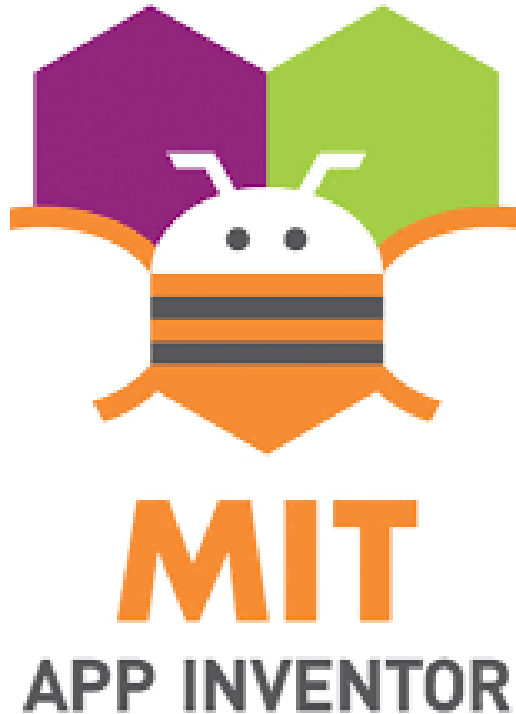Special Topics in Design I
(Prototyping in IOT)
DSL 810

Topic 5
Networking and Communications
Instructor: Jay Dhariwal,
Asst. Prof., IIT Delhi

24th February 2020

MIT App Inventor

- Cloud based tool to develop apps for android
- Developed by Google and maintained by MIT Media Lab, CSAIL MIT
- Develop UI in Designer Editor and program it in Blocks Editor
- Login to https://appinventor.mit.edu using gmail account
- Lets get started! Beginner Tutorials

Bee buzzing App

- Graphical Programming
- Flowchart based programming
- Event based programming
- Designer Editor, Blocks Editor
- Changing the sound mp3
- Testing using the Emulator OR
- MIT AI2 Companion App with same WiFi network
- Build - download and install apk file on Android Phone – your first App!
- File Names – only numbers, letters, underscores.
- Change sound, image, other changes.
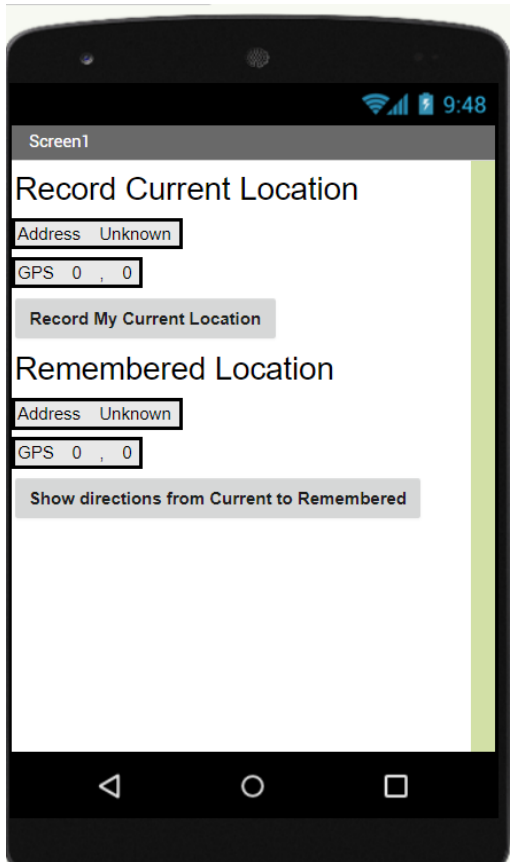
Text to Speech (Talk to Me) App

- Text to Speech [Video1](#)
- Accelerometer sensor [Video2](#)
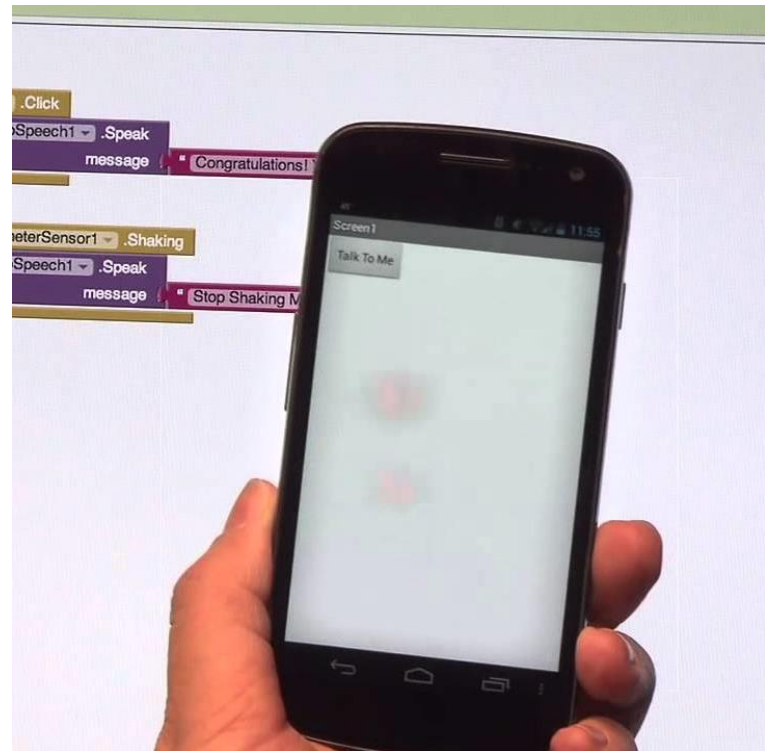
More Beginner Apps

