TECHNOLOGY CAMP

DAY 4 : INTERNET-OF-THINGS

Building IoT devices with GPIOs

Session 2

YELLOW CIRCLE INC
PO Box 2383
Elk Grove, CA 95759-2383

Teacher Lesson Plan

Building IoT devices with GPIOs

Session Name:

Building IoT devices with GPIOs

Summary:

This lesson explores IoT device control by using circuits built on a breadboard and connected to the Raspberry Pi by GPIO pins. GPIO stands for general-purpose input/output. On a Raspberry Pi, GPIO pins can be used to make connections with external devices. By connecting components to the correct GPIO pins and programming software to either send or receive information from the pins, the Raspberry Pi can be used to communicate with and/or control external devices.

Time Allotment:

65 minutes

Learning Objectives:

- *Learn what GPIOs are*
- *Learn how GPIO pins on the Raspberry Pi can be used to create IoT devices*
- *Learn how to use Node-RED on Raspberry Pi to program IoT device control*

Supplies:

- *Scrap paper / notepad to take notes*
- *Laptop / computer with Internet access*
- *Raspberry Pi and connected breadboard with pre-built circuits*

Learning Activities:

- **(2 minutes) - Session overview**

  This lesson explores IoT device control by using circuits built on a breadboard and connected to the Raspberry Pi by GPIO pins. Students learn what are GPIOs, how

GPIO pins on the Raspberry Pi can be used to create IoT devices, and how to use Node-RED on Raspberry Pi to program IoT device control

- **(5 minutes) – What are GPIO pins?**

  GPIO stands for general-purpose input/output. On a Raspberry Pi, the GPIO pins are rows of small metal posts (called pins) that can be used to make connections with external devices. Different pins have different functions and capabilities. By connecting components to the correct GPIO pins and programming software to either send or receive information from the pins, the Raspberry Pi can be used to communicate with and/or control external devices.In 2018, how many devices were connected to the internet?

- **(8 minutes) – What is a breadboard?**

  A breadboard is a solderless device for temporary prototype with electronics and test circuit designs. Most electronic components in electronic circuits can be interconnected by inserting their leads or terminals into the holes and then making connections through wires where appropriate. The breadboard has strips of metal underneath the board and connect the holes on the top of the board. -wiring.org.co

  http://wiring.org.co/learning/tutorials/breadboard/

  Demonstrate the breadboard and explain:

  - *The channel down the middle splits the breadboard into two separate work areas. Rows on the left of the channel are not connected to rows on the right of the channel.*
  - *In the middle of the board are 2 sets of columns marked A-E and F-J and numbered in rows. Each row is separate from every other row, but all of the holes in a row are "wired" together.*
  - *Wires in columns A through E will be connected together if they are all in the same row.*
  - *Wires placed in different rows of the same column (for example rows 3 and 4 in column A) will NOT be connected.*

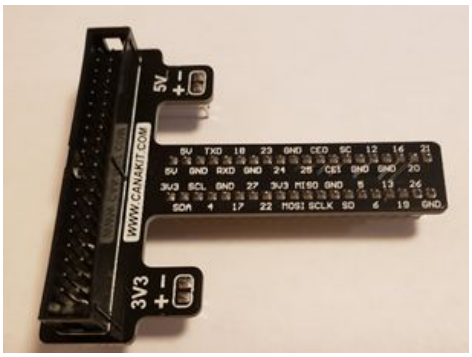
- *The two columns on the left and right of the board marked + and – are for carrying voltage. They are oriented differently from the middle section.*
- *For these sections, the connections run vertically (the red + and line down the positive side and the blue – and blue line down the negative side are visual reminders)*
- *We will not be using these sections today.*

