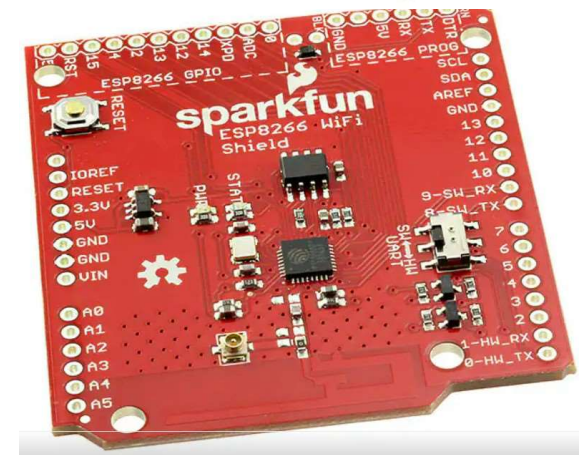
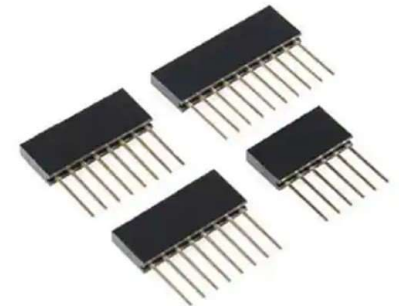
DHT Temperature-
Humidity Sensor

Course Kit and Materials



Osoyoo WiFi Internet of Things
Learning Kit

Arduino Stackable Header Kit



ESP8266 WiFi Shield

Agenda:

- Cyber-Physical Systems Model:
- Node-RED: An Overview
- Node-RED Widgets
- Lab: Node-RED DC Fan Controller



Industry 4.0:



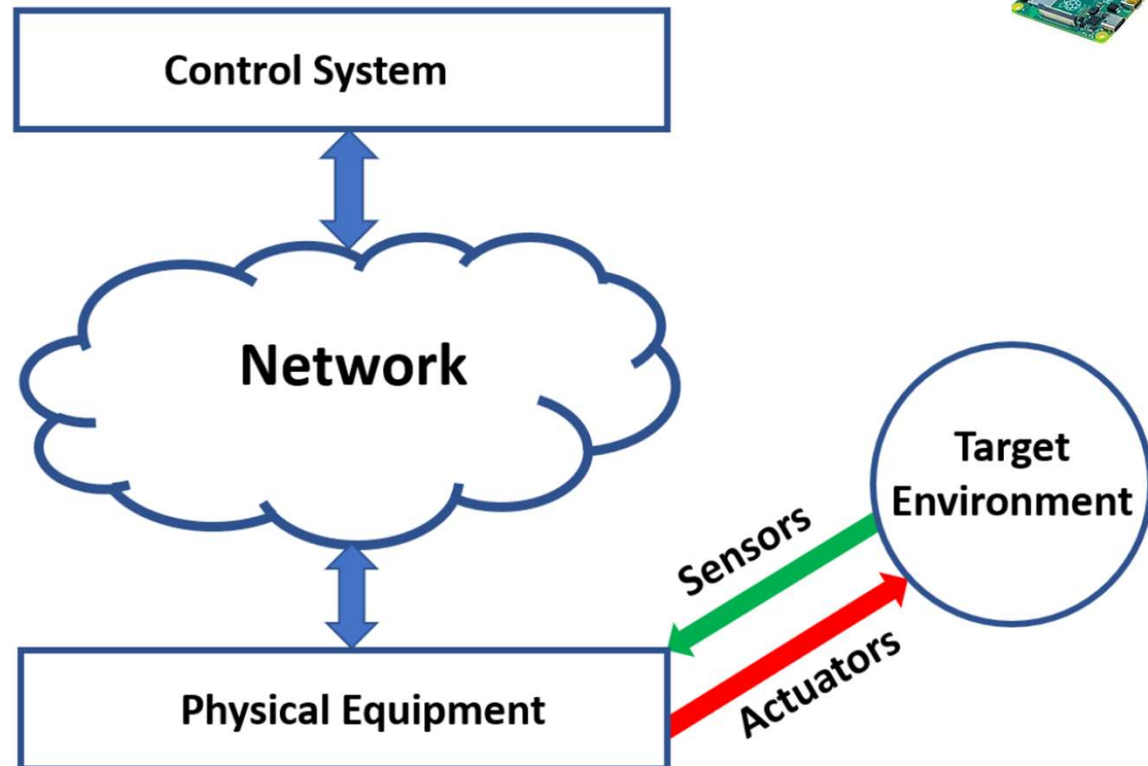
“ Over the last two decades, there has been massive progress in the fields of information technology, automation, robotics, Big Data, sensor technology (Internet of Things) and Artificial Intelligence (Russmann et al. 2015). These technological advances may be interpreted as prerequisites for the increasingly popular concept of Industry 4.0, which refers to the current trend of digitalization, automation, and data exchange in manufacturing (Kagermann et al., 2013; Schwab, 2016).”

“CPS is an engineering system designed to control and operate physical processes through computers” (Mughees, 2020).



The Model provides a hands-on framework for exploring CPS System Concepts using Physical Computing Techniques. Design Opportunities!!

CPS System Model



Node-RED: An Overview



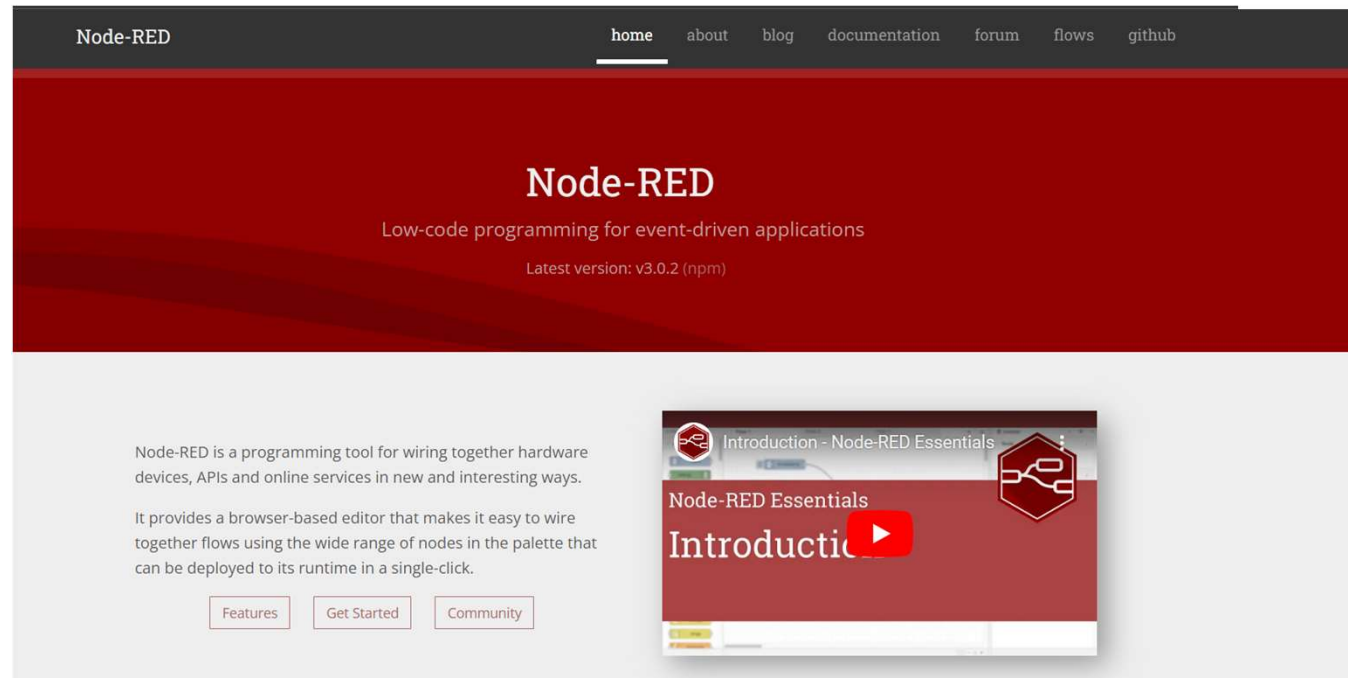
- A flow-based programming language
 - a) flow-based programming is a way of describing an applications behavior as a network of “boxes” or “nodes”.
 - b) flow-based programming defines the node’s purpose
- A model that lends itself very well to a visual representation
- The model makes it more accessible to a wider range of users.
- Consists of Node.js based runtime that allows access to flow editor through a web browser.

