LoRaWAN Experience

James H Phelan
jhphelan@hal-pc.org
In association with
Karl Schleicher
k_schleicher@hotmail.com

2020.06.13 -

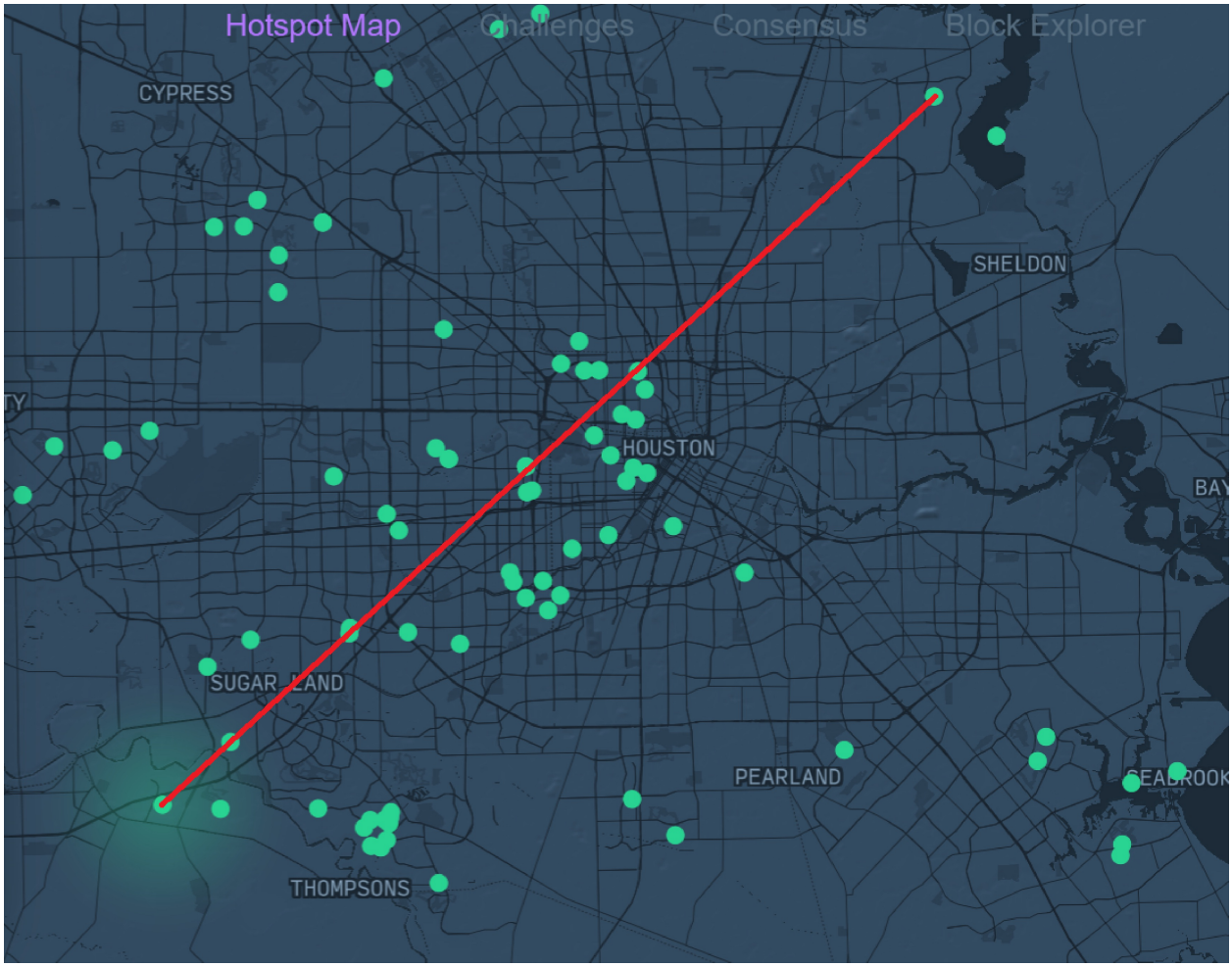
Karl Schleicher first announced (~2020.03.01) to the USAi Labs group a Gas Sensor Project:
C:\Users\Me\Documents\My Downloads\Robotics\USAi Labs\LoRaWAN Project
consisting of using a city-wide network of various gas sensors
https://www.amazon.com/gp/product/B07D9H74LT/ref=ppx_yo_dt_b_asin_title_o00_s00?ie=UTF8&psc=1
using low power, small data packet, spread-spectrum radios called LoRa or LoRaWAN:
https://en.wikipedia.org/wiki/LoRa
and the Helium network:
https://www.helium.com/lorawan?gclid=EAIaIQobChMI8YSd8dL_6QIVCkqGCh0qeQxREAAYASAAEgIe9PD_BwE