- [Digital Doodle App Video](#)

- [Ball Bounce App Video](#)

- [Multiple Screens Video](#)

- [More tutorials](#)
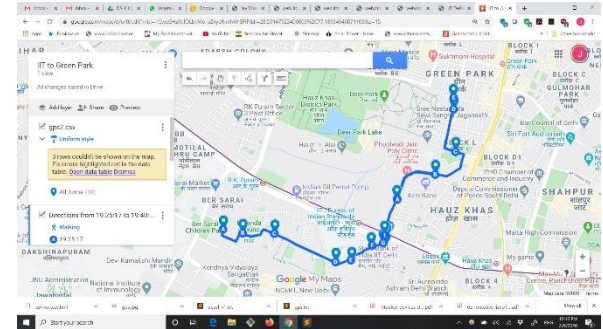
- Convert txt to csv and then csv to kml
- Use GPS Visualizer, My maps from google
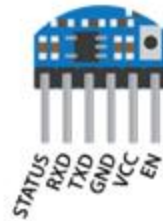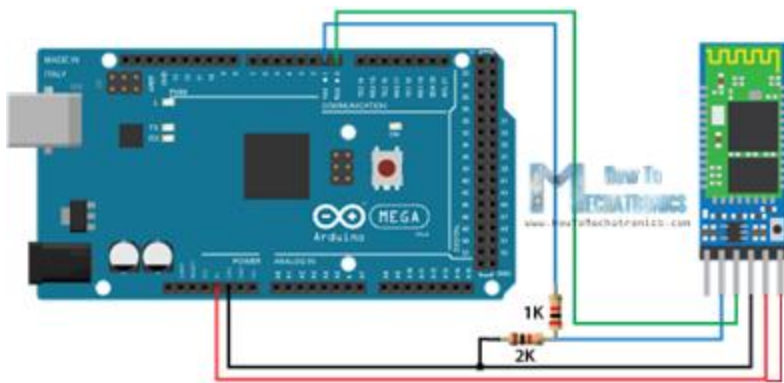- Video

Mobile App with GPS sensor

# HC-05 Bluetooth Module name change steps

- Make sure HC-05 module not paired with any device

- Make circuit connections (key pin = enable pin).

- Upload code in Arduino IDE

- Remove Vcc pin before unplugging power from USB cable

- Plug power from USB cable to Arduino. Connect Vcc. LED blinks slowly every 2 seconds

- HC-05 is in AT command mode

- Go to Serial Monitor (9600 baud rate, NL+CR) and give AT commands:

  AT
  AT+NAME?
  AT+NAME=XXX_HC05
  AT+NAME?

- now remove the power to arduino. remove key pin and then plug power back to arduino.

- Now the light starts blinking rapidly suggesting the bluetooth module is ready to receive and transmit.

- Test it with Arduino code and Mobile App
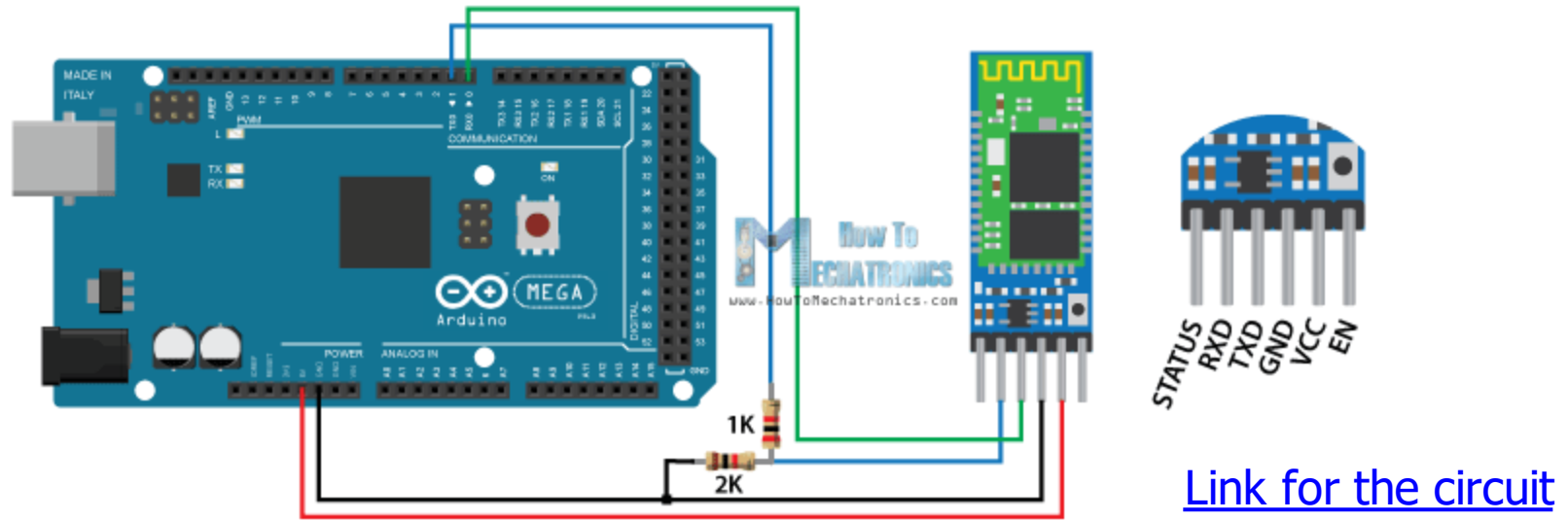
Bluetooth communication

- Wireless with HC-05 module with a range up to 10 m.

- Resistors for potential divider circuit to achieve 3.3 V for HC-05 Rx pin

- default baud rate of 9600 bits per second for Rx and Tx. Default baud rate in AT command mode is 38400.

- [please remember to disconnect Rx, Tx pins for Serial Communication](#) on UART pins (0,1) for Uno

- Hardware circuit connections, code in Arduino IDE, code & install Mobile App, pair Bluetooth module in the phone, password (1234), Connect in the App, [Procedure](#)

- [Serial, Software serial, Serial & Software Serial with DHT sensor codes in Arduino IDE](#) (send and receive data to and from Mobile App and Arduino IDE)

- [App making codes in App Inventor](#) [Video](#)

- [Slider in Mobile App for PWM based control of hardware](#)   [RGB Led PWM control](#) MOSFET
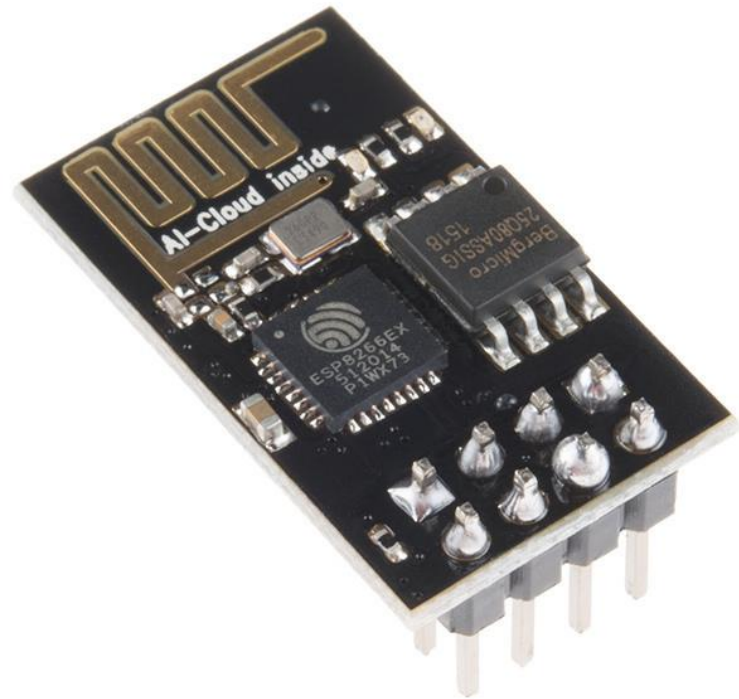
[Music Player using Mobile App, Arduino and a Buzzer](#) using HC-05 Bluetooth Module
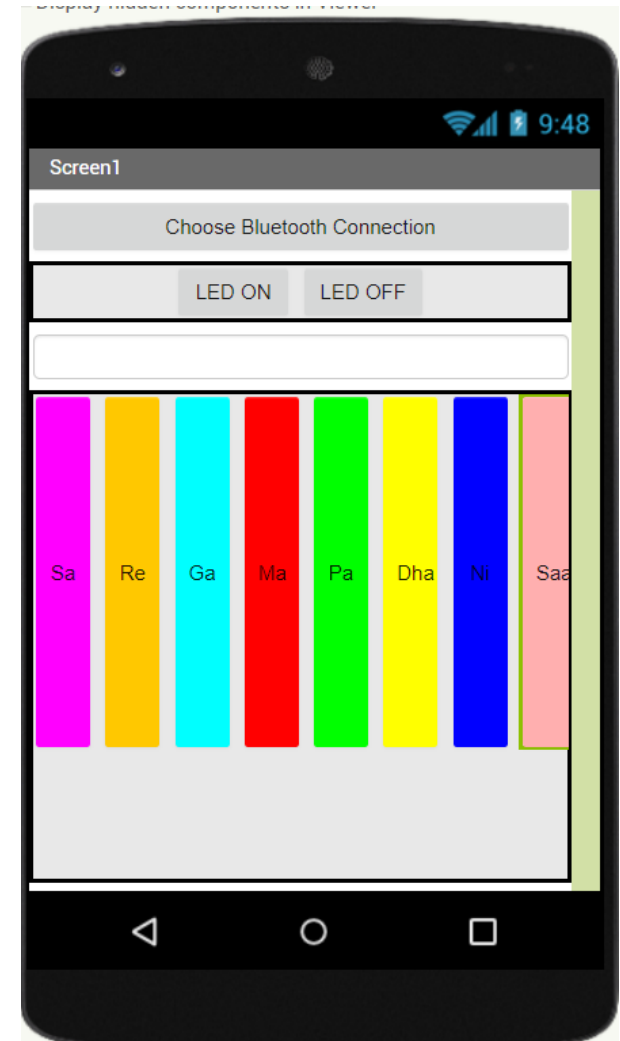
Wi-Fi communication

- [NodeMCU vs. Arduino + Wi-Fi Module](#) (Memory, 3.3 V, Libraries, Board size)
- Installation in Arduino IDE
- Blinking LEDs
- Simple Web Server
- Sending sensor data to cloud
- Easy coding in TUNIOT

NodeMCU installation

- Instructions
- How to place NodeMCU on the breadboard?
- Go to File > Preferences. In the "Additional Boards Manager URLs" field, type (or copy-paste) *http://arduino.esp8266.com/stable/package_esp8266com_index.json*. Network = no proxy (use Personal Hotspot). Don't forget to click OK!
- Then go to Tools > Board > Board Manager. Type "esp8266" in the search field. The entry "esp8266 by ESP8266 Community" should appear. Click that entry and look for the install button on the lower right.
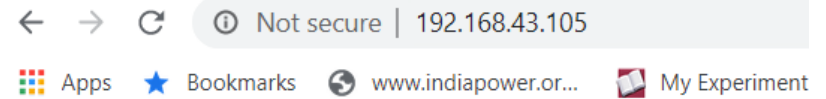
NodeMCU pinout

- 3.3 V so 5 V power supply for sensors such as ultrasonic HC-SR04 sensor
- GPIO pins are pins in Arduino IDE

- Blink LED on GPIOpin 16 (D0) /2 (D4)

- Select NodeMCU 1.0 and Port.
- [Blink an external LED on any pin](#)

