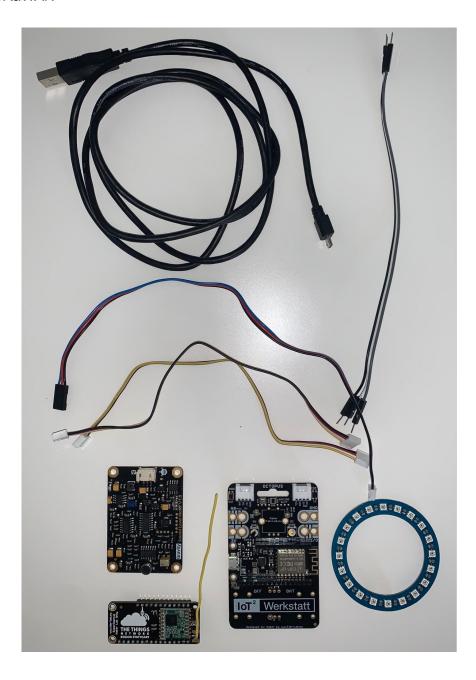
# Dokumentation zur Entwicklung der Ampel

## Was wird benötigt?

- 1 X IoT Octopus
- 1 X LoRa-Wings mit verlöteter Antenne
- 1 X Analog Sound Level Meter
- 1 X Grove RGB LED Ring (20 WS2813 Mini)
- 2 X Jumper Kabel (digital)
- 1 X Jumper Kabel (analog)
- 3 X einfache Kabel
- 1 X Mini USB Kabel
- 1 X LoRaWAN Gateway siehe Anleitung Dokumentation zur Verwendung von LoRaWAN



#### Allgemeine Hinweise und Funktionsweise der Ampel

Nachfolgend wird beschrieben wie die Hardware vorbereitet und der Code übertragen werden kann. Hierfür werden zunächst die Schritte erläutert und nachfolgend auch durch Bilder/Screenshots dargestellt.

Die Ampel misst die Lautstärke alle 250 Millisekunden. Je nach eingestellten Grenzwerten und aktueller Lautstärke leuchtet die Ampel rot, gelb oder grün. Alle 60 Sekunden wird die durchschnittliche Lautstärke mittels LoRaWAN versendet. Wenn dies geschieht leuchtet die Ampel blau, während dieser Zeit wird keine Lautstärke gemessen.

#### Hardware vorbereiten

Wir beginnen damit alle Bauteile an den Octopus anzuschließen. Hierbei wie folgt vorgehen:

- 1. LoRa-Wings auf den Octopus stecken.
- 2. Den LED Ring mit einem digitalen Jumper-Kabel mit dem Grove I<sup>2</sup>C Anschluss (oben rechts) verbinden.
- 3. Das analoge Jumper Kabel mit den Sound Sensor verbinden und das digitale Jumper Kabel mit dem analogen Anschluss des Octopus verbinden.
- 4. Nun das analoge und digitale Jumper Kabel miteinander verbinden, hierzu die Beschriftung am Anschluss des Sound Sensors und Octopus beachten: (Achtung die Farben können je nach Kabel unterschiedlich sein)

Beschriftung am Sound Sensor	Beschriftung am Octopus
A (blau)	A0 (gelb)
+ (rot)	3V (rot)
- (schwarz)	GND (schwarz)
nicht verbinden	NC (weiß)

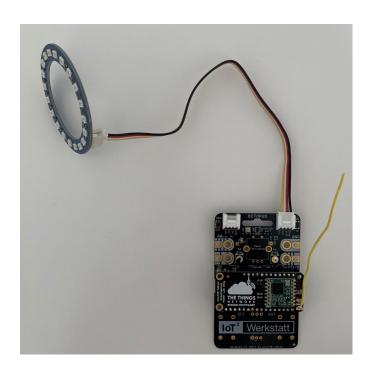
5. Das Mini USB Kabel mit dem Octopus verbinden

Fotos zu den einzelnen Schritten:

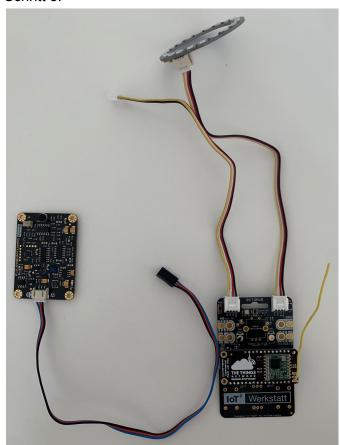
#### Schritt 1:



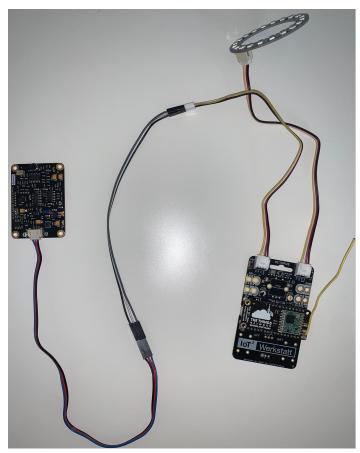
Schritt 2:



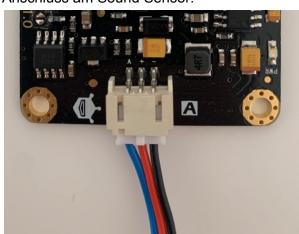
Schritt 3:

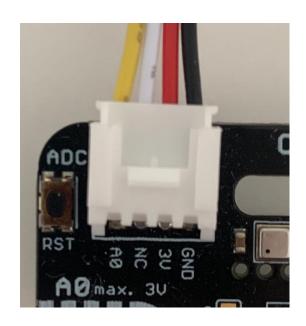


Schritt 4:

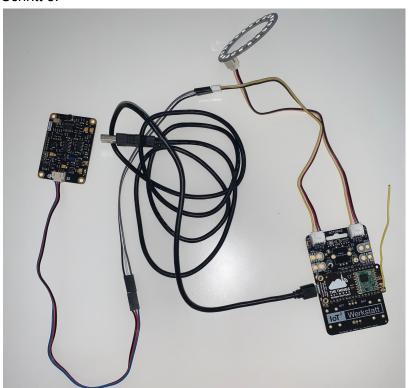


Anschluss am Sound Sensor:





## Schritt 5:



#### Code kopieren, vorbereiten und übertragen

Nachdem die Hardware miteinander verbunden wurde können wir den Code auf den Octopus übertragen. Hierbei wie folgt vorgehen:

<u>Achtung:</u> Bevor diese Schritte durchgeführt werden, sollte die Dokumentation zur Verwendung von LoRaWAN abgeschlossen sein.

- 1. Code kopieren:
  - https://github.com/MevreHTW/Lernampel/blob/main/Ampel/Code\_Ampel.txt
- 2. Arduino IDE öffnen und eine neue Datei erstellen.
- 3. Alles markieren (Strg + A) und den Code einfügen.
- 4. Die Datei abspeichern.
- 5. Falls gewünscht die Grenzwerte für die Lautstärke anpassen (green, yellow, red).
- 6. Auf der Seite thethingsnetwork.org das gewünschte end device auswählen.
- 7. Im Bereich Overview die AppEUI, DevEUI und den AppKey kopieren und im Code einfügen. Achtung, die richtige Formatierung (<>) und Sortierung Isb beachten!
- 8. Die Datei erneut speichern.
- 9. In der Arduino IDE unter Werkzeuge den Upload Speed auf 115200 stellen und den richtigen Port auswählen.
- 10. Das LoRaWAN Gateway an den Strom anschließen
- 11. Den Octopus über das USB Kabel mit dem Computer verbinden und den Code auf den Octopus übertragen.
- 12. Wenn die Übertragung abgeschlossen ist wird in der Konsole "Hard resetting via RTS pin ..." angezeigt.
- 13. Auf Werkzeuge → Serieller Monitor klicken um die Ausgaben zu überwachen.
- 14. Auf der Seite <u>hethingsnetwork.org</u> sollten die gesendeten Daten im Bereich Live data angezeigt werden.