Node-RED: An Overview...

Node-RED Website



**Latest version:
v3.0.2 npm**



The screenshot shows the Node-RED website homepage. At the top, there is a navigation bar with links for 'home', 'about', 'blog', 'documentation', 'forum', 'flows', and 'github'. The main heading is 'Node-RED' in a large, bold font, followed by the tagline 'Low-code programming for event-driven applications' and 'Latest version: v3.0.2 (npm)'. Below this, there is a section with text describing Node-RED as a programming tool for wiring together hardware devices, APIs, and online services. It mentions a browser-based editor and the ability to deploy flows. There are three buttons: 'Features', 'Get Started', and 'Community'. To the right, there is a video player showing an 'Introduction - Node-RED Essentials' video.

Source: <https://nodered.org/>

Question 1

What is the latest version of Node-RED is available?

- a) v3.0.3**
- b) v3.0.0**
- c) v3.0.2**
- d) none of the above**



Node-RED: An Overview...

Node-RED Website












Node-RED's online resources

Source: <https://nodered.org/>

docs

Documentation

 Getting Started Everything from first install to deploying flows	 User Guide The definitive guide to using Node-RED	 Frequently Asked Questions And hopefully some answers
 Tutorials Examples of what you can do, taken one step at a time	 Cookbook Recipes to help you get things done with Node-RED	 Developing Flows Best practices for creating clear and reusable flows
 Creating Nodes How to create nodes to extend the Node-RED palette	 Developing the core Help to develop the core of Node-RED	 API Reference Admin, runtime and storage APIs

Node-RED: An Overview... ChatGPT Activity



Where can Node-RED be found on a Raspberry Pi?



Node-RED can be found pre-installed on the Raspbian operating system, which is the default operating system for Raspberry Pi. If you have Raspbian installed on your Raspberry Pi, you can find Node-RED in the Raspberry Pi menu under "Programming" or by typing "node-red" in the terminal.

If Node-RED is not pre-installed on your Raspberry Pi, you can install it using the following command in the terminal:

```
csharp
```

 Copy code

```
sudo apt-get install nodered
```

This will install Node-RED and its dependencies on your Raspberry Pi. After installation, you can start Node-RED by typing "node-red" in the terminal.

ChatGPT: Zero-Shot
Learning Exercise

Node-RED: An Overview... ChatGPT Activity



**ChatGPT: Zero-Shot
Learning Exercise:
Response Validated**

}	Programming	>	Arduino IDE
	Education	>	BlueJ Java IDE
	Office	>	DB Browser for SQLite
	Internet	>	Geany
	Sound & Video	>	Greenfoot Java IDE
	Graphics	>	Mathematica
	Games	>	mu
	Other	>	Node-RED

Node-RED: An Overview...

Node-RED Running on a Raspberry



```
pi@raspberrypi: Node-RED console
File Edit Tabs Help

Start Node-RED

Once Node-RED has started, point a browser at http:
On Pi Node-RED works better with the Firefox or Chrome browser

Use node-red-stop to stop Node-RED
Use node-red-start to start Node-RED again
Use node-red-log to view the recent log output
Use sudo systemctl enable nodered.service to autostart Node-RED at every boot
Use sudo systemctl disable nodered.service to disable autostart on boot

To find more nodes and example flows - go to http://flows.nodered.org

Starting as a systemd service.
█
```

Online Editor URL address is located here!

Node-RED: An Overview...

