Intelligent Dipstick (IDS)

Harold Sleeper

Table of Contents

Parts Needed

- o 1 ESP32 Dev Kit C v4

- o 1 Screw Terminal Block Breakout Board for ES32 Dev Kit C

- o 2 green LED lights

- o 2 red LED lights

- o 4 220-ohm resistors

- o 1 MAX 6675 + K-Type Thermocouple

Refer to Capstone Gateway Diagram.pdf and Capstone Sensor Diagram.pdf as an assembly

guide.

Setting Up Arduino IDE 1.8.19 for IDS Sensors and Gateway

Arduino's IDE needs to be configured to compile its sketches appropriately for the ESP32.

** Disable any antivirus software temporarily for the next steps! **

1. First in the Arduino IDE, go to the *File* menu and choose *Preferences*

   This should open a *Preferences* window. Within the *Preferences* window near the bottom,

   should be a text field labeled *Additional Boards Manager URLs.*

   Copy the next two lines into the text field and press *OK.*

https://dl.espressif.com/dl/package_esp32_index.json,

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json
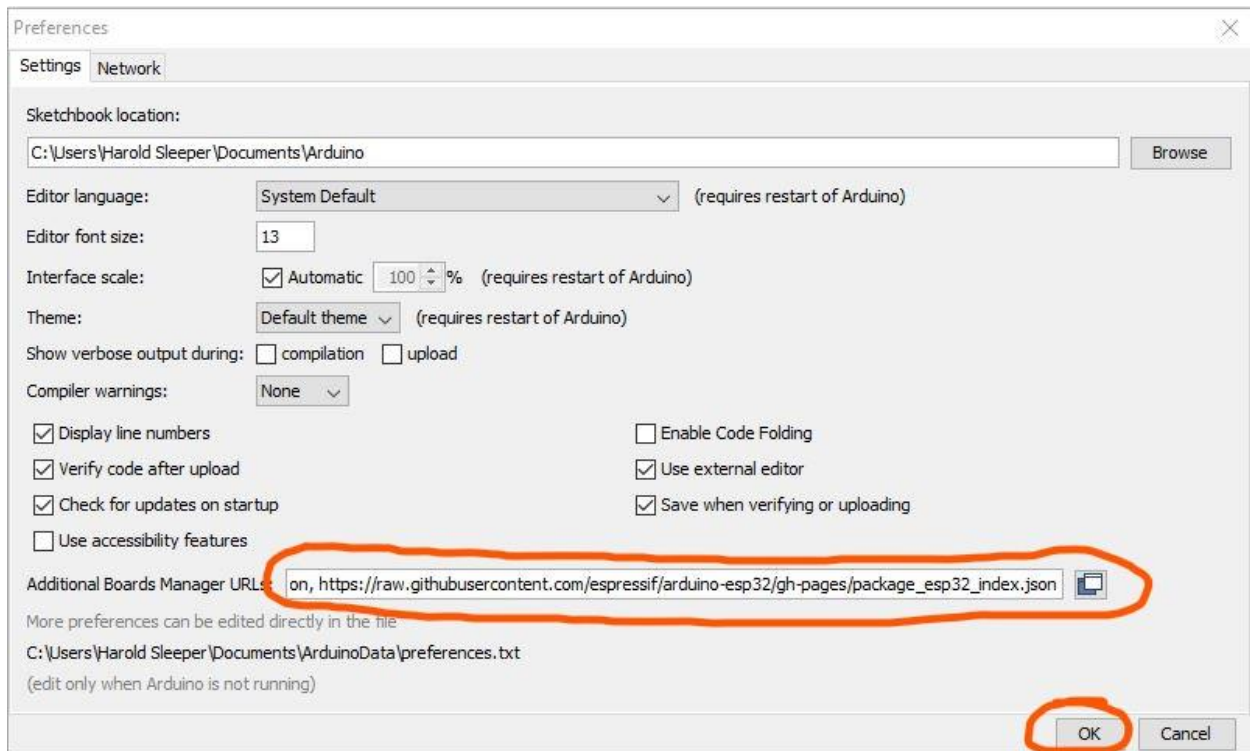
The two separate URLs should be separated by a comma as shown and as it appears in this text.

2. Next in the Arduino IDE, go to the *Tools* menu, choose *Board*, then go to *Boards Manager…*

    This should open the *Boards Manager* window.

    Search for ESP32 and press install button for the ESP32 by Espressif Systems.

3. Go to [CP210x USB to UART Bridge VCP Drivers - Silicon Labs (silabs.com)](#)

    Near the top of the webpage should be a hyperlink labeled *Downloads.* Click it!