- **(15 minutes) – Demonstration: Breadboards and Circuits**

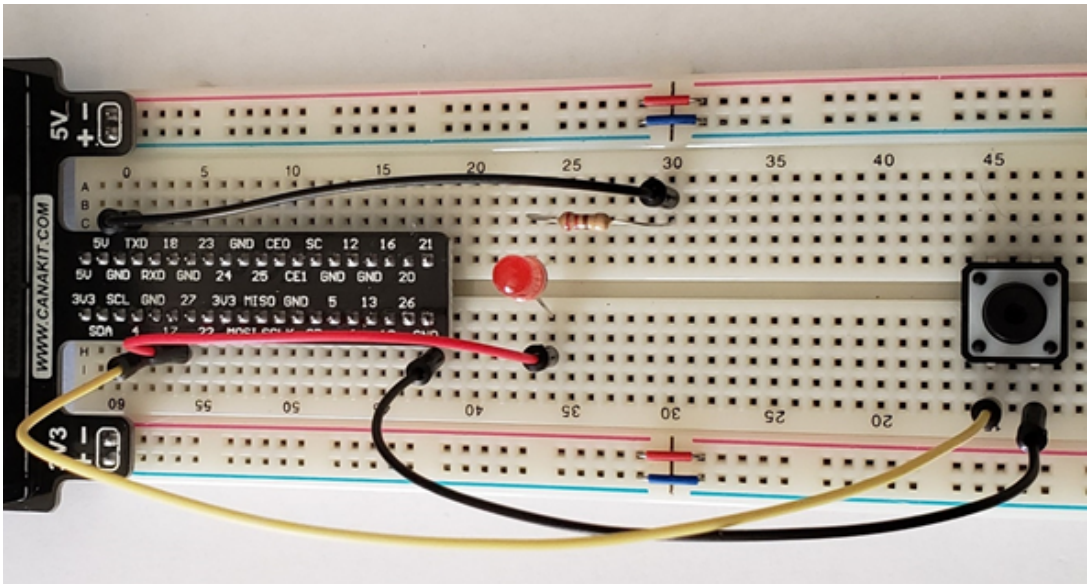
  Explain to students that their breadboards have been pre-wired for our lessons, but we will review the configuration so that they can set up their own equipment in the future.

  Remind students that we won't be making changes to the circuits today, but that if they use breadboards in the future, they must be careful that the Raspberry Pi is turned off before being connected to the breadboard or making changes to the circuits, because stray current or mis-wired circuits could produce stray current that could damage components or the Raspberry Pi itself.

  Next, point out the GPIO pins on the Raspberry Pi and the GPIO Interface Board on the breadboard. Explain that the interface board makes it easier to connect circuits to the GPIO pins on the Raspberry Pi by moving the connections to the breadboard.



Point out that their pre-wired breadboard contains 2 separate circuits that will be used in future lessons, and LED and a button.



LED Circuit

- *Explain that the LED has 2 connections, the Anode (+) and the Cathode (-).*
- *The Anode is the longer wire and the Cathode is the shorter*
- *The LED is inserted across the channel with the Anode on one side and the Cathode on the other to help protect against shorts. (Remind them that the channel separates the 2 rows into separate connections.*
- *Explain that the colored wire is connected to the Anode and to GPIO pin 17. This is the pin that will provide the instructions to the LED.*
- *Explain that to protect the LED from too much voltage (which would cause it to burn out), a resistor is connected to the cathode and then to the ground pin via the black wire.*

Button Circuit

- *Explain that although the button has 4 pins, we will only be using the 2 on the left side.*
- *Again point out that the button is positioned across the channel to assure that the left and right sides are fully isolated. (We won't be using the right side pins today)*
- *Explain that a resistor is not needed for the button. So the two left side pins are wired, respectively to the GPIO 4 pin by the colored wire and a ground pin by a black wire.*

Connecting the Breadboard to the Raspberry Pi

- *Explain that a ribbon cable is used to connect the interface board to the GPIO pins on the Raspberry Pi.*
- *Show students the notch on the Interface Board ribbon connection socket aligns with the corresponding alignment "bump" on the ribbon cable. Remind them that you must be careful when connecting the ribbon cable no to unseat the interface board (recommend supporting the interface board from below while plugging in the cable from above).*

- *The red stripe on the ribbon cable indicates the pin 1 side will be on the left when viewing from the notch side.*

- *Plug the other end of the ribbon cable into the GPIO pin assembly of the Raspberry Pi, so that the red strip is on the same side as the SD card (the side opposite the USB Ports).*

- **(3 minutes) - Lab: Raspberry Pi Set up**

  (Confirm that all students have both VNC and Node-RED running and open on their laptops before proceeding. Follow these instructions only if any students shut down their Raspberry Pi at the end of the last lesson)

  - *Please **<u>do not</u>** unplug or turn off the PI without shutting it down from the App Menu ( in the upper left corner). Doing so could corrupt the SD card.*
  - *Open your laptop and double click on the VNC Viewer icon.* 
  - *Login using the default credentials:*
  - *UserID: pi*

- *Password: raspberry*

- *When the Raspberry Pi desktop appears, click on the VNC icon*  *in the upper right and note your IP address listed under Connectivity*

- *Open a new browser tab and enter* ***http://{Your IP Address Here}:1880*** *to start Node-RED*

- **(15 minutes) – Activity : Programming Device Control with Node-RED**

1. *Scroll down to the Raspberry Pi section and drag the rpi-gpio node with the raspberry on the right onto the workspace. This is an output node. We are this node to use the LED to display the output of the on and off buttons we will be creating.*