Blue LED Connected to GPIO2

LED Connected to GPIO16
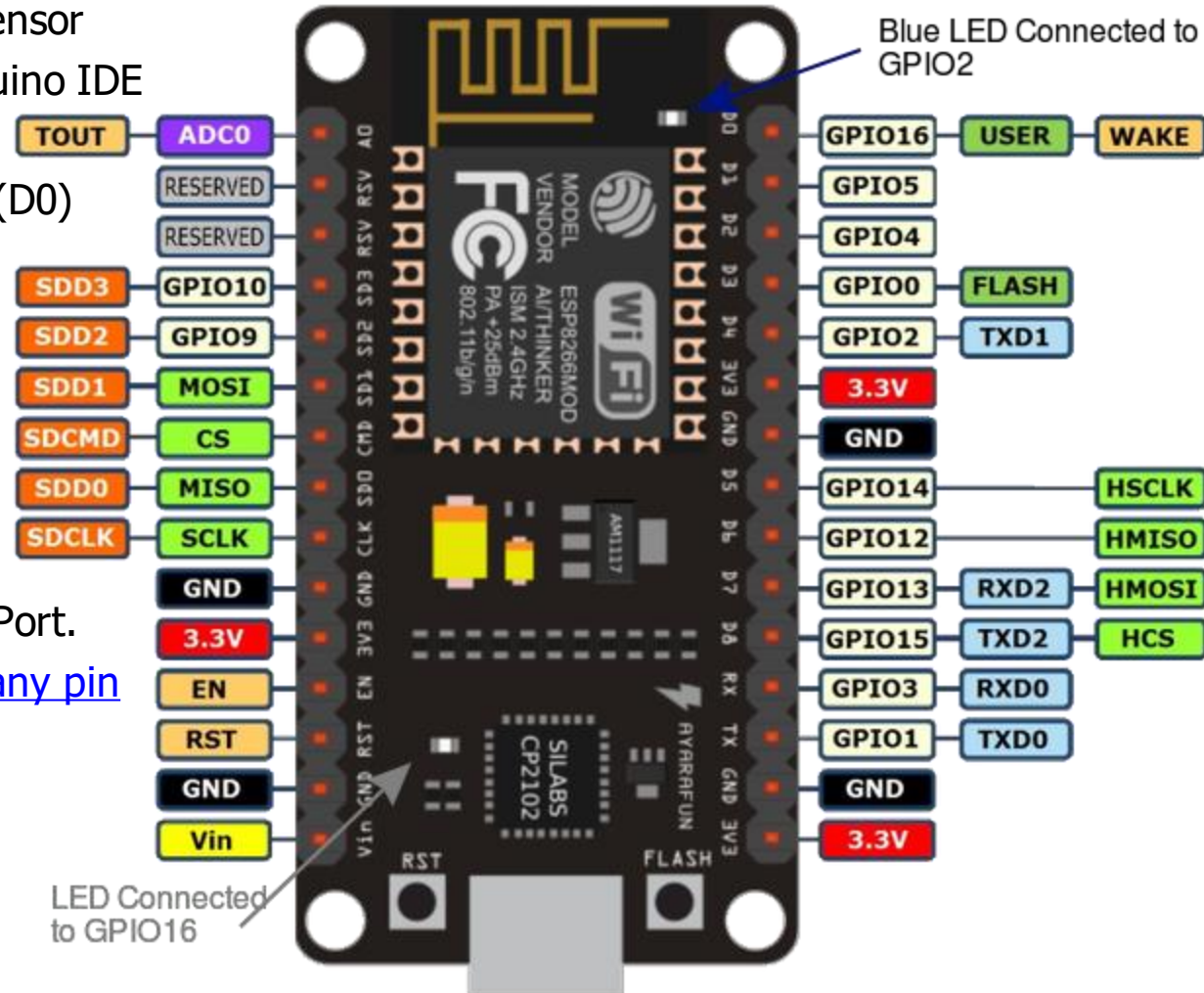
COM4

.
Connected to AndroidAP
IP address: 192.168.43.105
Web server started!

← → C ① Not secure | 192.168.43.105

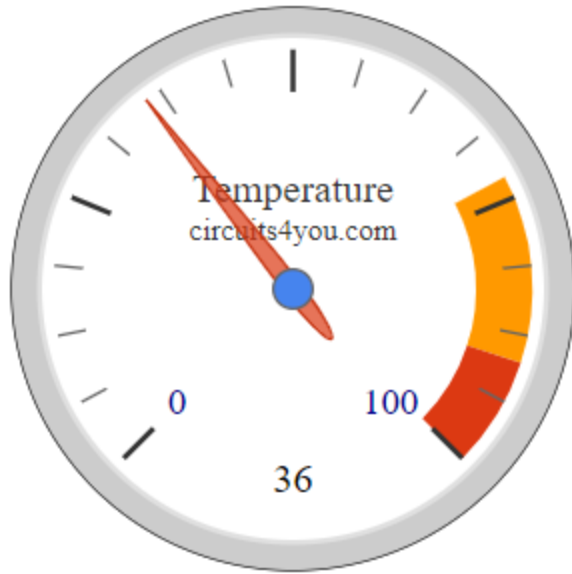::: Apps ★ Bookmarks ⊗ www.indiapower.or... 📖 My Experiment

Simple NodeMCU Web Server

[ON] [OFF]

Creating a NodeMCU Web server

- Using NodeMCU as an IOT device
- IP address on the serial monitor
- View source and try it on
  w3schools javascript
- Type in anything within the buttons Arduino

Connect a potentiometer to the A0 pin
Of NodeMCU

Creating a
NodeMCU Web
server with
advanced javascript

Connect NodeMCU to IOT cloud

- Create an account on [ThingSpeak](#) (Mathworks)
- Setting up a channel on ThingSpeak
- Install ThingSpeak library for Arduino IDE
- Working with DHT11/DHT22 sensor
- Firstly, install [DHT library](#) in Arduino IDE and run the DHTtester example
- Download [Arduino IDE Code](#) to connect NodeMCU to ThingSpeak
- Mixing sensor specific code with cloud upload code
- TunIOT for understanding code segments (Gulshan)