On 2020.05.29 Karl sent me an Arduino Mega 2560 with a Dragino LoRa shield v1.1 “device” pre-programed with a “Hello, world!” data message. He was not able to see it on his dashboard as the nearest Helium hotspot is across Lake Houston 3 mi away by straight-line Google Maps measurement.

On 2020.06.02 Karl sent me info on how to request a free hotspot from emrit.io: https://www.emrit.io/

I did and we both received hotspots on 2020.06.11.
Set mine up according to directions but get the error message:
"You do not own this hotspot and cannot add it to your wallet."
We suspect that the hotspots are owned by EMRIT and they use them to mine Helium data credits which they share in some unclear way with the hotspot hosts. Or maybe I first have to email EMRIT (success@emrit.io) that I successfully installed the hotspot which I did today.

Plugging in his Arduino/Dragino device during the USAi Labs virtual meeting (COVID-19 pandemic) he was able to see the data packets from the device via my hotspot “Nutty Magenta Viper” located in Humble, TX 77346. Karl’s hotspot “Smooth Fern Ardmadillo” is located diagonally across Houston in Richmond, TX. Except for “Dandy Chili Goblin” located across Lake Houston on Lake Breeze Ln, Crosby, TX, Karl and I are the farthest apart hotspots on the NE-SW diagonal!



I created my own Helium dashboard. I was able to add the Arduino/Dragino device using the following IDs:

WLAN:60:81:F9:6A:66:88
ETH:60:81:F9:6A:66:86
wallet address:
14DN6yWS2YLaXQauJMHZCXMBv4hCTRcoHBVmaNmX93LyTxqvWS2

but I could see no data from it, even tho it was visible on Karl's dashboard during today's (2020.06.13) USAi Labs meeting.

Karl send me the arduino_mega_dragino_hello.ino Arduino sketch but I was unable to load it due to missing libraries.
Karl advised –

In the Arduino IDE follow the menus:
sketch->include library-> manage libraries
search for "mcci lorawan lmic"
Select the library by IBM, Matthis Kooljman et al
I am using the latest - version 3.2.0
I think that is all you need

That fixed the invalid library errors.

I think I used Sketch / Add .ZIP Library / (as with the OSR Arduino LED display) and either the

arduino-LoRx-master_20200220.zip or
LoRa_Shield_Sketch_For_MQTT.zip

I don't recall. That may have been unnecessary with the above include library.

I had errors uploading to the Arduino until I recognized I needed to change the Board setting from the Uno of the Open Source Rover LED display head to the Mega 2560 of the LoRa device. Afterward it loaded ok.0

Noticed that the sensors from Amazon above are labeled "Flying Fish". Googled "flying fish gas sensor" and found a number of gas sensor projects that should be adaptable.

I want to adapt the Arduino sketch to change the data message to "Hello from Humble, TX!"

```
21      static uint8_t mydata[] = "Hello from Humble, Texas!";
```

but wasn't sure if there's a variable for the message length that also needs to be changed. Apparently not, as later in the code it determines the length of mydata:

```
163     LMIC_setTxData2(1, mydata, sizeof(mydata) - 1, 0);
```

I also want to change the data send interval from 10 seconds to 30 seconds and later to 1 minute to save data transmission cost as that should be more than frequent enough to monitor air quality for a gas leak warning system as the human response will be much slower than that! That is done here:

```
24      // THIS IS THE TIMING INTERVAL IN SECONDS
25      // ORIGINAL = 10 seconds
26      // Schedule TX every this many seconds (might become longer due to duty
27      // cycle limitations).
28      const unsigned TX_INTERVAL = 30;
```