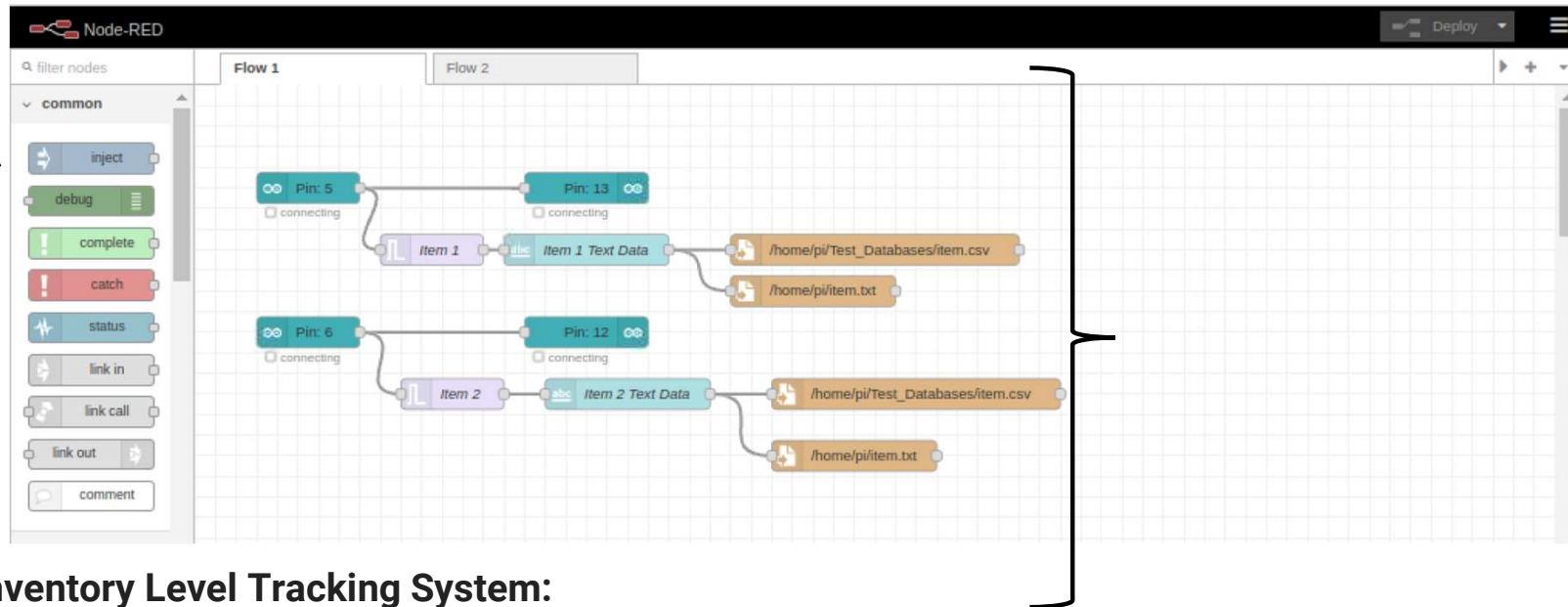
Node-RED Running on a Raspberry Pi



Widget
Palette



Node-RED
Online Editor



**Build A Configurable Inventory Level Tracking System:
Digi-Key Project Article: Maker.io staff = Dr. Don Wilcher**

<https://www.digikey.com/en/maker/projects/build-a-configurable-inventory-level-tracking-system/213191d4ae184912b215010cc24a039a>

Node-RED: Widgets

What is a Node-RED Dashboard?




- A set of dashboard nodes for Node-RED.
- A module that provides a set of nodes in Node-RED to quickly prototype a live data dashboard.
- Requires a runtime environment that operates JavaScript code.
- Node.js is the common runtime environment that enables the Node-RED Dashboard to operate.
- To install a stable version, use the **Menu-Manage palette option**.
 - a) Search for node - red dashboard
 - b) or run the following command in a linux terminal: `npm install node-red-dashboard`


Node-RED: Widgets

Installing Node-RED Palettes



ChatGPT: Zero-Shot
Learning Exercise:
Response Validated

 What linux command is used to install the Node-RED dashboard widgets?

 The Node-RED Dashboard widgets can be installed via the Node-RED palette manager. You can use the following `npm` command in the terminal to install the Node-RED Dashboard:

```
bash Copy code  
cd ~/.node-red  
npm install node-red-dashboard
```

This command installs the Node-RED Dashboard package in the `~/.node-red` directory under your home directory.

After running this command, you need to restart Node-RED for the changes to take effect. You can restart Node-RED by using the following command:

```
Copy code  
node-red-restart
```

Note that this command assumes that you have Node-RED already installed on your Raspberry Pi. If you don't have Node-RED installed, you can install it by following the instructions in my previous answer.

Source:

Question 2

What linux command to install the Node-RED palette was given by ChatGPT?

- a) npw install node-red dashboard**
- b) npu install node-red dashboard**
- c) install node-red dashboard**
- d) npm install node-red-dashboard**



Node-RED: Widgets

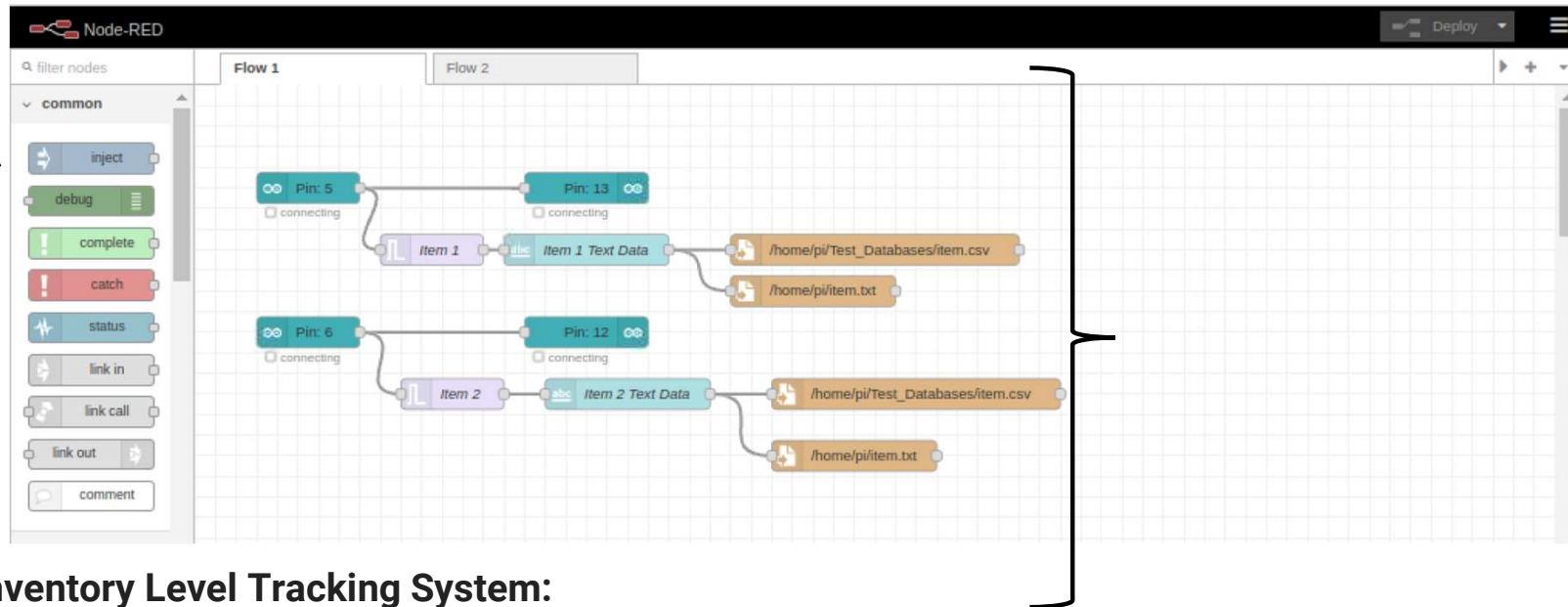
Node-RED Widget Palette Example Project



Widget
Palette



Node-RED
Online Editor



**Build A Configurable Inventory Level Tracking System:
Digi-Key Project Article: Maker.io staff = Dr. Don Wilcher**

<https://www.digikey.com/en/maker/projects/build-a-configurable-inventory-level-tracking-system/213191d4ae184912b215010cc24a039a>

Node-RED: Widgets Example Dashboard



Configurable Inventory Total Event System

Inventory

No inventory pulled

No inventory pulled

Configurable Inventory Total Event System

Inventory

hot dogs and hamburgers

peanut butter and jelly

Widgets allow the developer to
build Interactive HMIs and
Dashboards

Source: <https://www.digikey.com/en/maker/projects/build-a-configurable-inventory-level-tracking-system/213191d4ae184912b215010cc24a039a>