    Choose *CP210x Universal Windows Driver* from the options.

Software Downloads

Software (11)

Software · 11

CP210x Universal Windows Driver	v10.1.10 1/13/2021
CP210x VCP Mac OSX Driver	v6.0.1 3/31/2021
CP210x VCP Windows	v6.7 9/3/2020
CP210x Windows Drivers	v6.7.6 9/3/2020
CP210x Windows Drivers with Serial Enumerator	v6.7.6 9/3/2020

Extract the downloaded zip file into a location of your choosing and within that folder open

CP210xVCPInstaller_x64.exe (this assumes you are using a 64-bit computer, otherwise

choose the …_x86 option)

Follow the on-screen installation instructions

Prepare Blues Wireless Notecard

- Please follow the instructions found here on the Blues Wireless Developer site for directions on installing the Blues Wireless Notecard onto the Blues Wireless Notecarrier-AF.

- Once the Notecarrier is assembled use the instructions found here to create a Notehub Project and obtain a ProductUID.

- Save the ProductUID for use later in this project.

Installing Arduino Libraries

a) Go to the Arduino IDE *Tools* menu, then choose *Manage Libraries*

b) Search for "Blues" in the input box and click the *Install* button next to the "Blues Wireless Notecard" result.

c) Search for MAX6675 and install the library by Yurri Salimov

d) Search for NimBLE and install the library by h2zero

Uploading Arduino Programs onto the ESP32 Dev Kit C v1 and Adafruit Feather

To install Capstone_Sensors.ino onto the ESP32 Dev Kit C:

1. Connect the Dev Kit C to a USB port via micro-USB connector

2. Open Capstone_Sensors.ino in Arduino IDE

3. Navigate to the *Tools* menu in Arduino IDE, then choose *Port* and select the appropriate serial port associated with the newly connected Dev Kit C

4. Within the *Tools* menu, choose *Board* menu, select *ESP Arduino*, and from that menu choose "ESP32 Dev Module"

5. Navigate to the *Sketch* menu in Arduino IDE and select the *Upload* option (alternatively press Ctrl+U). The program will compile and upload onto the ESP32 board. *

To install Capstone_Gateway.ino onto the Adafruit Feather:

1. Connect the Adafruit to a USB port via micro-USB connector (Pro Tip: disconnect other ESP32s from the same computer to avoid confusion as to which serial port to use)

2. Open Capstone_Gateway.ino in Arduino IDE

3. Go to line 8 in the code and use your own ProjectUID to replace the Product UID placeholder within the quotes. As seen below:
   #define productUID "com.your-company.your-name:you_project"

4. Navigate to the *Tools* menu in Arduino IDE, then choose *Port* and select the appropriate serial port associated with the newly connected Adafruit

5. Within the *Tools* menu, choose *Board* menu, select *ESP Arduino*, and from that menu choose "Adafruit ESP32 Feather"

6. Navigate to the *Sketch* menu in Arduino IDE and select the *Upload* option (alternatively press Ctrl+U). The program will compile and upload onto the ESP32 board.

*Depending on the manufacturer, the Dev Kit C might require the user to press and hold the Boot button on the ESP32 board while Arduino IDE tries to connect to it. Release boot button when uploading proceeds.
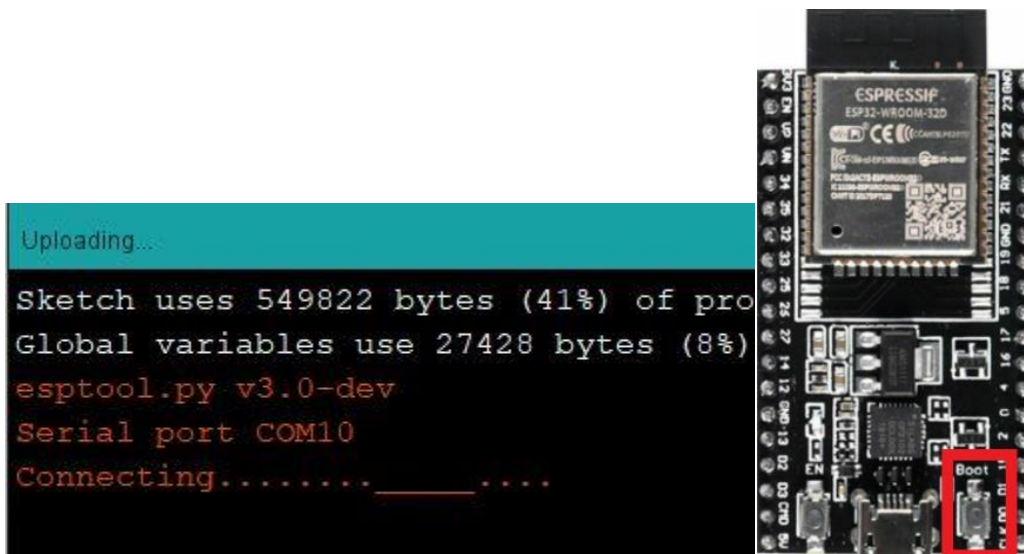
Routing from Notehub to Datacake

- Create a Datacake account

- After creating an account, you should be redirected to your workspace. If you do not get redirected, go to Datacake's main site and click on the Dashboard button at the top right hand corner of the webpage.