2. *DoubleClick on the node and complete the edit rpi-gpio out node as below: Note the box for initialize pin state is checked and the initial value from the dropdown is set to low (0). This starts the LED in an off state at deployment. (Setting it to 1 will cause the LED to flash on briefly at deployment.)*

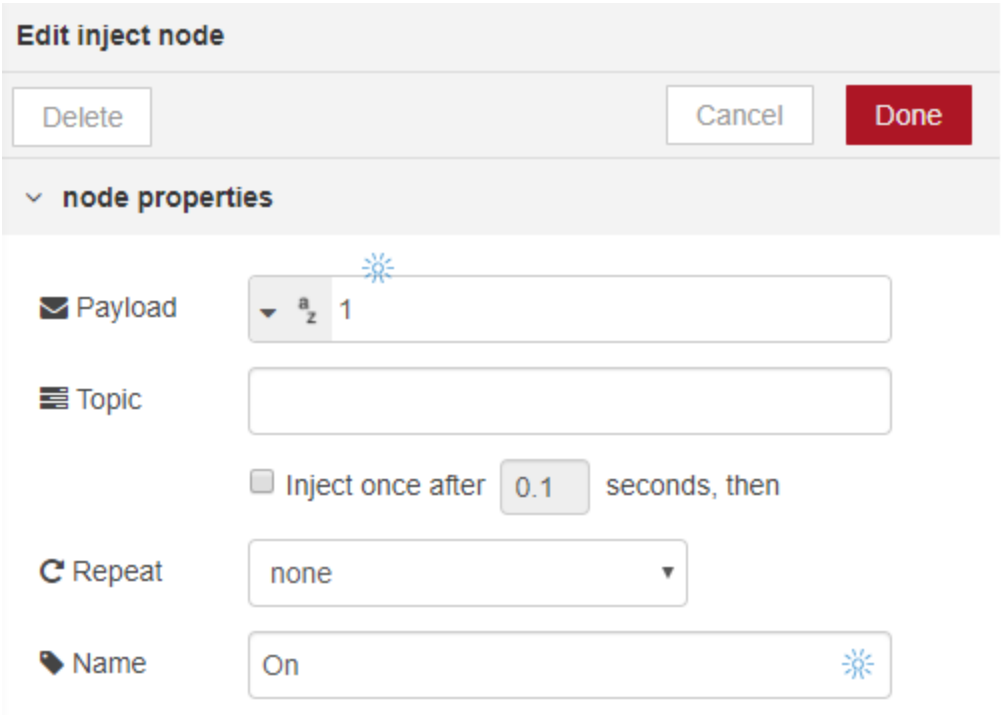
Note: GPIO17 is designated as pin 11. Do not be confused. The GPIO numbers do not correspond to the pin numbers because many of the pins are named (e.g. Ground, 3.3V Power, etc.) rather numbered. Node-RED also sees that pin 7 (GPIO 4) is in use, but we will not be coding for that pin in this lesson, so this information is of no concern.

3. *Scroll back to the top and drag an inject node onto the workspace. An inject node is used on inject (send or insert) a payload into a flow. In this case it will be sending a value to the rpi-gpio node to tell it to turn the LED on or off*