Node-RED: Widgets Example Dashboard



**Build A Configurable Inventory Level
Tracking System: Digi-Key Project Article:
YouTube Video**

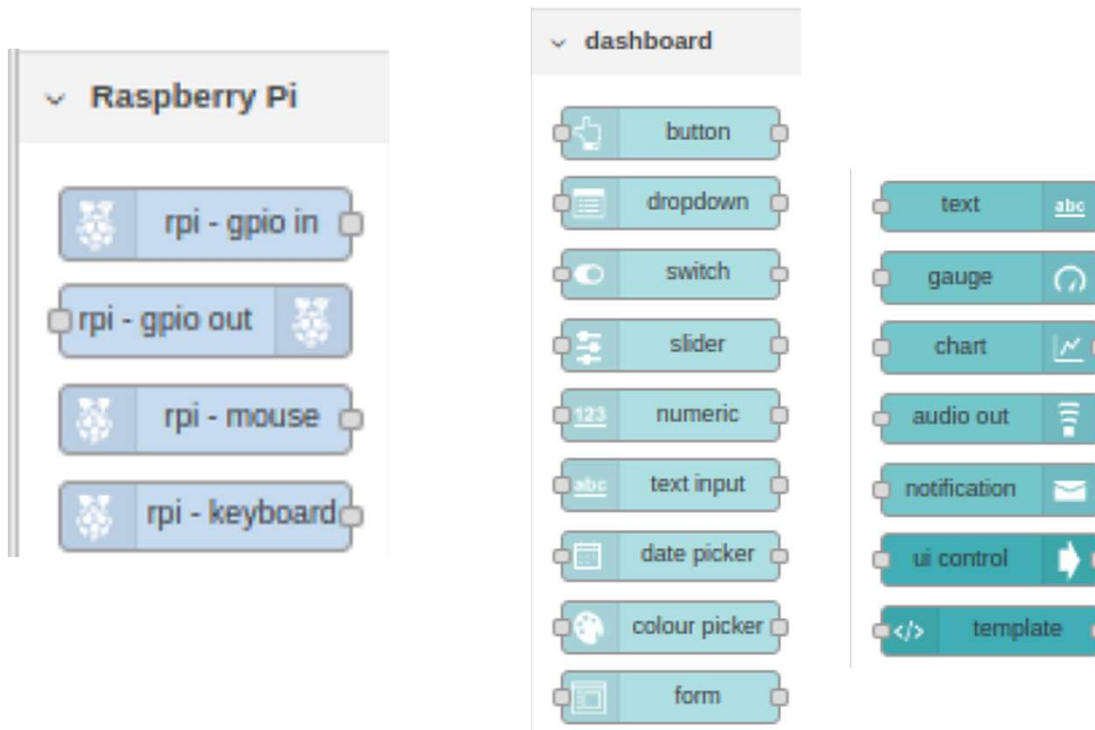


<https://youtu.be/4C6PC0tRjkM>

Source: <https://www.digikey.com/en/maker/projects/build-a-configurable-inventory-level-tracking-system/213191d4ae184912b215010cc24a039a>

Node-RED: Widgets

Raspberry Pi and Dashboard Widgets



The screenshot displays the Node-RED widget palette, organized into three main sections:

- Raspberry Pi:** Contains four widgets: `rpi - gpio in`, `rpi - gpio out`, `rpi - mouse`, and `rpi - keyboard`.
- dashboard:** Contains ten widgets: `button`, `dropdown`, `switch`, `slider`, `numeric`, `text input`, `date picker`, `colour picker`, and `form`.
- General Widgets:** Contains seven widgets: `text`, `gauge`, `chart`, `audio out`, `notification`, `ui control`, and `template`.

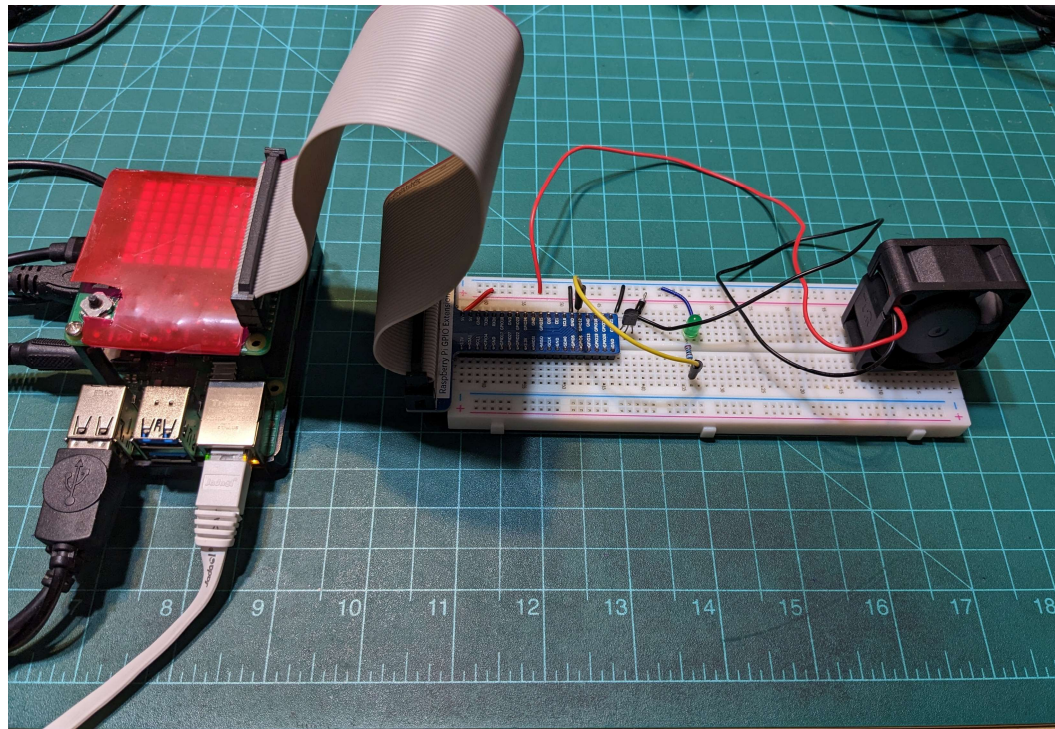
Question 3

In reviewing slide 22, which dashboard widget would be appropriate to show the binary status of a triggered event?

- a) button**
- b) numeric**
- c) text**
- d) none of the above**



Lab: Node-RED DC Fan Controller



Lab: Node-RED DC Motor Controller. . .



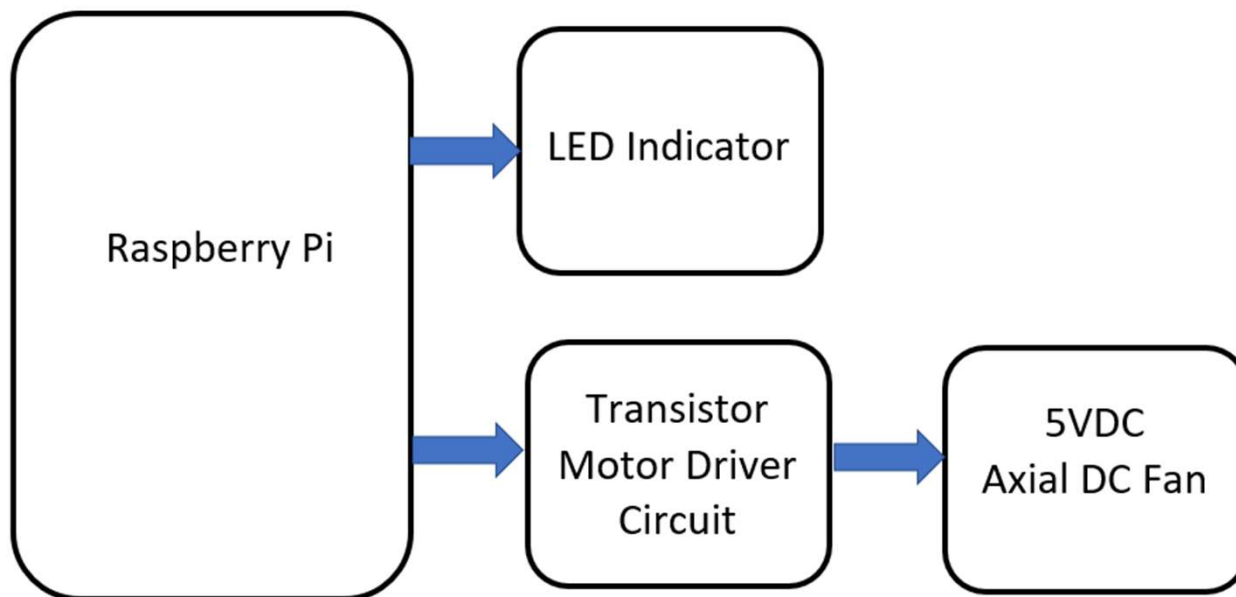
Learning Objectives:

- You will learn to wire a prototype transistor driver DC Motor Control Circuit on a solderless breadboard.
- You will learn to build a flow diagram using Node-RED Widget nodes.
- You will learn to build an Interactive Human Machine Interface (I-HMI) Dashboard using Node-RED Widgets.
- You will learn to operate the wireless DC Motor Controller using the I-HMI Dashboard.

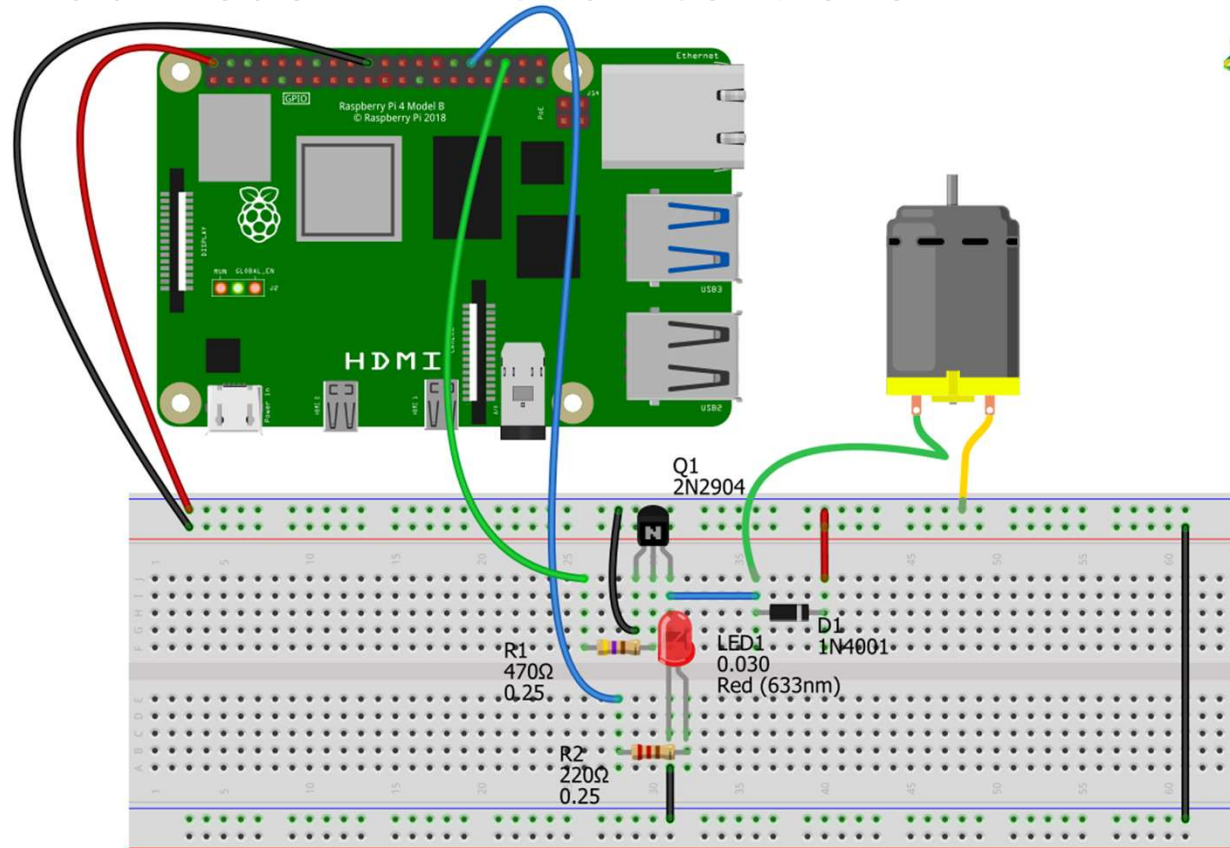
Lab: Node-RED DC Motor Controller. . .



A Conceptual Wireless DC Motor Controller System Block Diagram



Lab: Node-RED DC Fan Controller . . .

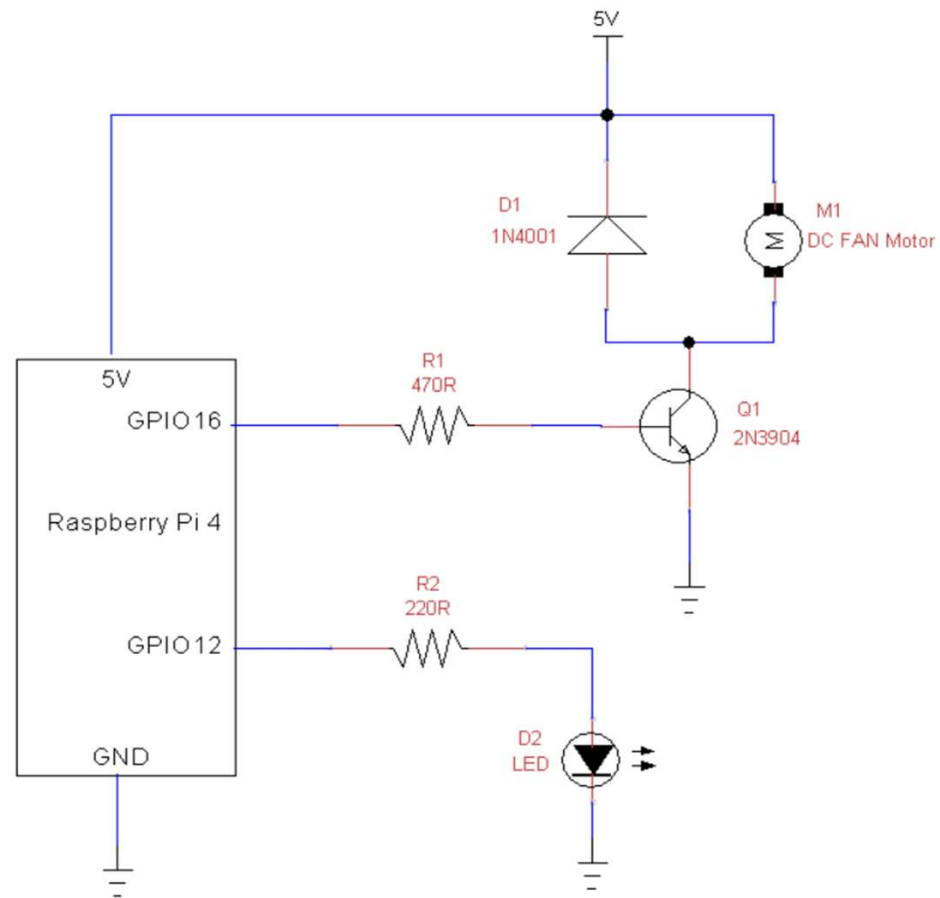


Conceptual Electrical Wiring Diagram:
Green Wire>GPIO16
Blue Wire>GPIO12

Lab: Node-RED DC Fan Controller. . .



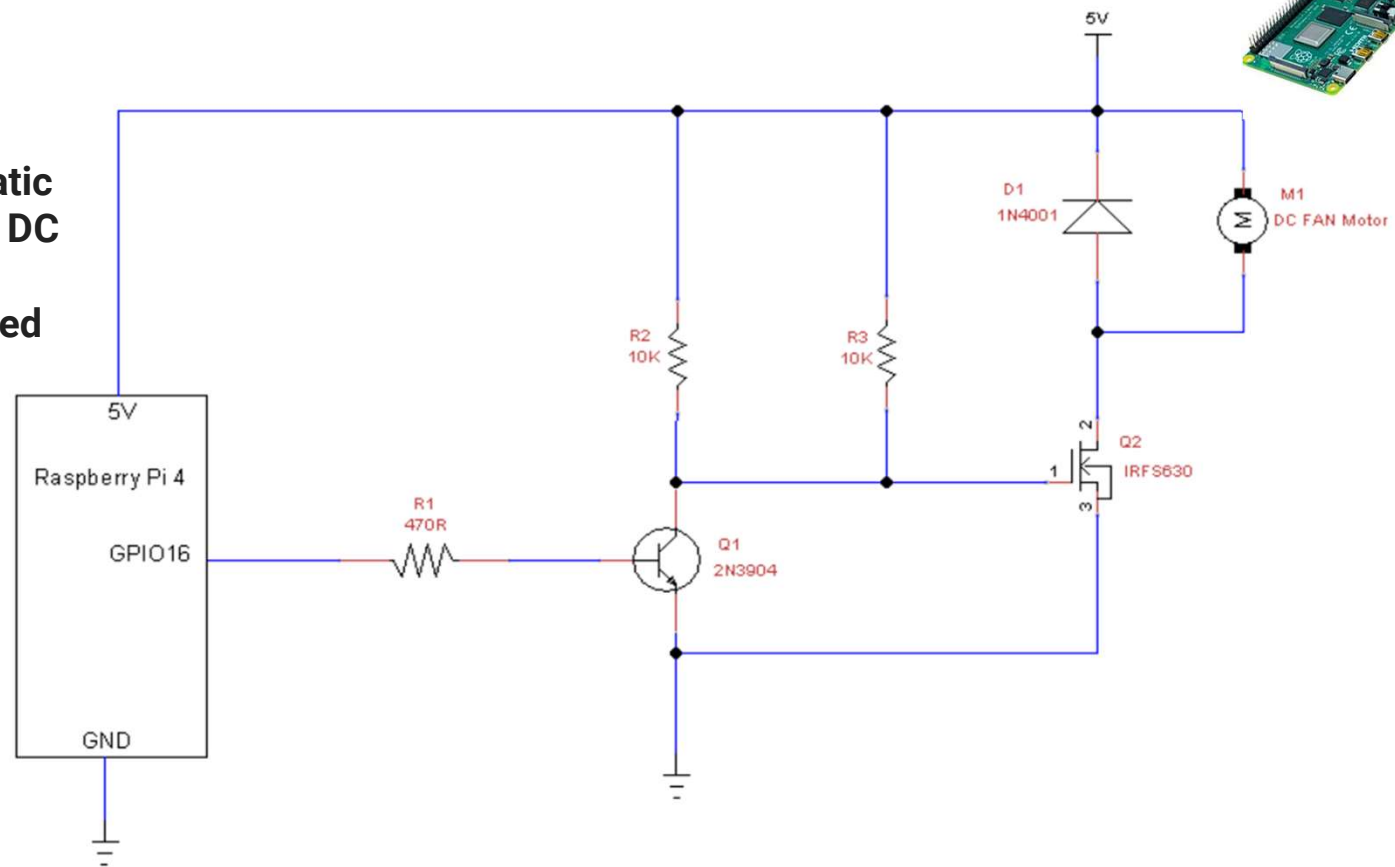
Electronic
Circuit
Schematic
Diagram



Lab: Node-RED DC Fan Controller. . .



**Electronic Circuit Schematic
Diagram: Power MOSFET DC
Motor Driver Circuit:
Circuit Provides an inverted
driver state.**



Question 4

In reviewing slide 29, with GPIO16 pin being activated(ON), what is the state of the DC motor?