- Sign into the dashboard using your newly created credentials

- Click on the Devices from the menu on the left of the dashboard, then click the blue button on the right hand of the dashboard called Add Device.
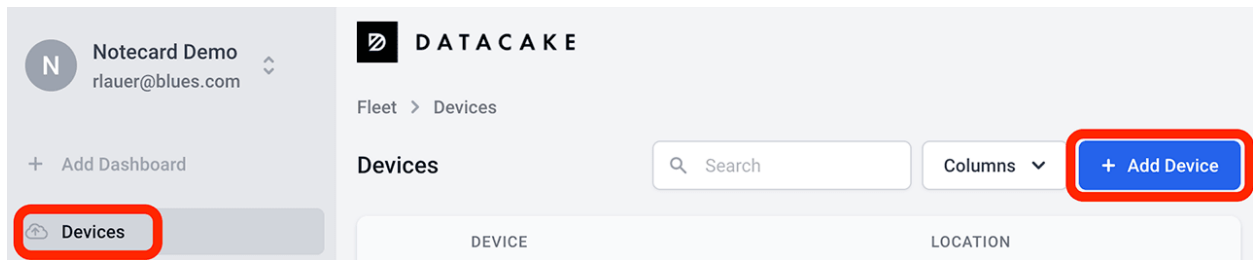
- Choose the API device type, then select New Product and provide a Product Name

- In a separate window, go to sign into Blues Notehub and

  o choose the Project you created when setting up Notehub.

  o Then, choose Devices from the menu on the left and copy the Device UID

- Back on the Datacake device creation screen, paste your Device UID in the Serial Number field and provide a name to label the device.

- Choose the free plan, you can upgrade later

- Once you have created your Datacake device, you should be redirected to an overview of your Datacake devices. Click on the newly created device and open the device view.

- Choose the Configuration tab to configure the device

- Look for the Fields section and click on Add Fields. This will allow you to create numerous database fields to hold data that comes from Notehub. Add one called Temp 1 with Identifier TEMP_1; Time, with Identifier TIME; and, Sound, with Identifier SOUND. All fields should be of type Float.

- Navigate back up the page to the HTTP Payload Decoder section.

- Copy and paste the following code into the code editor, replacing any existing code:

```
function Decoder(request) {
    var data = JSON.parse(request.body);
    var device = data.device;

    var decoded = {};
    decoded.temp1 = data.body.temp1;
    decoded.sound = data.body.sound;
    decoded.time = data.when;

    return [
        {
            device: device,
            field: "TEMP_1",
            value:decoded.temp1
        },
        {
            device: device,
            field: "SOUND",
            value:decoded.sound
        },
        {
            device: device,
            field: "TIME",
            value: decoded.time
        }
    ];
}
```

- Navigate to the HTTP Endpoint URL and copy it

- Back in Notehub, click on Routes in your Projects menu to the left and then click Create Route on the top right corner of the webpage.

- Select the General HTTP/HTTPS Request/Response route type

- Provide a name for the route and paste the HTTP Endpoint URL for the URL

- In the Notefiles dropdown, choose Select Notefiles and choose sensors.qo

- Then, click Create Route to save the route

- Back in Datacake, choose Devices from the menu on the right and select your device from the list in the center of the screen.

- Go to the Dashboard tab and click the toggle button to enable editing for your dashboard

- Click Add Widget

- Choose Chart from the list and create a Title in the Basic tab.

- Within the Data tab, click Add Field and choose Temp 1 from the Field drop down and then click the Save button

- Click Add Widget again and choose Value

- Select Current Value in the Timeframe tab

- Within the Basics tab, create a title for the widget such as Temperature Chart

- Within the Data tab, select Temp 1 from the Field drop down and click Save

Now the setup should be complete. Find a good power source and power the ESP Dev Kit C via its micro USB port. Power the Blues Wireless Notecard via the micro USB port directly on the Notecarrier-AF – not the port of the ESP32 Feather.