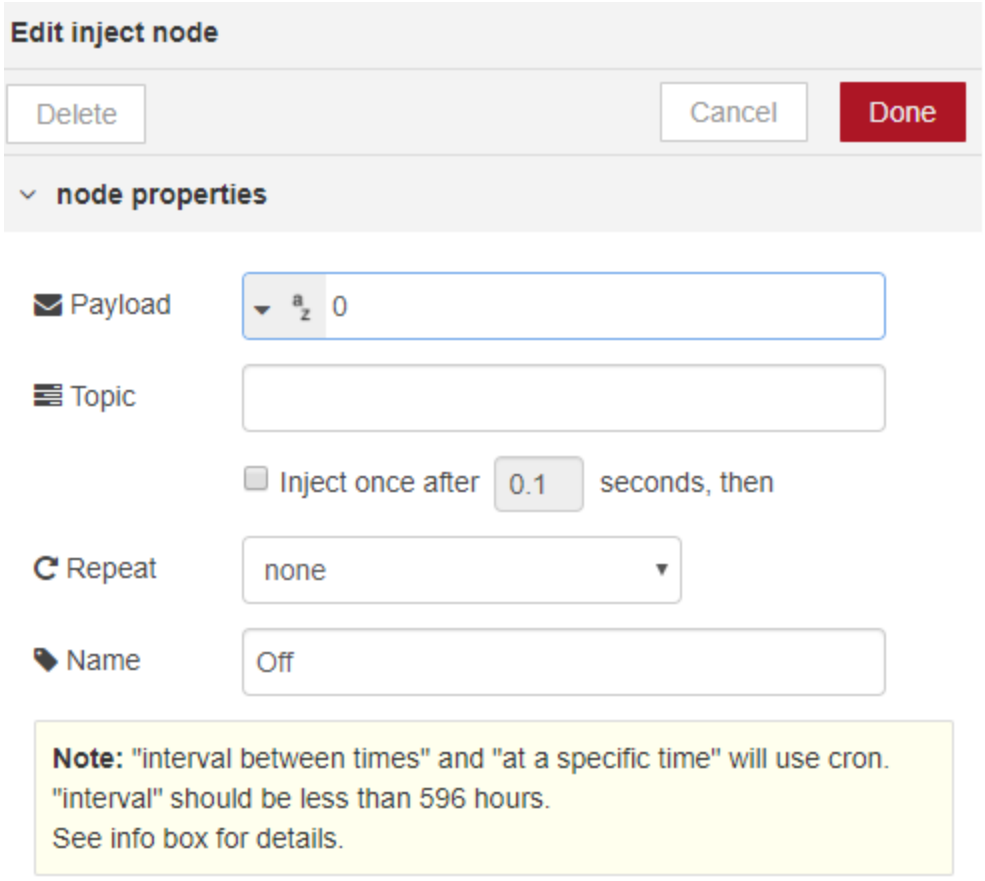
4. *DoubleClick on the inject node and complete the fields as below:*

    a. Payload: Select string and set to 1 (1=ON 0=Off)

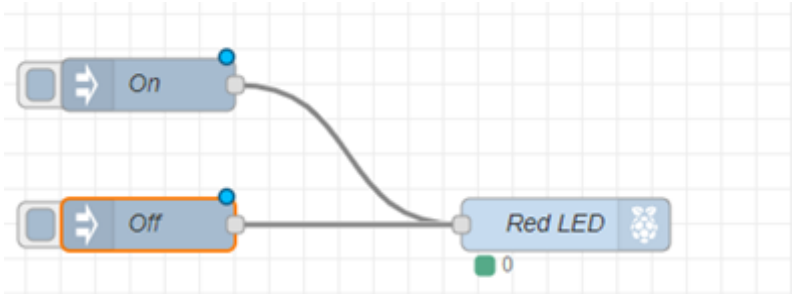
    b. Name: On



5. *We are now going to repeat steps 3-4 to create the Off button by replacing the payload value of 1 with a 0. (1=ON 0=Off)*



6. *Connect both buttons to the output LED node and deploy.*

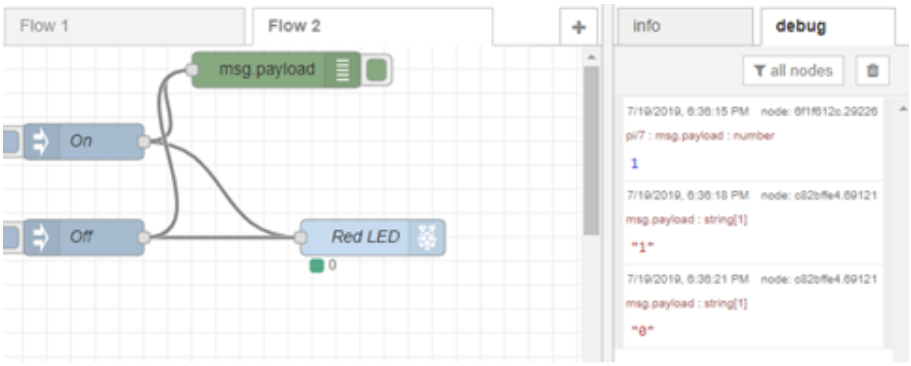


7. *Click the on button, and the LED should light.*

8. *Click the off button and the LED should turn off.*

Note: Students that the use of the on and off software buttons is similar to the way in which software in a phone app can be used to turn a smart switch on and off.

- **(15 minutes) Try It!**
  - *Modify the flows you built.*
  - *Can you create a flow that also outputs the messages to debug so you can see the button messages being sent to the output node?*



  - *Build your own flows.*
  - *Experiment with using different options and note the results.*
- **(2 minutes) Closing / Wrap-up**
  - *What GPIOs are*
  - *How GPIO pins on the Raspberry Pi can be used to create IoT devices*
  - *How to use Node-RED on Raspberry Pi to program IoT device control*
- **(1 minutes) - What's next?**

  Inform students to head back to the cafeteria for lunch break, and remind them to use the restroom before next session starts.