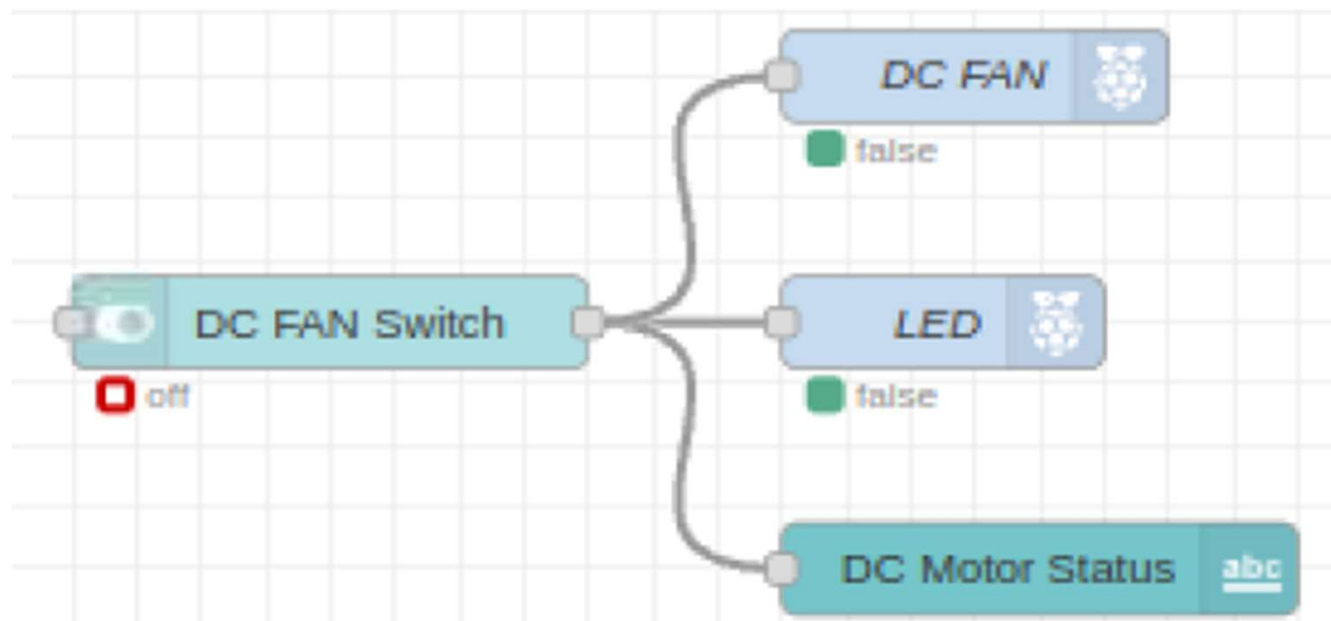
- a) ON**
- b) OFF**
- c) X – don't care**
- d) Hi Z (impedance)**



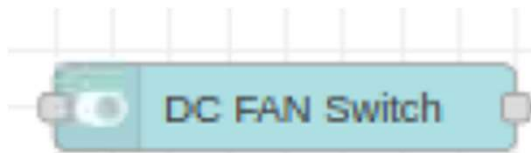
Lab: Node-RED DC Fan Controller . . . Building I-HMI Dashboard



I-HMI Flow Diagram



Lab: Node-RED DC Fan Controller. . . Configuring Widget Node Properties



Delete Cancel Done

Properties

Group [Wireless Controller] DC Fan Controlle

Size auto

Label DC FAN Switch

Tooltip optional tooltip

Icon Default

→ Pass through msg if payload matches valid state:

When clicked, send:

On Payload true

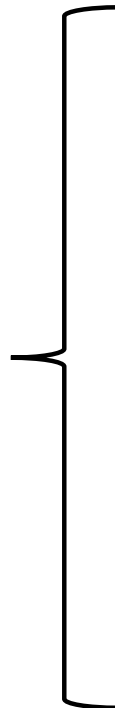
Off Payload false

Topic msg. topic

Class Optional CSS class name(s) for widget

Name

Lab: Node-RED DC Fan Controller. . . Configuring Widget Node Properties



Delete
Cancel
Done

Properties

Pin

3.3V Power - 1	2 - 5V Power
SDA1 - GPIO02 - 3	4 - 5V Power
SCL1 - GPIO03 - 5	6 - Ground
GPIO04 - 7	8 - GPIO14 - TxD
Ground - 9	10 - GPIO15 - RxD
GPIO17 - 11	12 - GPIO18
GPIO27 - 13	14 - Ground
GPIO22 - 15	16 - GPIO23
3.3V Power - 17	18 - GPIO24
MOSI - GPIO10 - 19	20 - Ground
MISO - GPIO09 - 21	22 - GPIO25
SCLK - GPIO11 - 23	24 - GPIO08 - CE0
Ground - 25	26 - GPIO7 - CE1
5D - 27	28 - SC
GPIO05 - 29	30 - Ground
GPIO06 - 31	32 - GPIO12
GPIO13 - 33	34 - Ground
GPIO19 - 35	36 - GPIO16
GPIO26 - 37	38 - GPIO20
Ground - 39	40 - GPIO21

BCM GPIO

Type Digital output

Initialise pin state?

Name

Lab: Node-RED DC Fan Controller. . . Configuring Widget Node Properties



Delete Cancel Done

Properties

Pin

3.3V Power - 1	2 - 5V Power
SDA1 - GPIO02 - 3	4 - 5V Power
SCL1 - GPIO03 - 5	6 - Ground
GPIO04 - 7	8 - GPIO14 - TxD
Ground - 9	10 - GPIO15 - RxD
GPIO17 - 11	12 - GPIO18
GPIO27 - 13	14 - Ground
GPIO22 - 15	16 - GPIO23
3.3V Power - 17	18 - GPIO24
MOSI - GPIO10 - 19	20 - Ground
MISO - GPIO09 - 21	22 - GPIO25
SCLK - GPIO11 - 23	24 - GPIO8 - CE0
Ground - 25	26 - GPIO7 - CE1
SD - 27	28 - 5C
GPIO05 - 29	30 - Ground
GPIO06 - 31	32 - GPIO12
GPIO13 - 33	34 - Ground
GPIO19 - 35	36 - GPIO16
GPIO26 - 37	38 - GPIO20
Ground - 39	40 - GPIO21

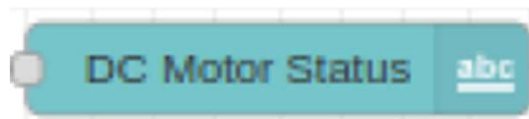
BCM GPIO: 12

Type: Digital output

Initialise pin state?

Name: LED

Lab: Node-RED DC Fan Controller. . . Configuring Widget Node Properties



Delete Cancel Done

Properties

Group [Wireless Controller] DC Fan Controlle

Size auto

Label DC Motor Status

Value format {{msg.payload}}

Layout

label value label value label value

label value label value

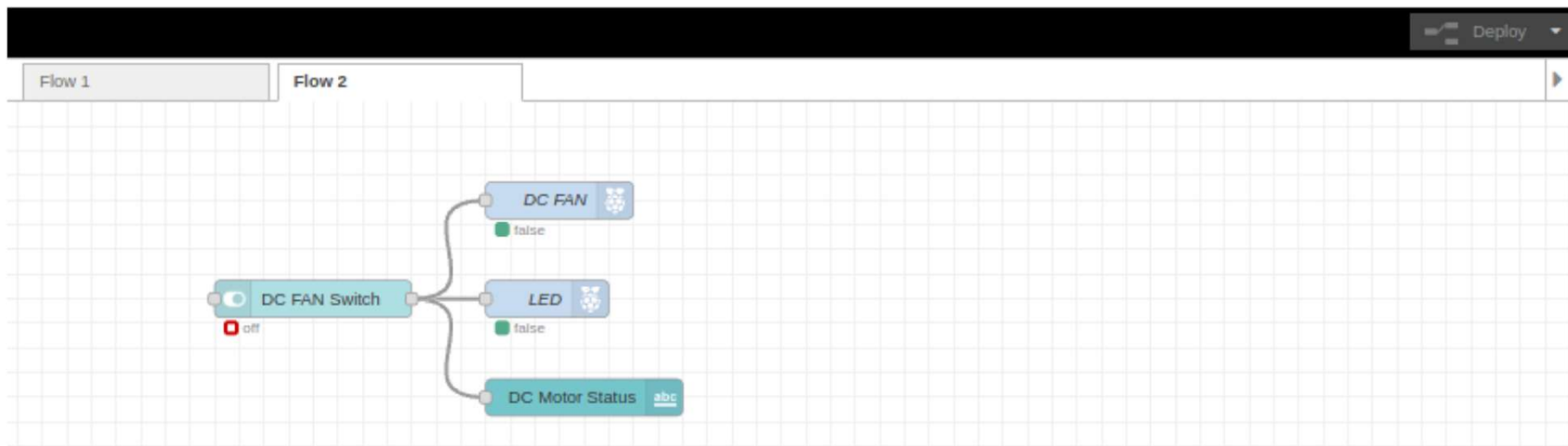
Class Optional CSS class name(s) for widget

Name

Lab: Node-RED DC Fan Controller . . . Publish Dashboard = Deployment



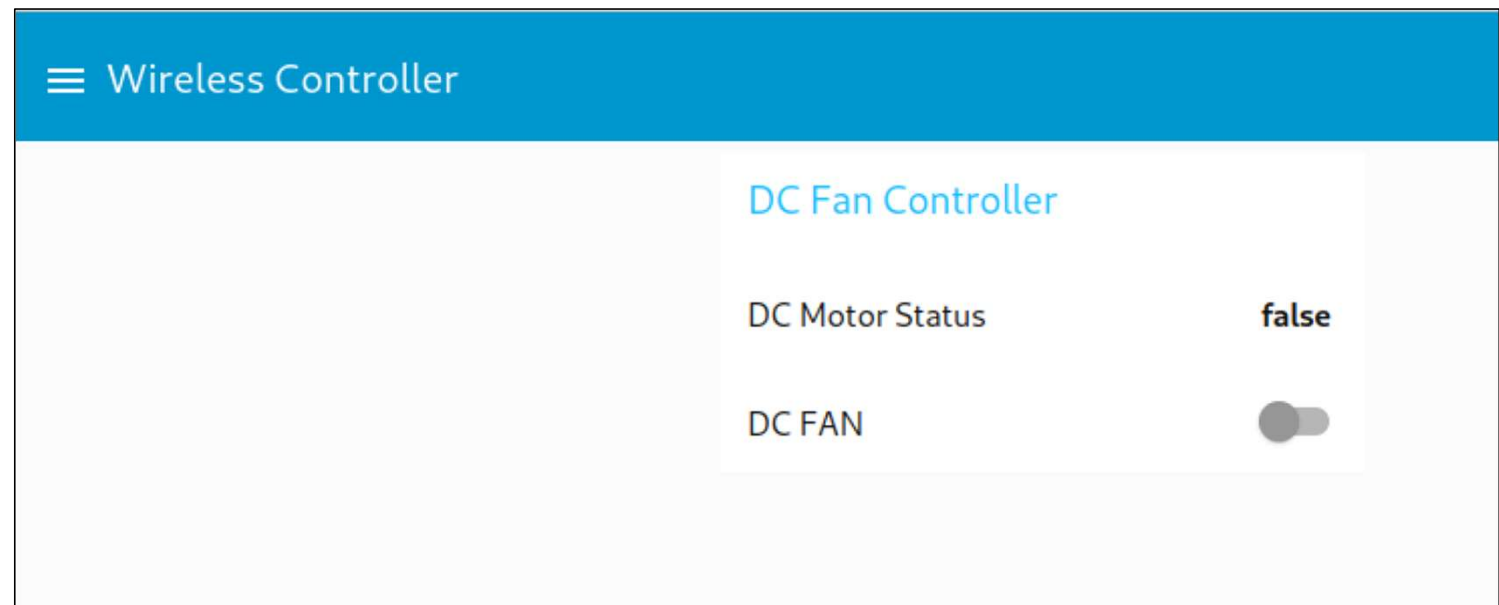
Click Here to Publish the Dashboard



Lab: Node-RED DC Fan Controller. . . Publish Dashboard = Deployment



I-HMI Dashboard



The screenshot shows a web interface titled "Wireless Controller". It features a section for "DC Fan Controller" with two controls: "DC Motor Status" set to "false" and "DC FAN" with a toggle switch.

Control	Value
DC Motor Status	false
DC FAN	<input type="checkbox"/>

Lab: Node-RED DC Fan Controller. . . Working Node-RED DC Fan Controller



Node-RED DC Fan Controller:
YouTube Video



<https://youtu.be/6BHe1xDPO3Y>

Question 5

In reviewing slide 36, would the Deploy button need to be pressed if the name LED was changed to Red LED on the node?

- a) YES**
- b) NO**



Thank you for attending

Please consider the resources below:

555 timer application: <https://www.engineersgarage.com/servo-motor-control-using-555-timer-ic/>

ElectronicsTutorial. (2021). Light sensors. [https://www.electronicstutorials.ws/io/io_4.html#:~:text=The%20light%20sensor%20is%20a,%20into%20electricity%20\(electrons\)](https://www.electronicstutorials.ws/io/io_4.html#:~:text=The%20light%20sensor%20is%20a,%20into%20electricity%20(electrons))

ESP8266 Hardware Design Guidelines: <https://www.espressif.com/en/support/documents/technical-documents>

Kagermann, H., Wahlster, W., & Helbig, J. (2013). *Recommendations for implementing the strategic initiative Industrie 4.0: Securing the future of German manufacturing industry*. <https://www.din.de/blob/76902/e8cac883f42bf28536e7e8165993f1fd/recommendations-for-implementing-industry-4-0-data.pdf>

Mughees, A. (Sept 05, 2020). *Discrete and process automation: From cyber-physical systems to pervasive intelligence*. <https://electronics360.globalspec.com/article/15647/from-cyber-physical-systems-to-pervasive-intelligence>

Osoyoo Website. (2022). WiFi iot learning kit. <https://osoyoo.com/2020/05/30/wifi-iot-learning-kit-for-Arduino/>

Russamann, M., Lorenz, M., Gerbert, P., Waldner, M., Justus, J., Engle, P., & Harrisch, M. (2015). *Industry 4.0 – The future of productivity and growth in manufacturing industries*. The Boston Consulting Group.

Schwab, K.(2016). *The fourth industrial revolution*. Penguin Random House.

Thank you for attending

Please consider the resources below:

Yang, S. (2011). *Internet-based control systems: Designs and applications*. Springer.

Skraba, A., Stanovov, V., & Semenkin, E. (2019). Modelling of dc motor and educational application in cyber-physical systems. *Materials Science and Engineering*, 537. <https://doi:10.1088/1757-899X/537/4/042008>

Skraba, A., Stanovov, V., & Semenkin, E. (2020). Development of control systems for study of PID controller in the framework of cyber-physical systems. *Materials Science and Engineering*, 537. <https://doi:10.1088/1757-899X/537/4/042008>

Wilcher, D. (2012). *Learn electronics with arduino*. Apress.



DesignNews

Thank You

Sponsored by