Realtime data update channel link
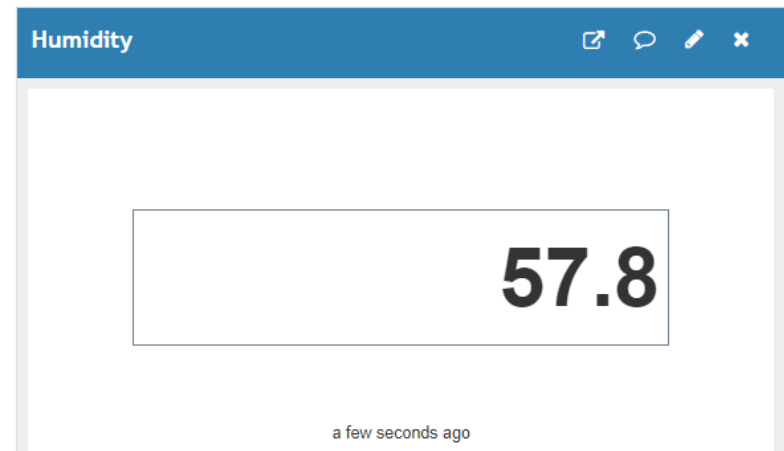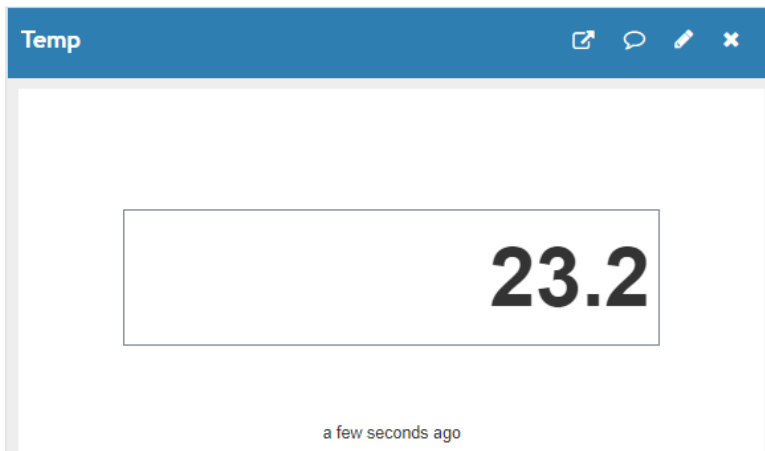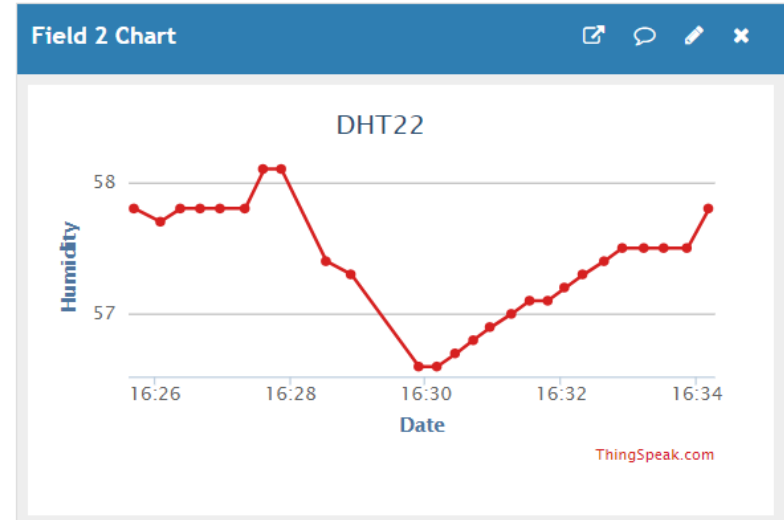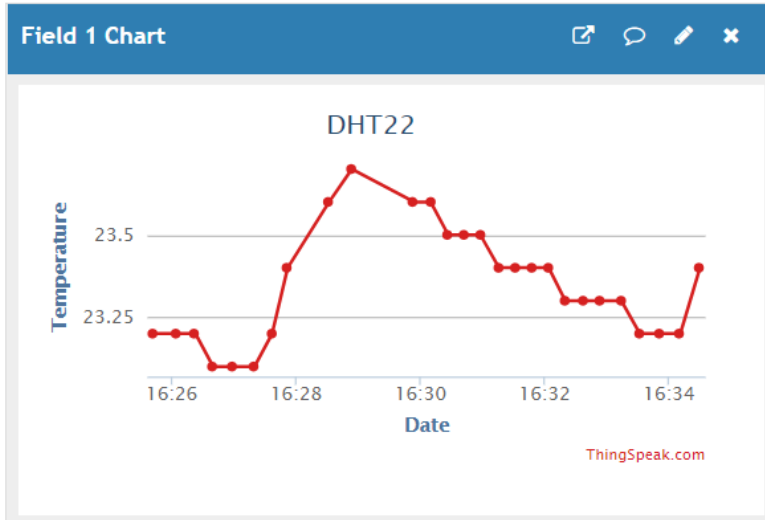
Data analytics using MATLAB later

Realtime data update channel link

Data analytics using MATLAB later

Working with HC-SR04 sensor

- Make the HC-SR04 sensor arduino code work on NodeMCU.
- Provide 5 V from Arduino. Connect Arduino ground to NodeMCU ground.
- Connections to GPIO pins 16 (Trigger) and 5 (Echo)
- Setup another channel on ThingSpeak.
- [Arduino Code](#)
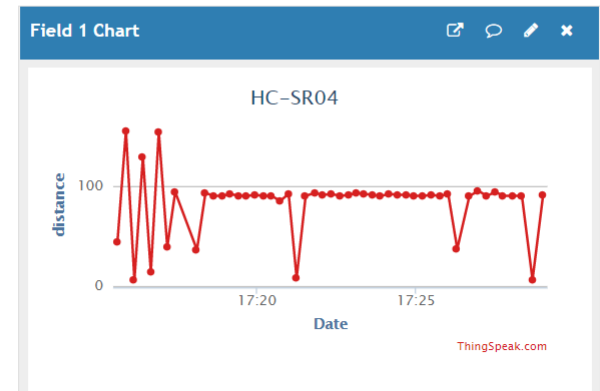- TunIOT for understanding code segments

NodeMCU+IOT cloud for any sensor
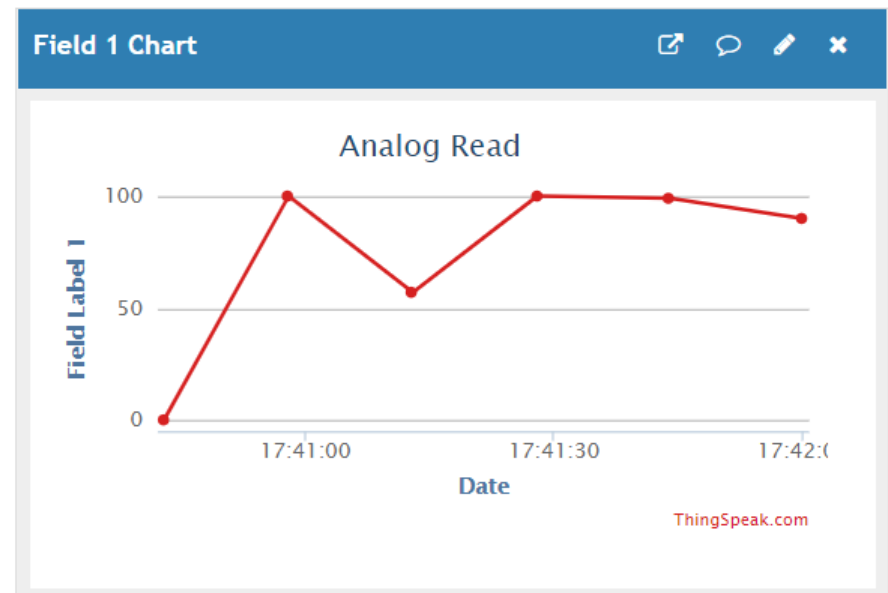
- Example for a potentiometer with analogRead
- [Arduino Code](#)

Channel Stats

Created: 3 minutes ago
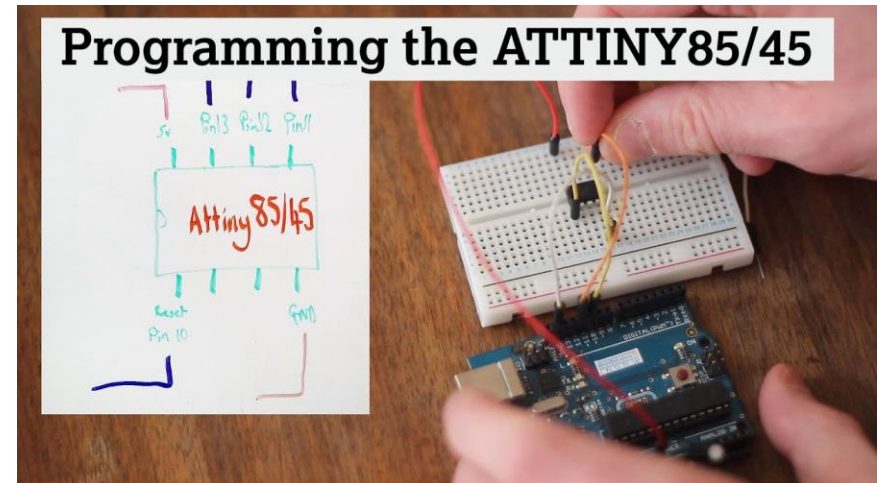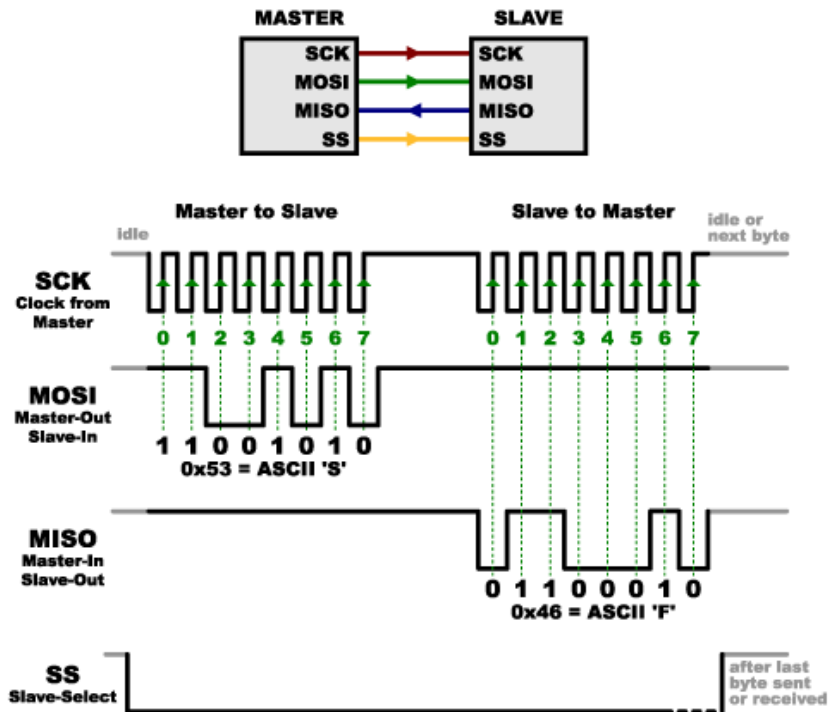Last entry: less than a minute ago
Entries: 6

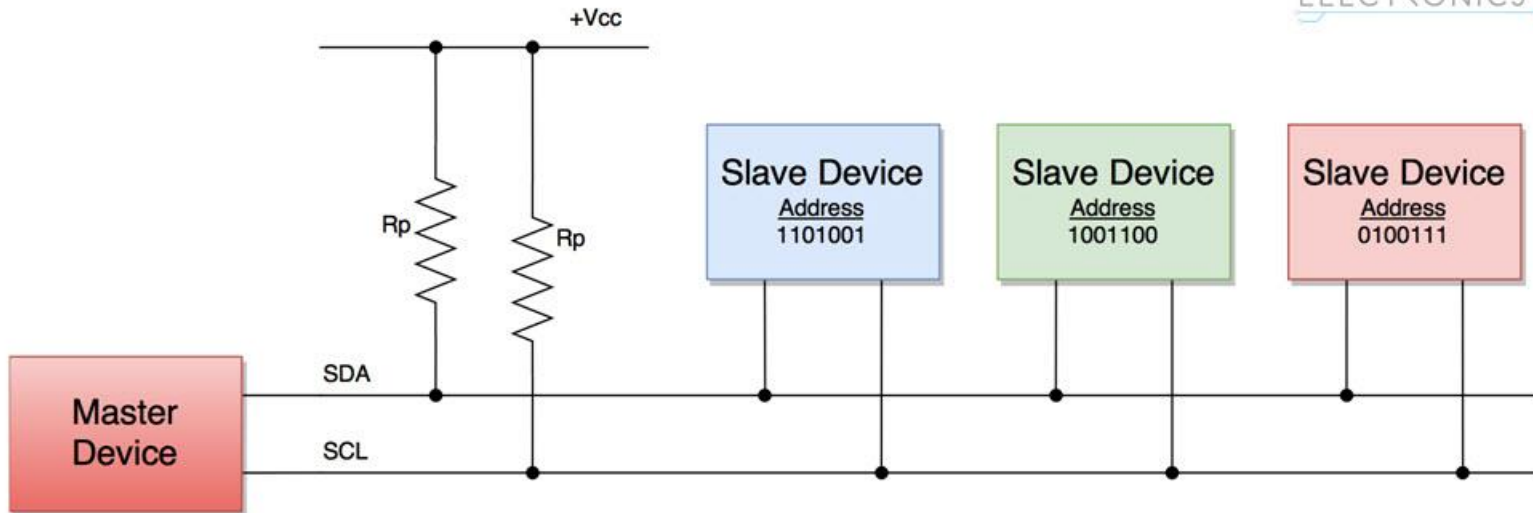# SPI communication

Programming the ATTINY85/45

I²C communication

Announcements

- [Minor Test 2 guidelines](#)
- [Minor Test 2 evaluation criteria](#)