## Screenshots zu den einzelnen Schritten:

#### Schritt 1:



### Schritt 2:

```
sketch_aug24a | Arduino 1.8.13
Datei Bearbeiten Sketch Werkzeuge Hilfe

sketch_aug24a

void setup() {
    // put your setup code here, to run once:
}

void loop() {
    // put your main code here, to run repeatedly:
```

### Schritt 4:

sketch\_aug24a | Arduino 1.8.13 Datei Bearbeiten Sketch Werkzeuge Hilfe Neu Strg+N Öffnen... Strg+O Letzte öffnen Sketchbook Beispiele Schließen Strg+W Speichern Strg+S Speichern unter... Strg+Umschalt+S Seite einrichten Strg+Umschalt+P Drucken Strg+P Voreinstellungen Strg+Komma Beenden Strg+Q

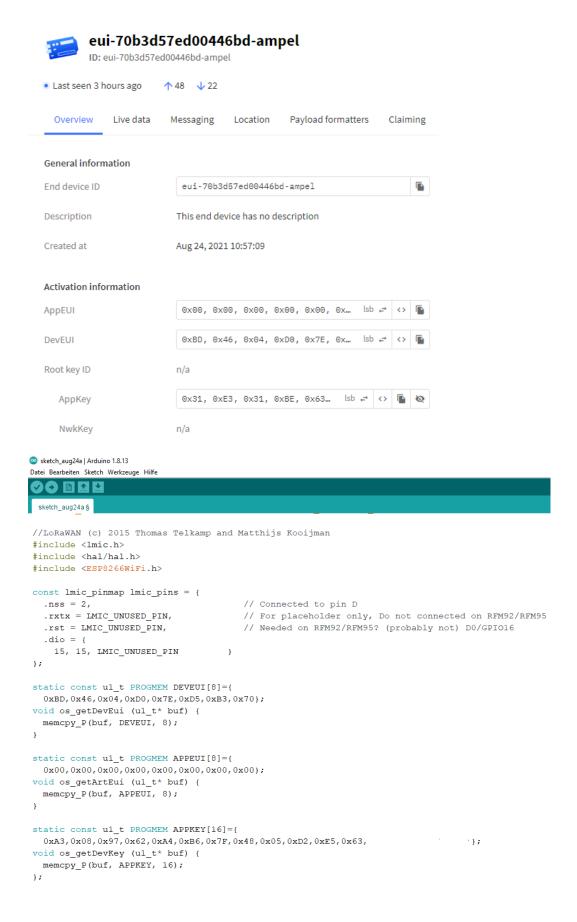
#### Schritt 5:

```
Ampel §
// Sound Sensor
#define SoundSensorPin A0
#define VREF 5.0
float voltageValue, dbValue;
int intDbValue;
int sumDbValues = 0;
int numberOfValues = 0;
int averageDbValue = 0;
// Diese Werte (in db) anpassen um die Grenzen der Ampel zu verändern
// Ein Geräuschpegel unter diesem Wert wird als grün angezeigt
float green = 70.0;
// Ein Geräuschpegel unter diesem Wert und über dem green Wert wird als gelb angezeigt
float yellow = 80.0;
// Ein Geräuschpegel über diesem Wert wird als rot angezeigt (sollte dem yellow Wert entsprechen)
float red = 80.0;
```

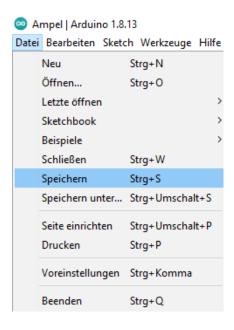
### Schritt 6:



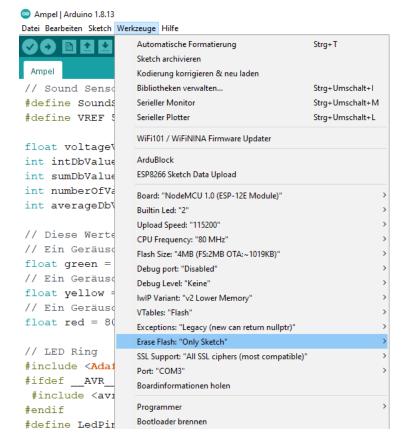
### Schritt 7:



#### Schritt 8:



#### Schritt 9:



## Schritt 10:

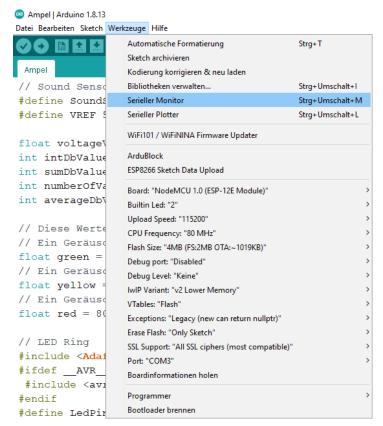


#### Schritt 11:

```
Hochladen abgeschlossen.

Leaving...
Hard resetting via RTS pin...
```

### Schritt 12:



## Schritt 13:

