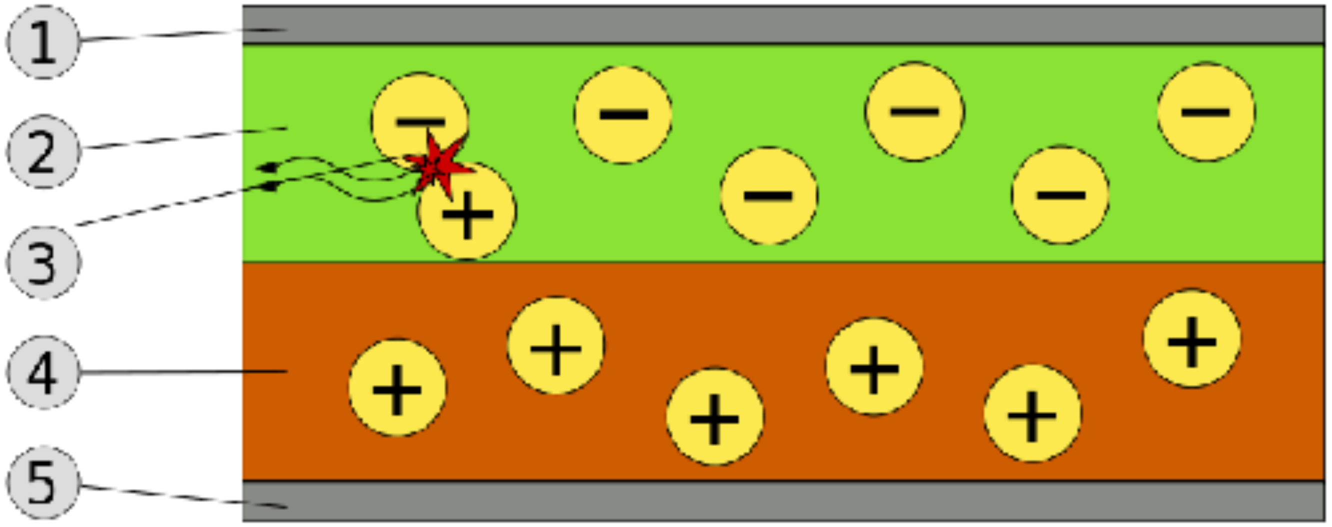
BARSicle

3. Arduino
OLED displays

What is OLED?



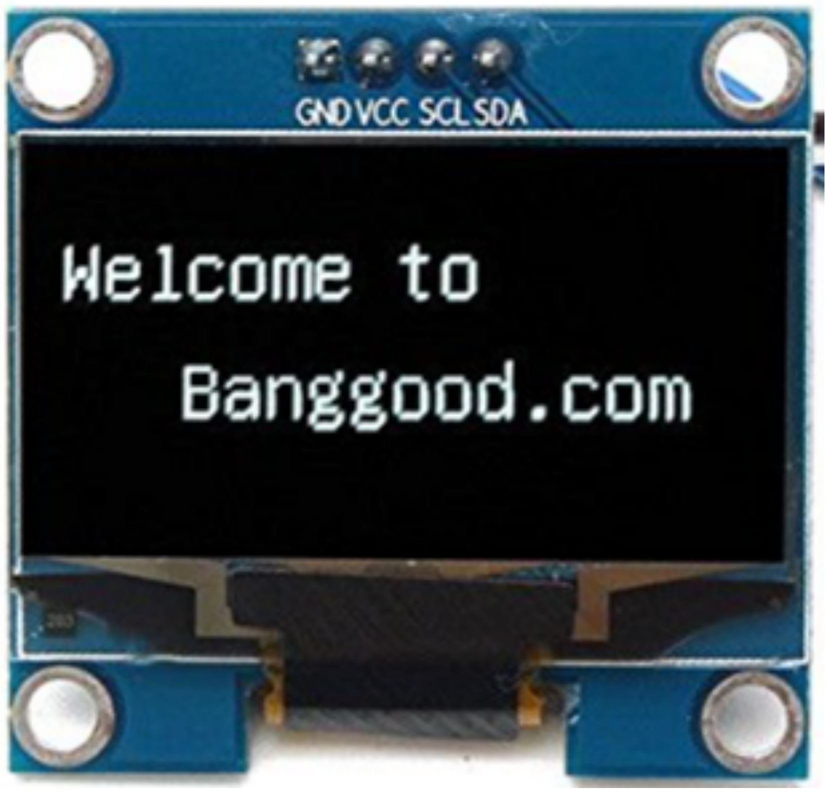
Schematic of a bilayer OLED: 1. Cathode (−), 2. Emissive Layer, 3. Emission of radiation, 4. Conductive Layer, 5. Anode (+)

- An **organic light-emitting diode** (**OLED**) is a light-emitting diode (LED) in which the emissive electroluminescent layer is a film of organic compound that emits light in response to an electric current.

- This layer of organic semiconductor is situated between two electrodes; typically, at least one of these electrodes is transparent.

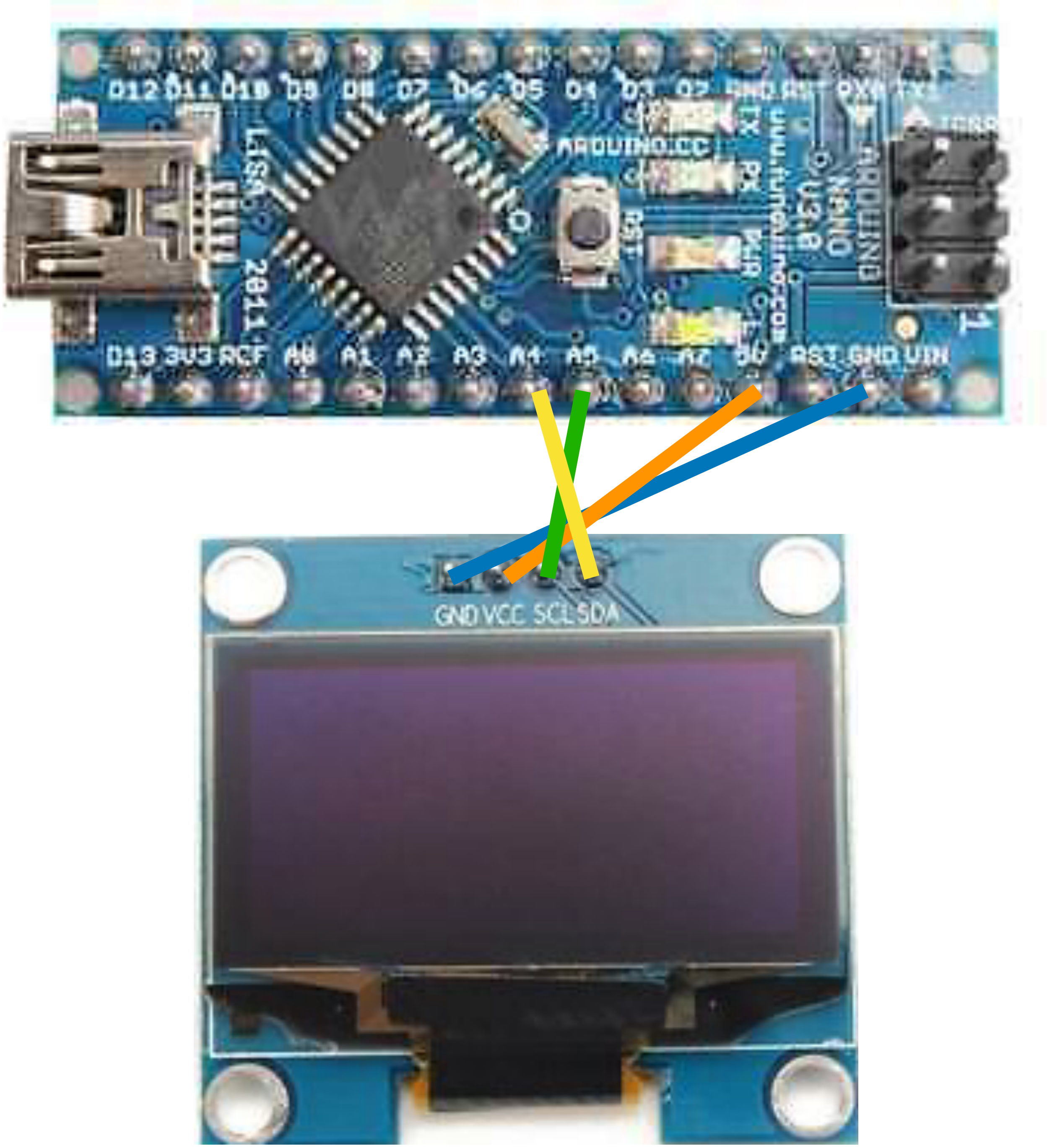
OLED 1.3”

- 5V operation
- I2C bus
  - SCL clock
  - SDA data
- 128 x 64 pixels
- We will use type with SH1106 controller, alternative is 12864 type
  (would mean change to Oled.h header for different type)



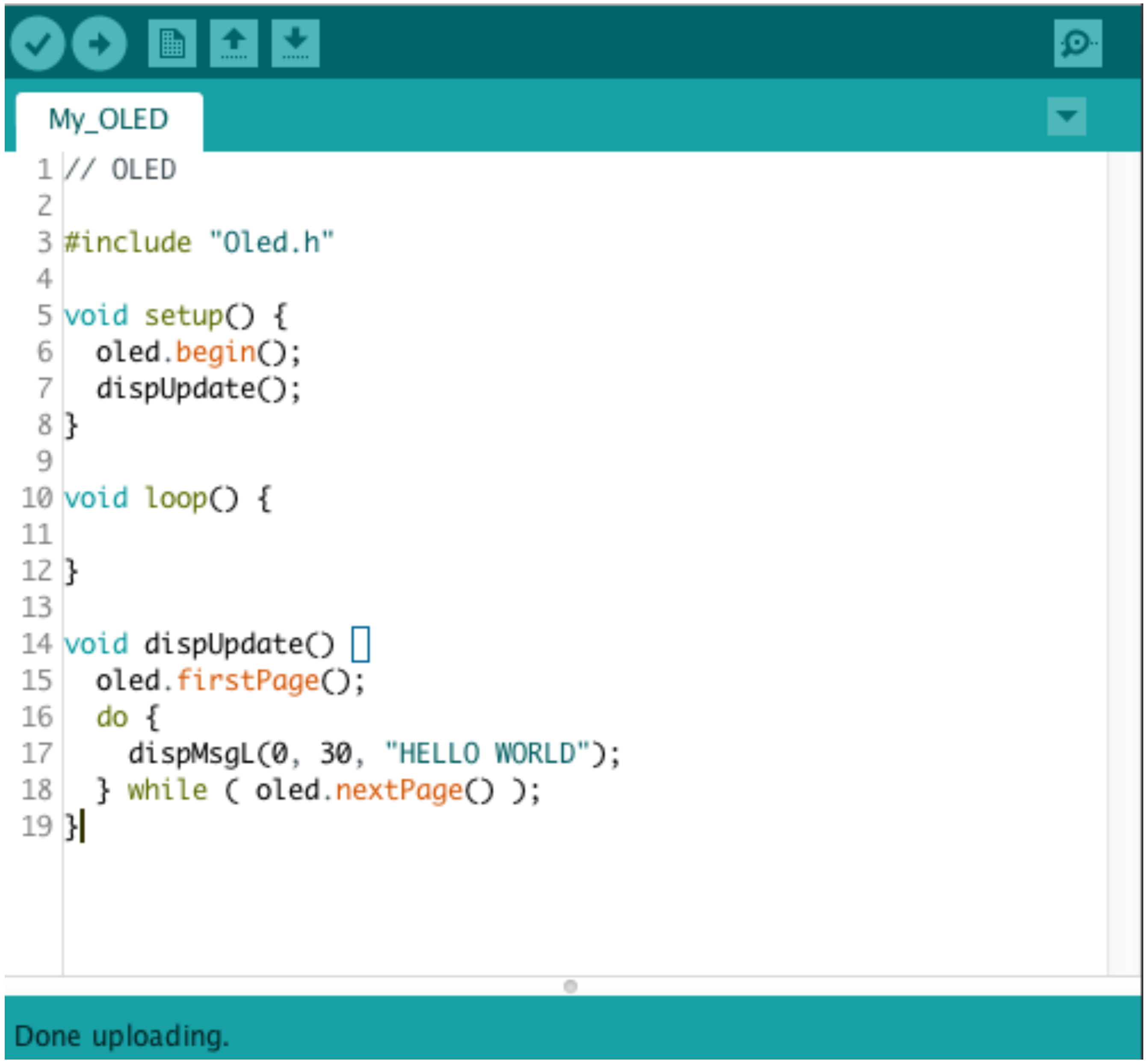
Wiring

- Nano -> OLED
  - GND -> GND (blue)
  - 5V -> VCC (orange)
  - A5 -> SCL (green)
  - A4 -> SDA (yellow)
- A4 & A5 are the I2C serial bus


D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 GND RST RX0 TX1
ARDUINO.CC
www.funduino.com
ARDUINO NANO V3.0
D13 3V3 REF A0 A1 A2 A3 A4 A5 A6 A7 5V RST GND VIN
GND VCC SCL SDA

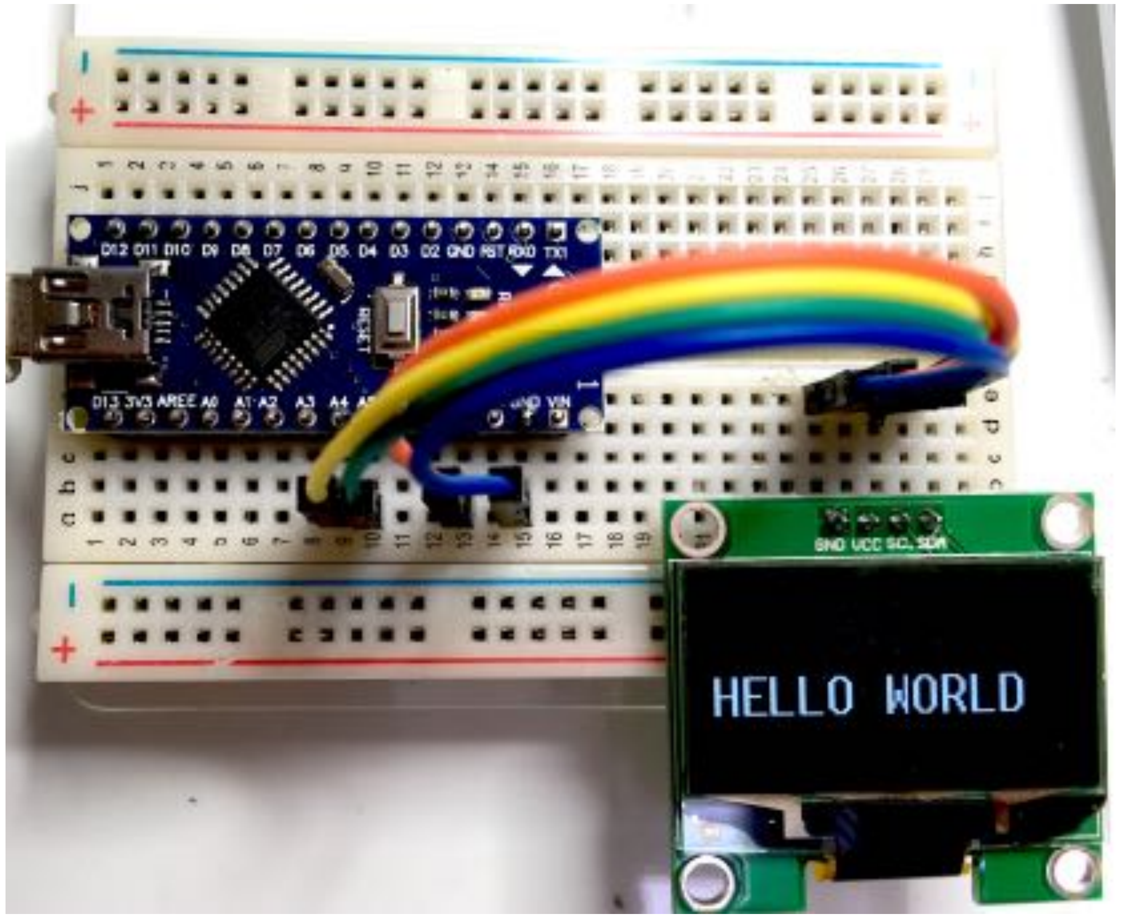
Say HELLO

- *File > Sketchbook > My_OLED*



```
// OLED

#include "Oled.h"

void setup() {
  oled.begin();
  dispUpdate();
}

void loop() {

}

void dispUpdate() {
  oled.firstPage();
  do {
    dispMsgL(0, 30, "HELLO WORLD");
  } while ( oled.nextPage() );
}
```

Done uploading.

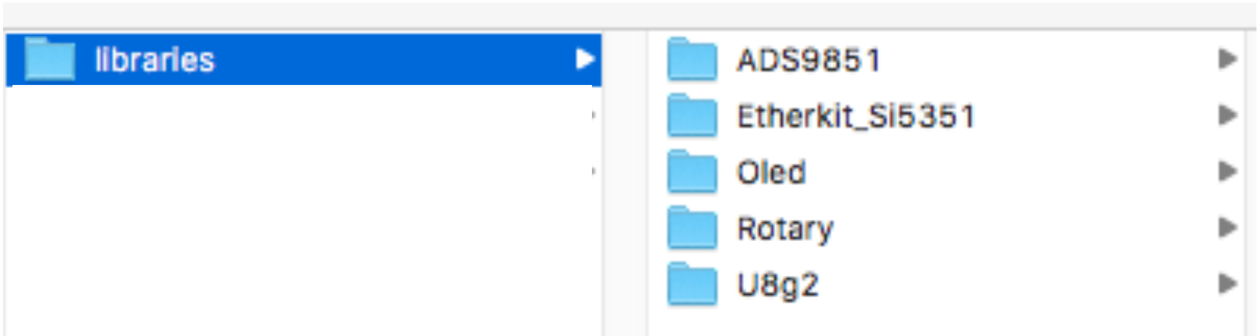


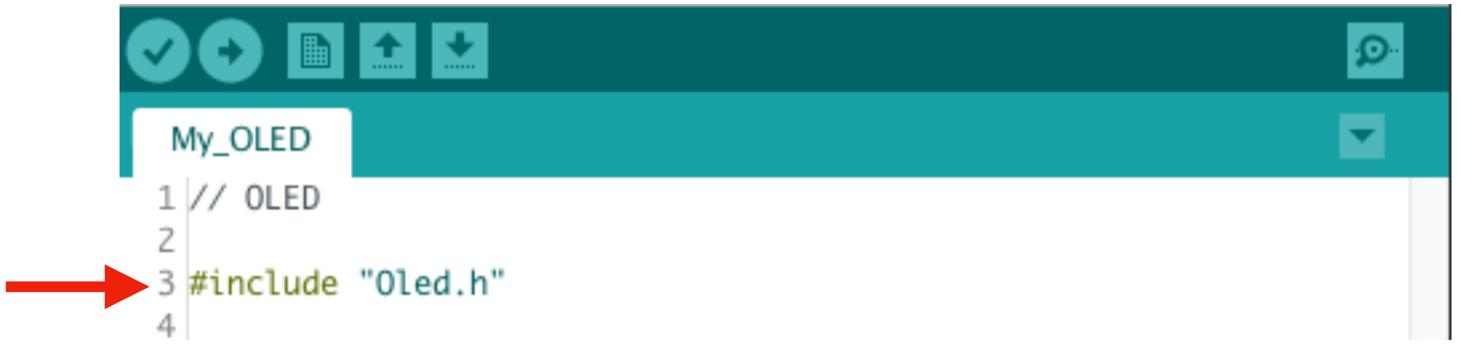
A4 A5 +5 GND **GND +5 A5 A4**

libraries & headers

- *libraries* contain code you can re-use in your sketch
- Here we *include* a set of functions from *Oled.h*
- This make it easy to display text, numbers, frequencies, date & time, bar graphs
- You can look at Oled.h. Go to libraries and open in text editor !!!DO NOT MODIFY!!!

Arduino >





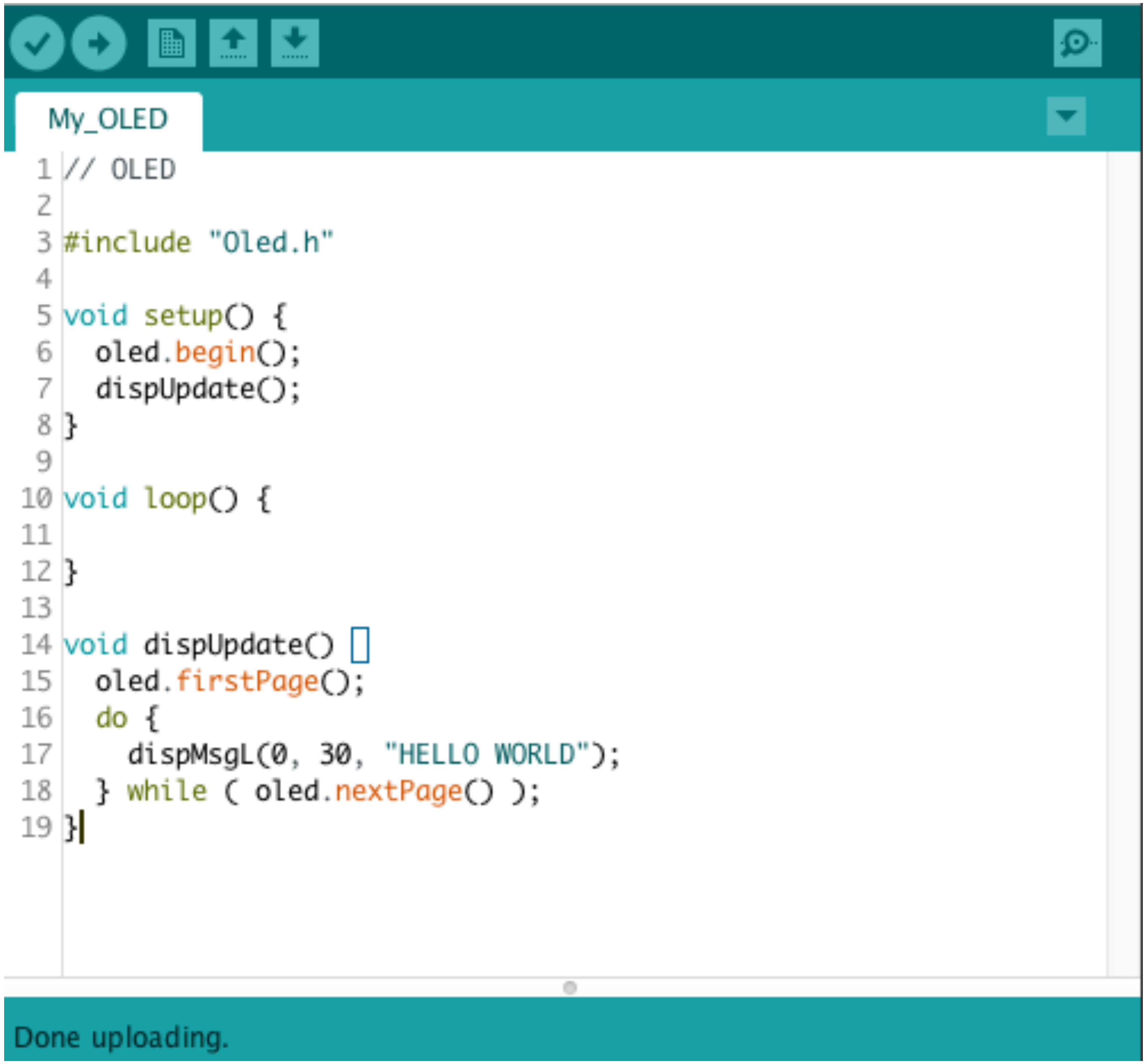
```
// OLED

#include "Oled.h"
```

Functions in Old.h

```
// functions & usage
void dispBar(u8g2_uint_t x, u8g2_uint_t y, byte h, byte l)
void dispScn(u8g2_uint_t sx, u8g2_uint_t sy, uint64_t *a)
void setPix(u8g2_uint_t x, u8g2_uint_t y, uint64_t *a)
void dispFreq(u8g2_uint_t x, u8g2_uint_t y, double f, double cf, byte d)
void dispStep(u8g2_uint_t x, u8g2_uint_t y, unsigned int s)
void dispMsgS(u8g2_uint_t x, u8g2_uint_t y, char *m)
void dispMsg(u8g2_uint_t x, u8g2_uint_t y, char *m)
void dispMsgL(u8g2_uint_t x, u8g2_uint_t y, char *m)
void dispMsgUL(u8g2_uint_t x, u8g2_uint_t y, char *m)
void dispNum(u8g2_uint_t x, u8g2_uint_t y, double n, byte d)
void dispNumL(u8g2_uint_t x, u8g2_uint_t y, double n, byte d)
void dispNumUL(u8g2_uint_t x, u8g2_uint_t y, double n, byte d)
void dispDate(u8g2_uint_t x, u8g2_uint_t y, byte dw, byte da, byte mo, byte yr)
void dispTime(u8g2_uint_t x, u8g2_uint_t y, byte h, byte m, byte s)
void dispTimeL(u8g2_uint_t x, u8g2_uint_t y, byte h, byte m, byte s)
```

Oled.h

- *#include "Oled.h"* adds its functions to your sketch
- So you can use them
- Next we will take a look inside *Oled.h*

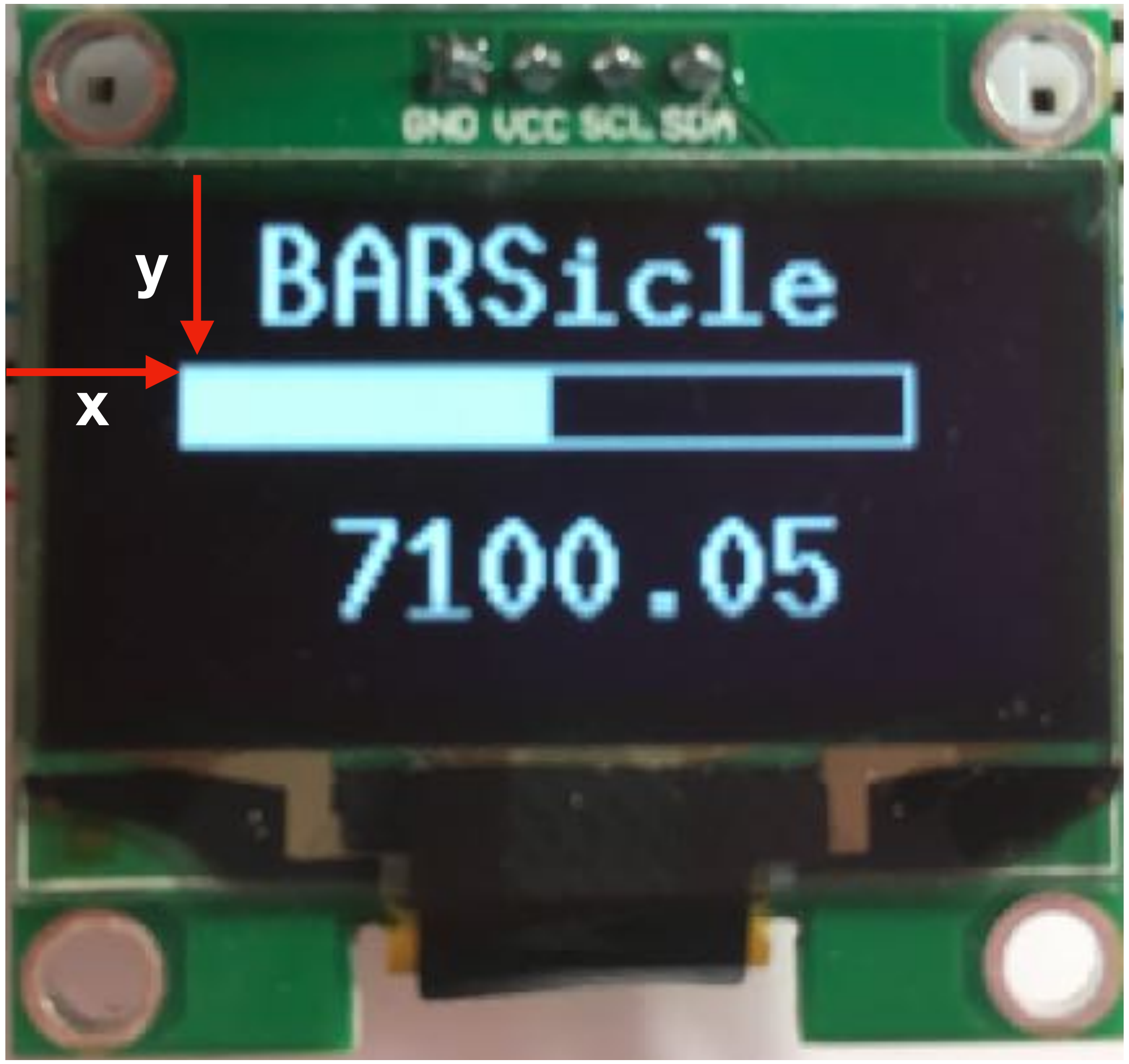


```
// OLED

#include "Oled.h"

void setup() {
  oled.begin();
  dispUpdate();
}

void loop() {

}

void dispUpdate() {
  oled.firstPage();
  do {
    dispMsgL(0, 30, "HELLO WORLD");
  } while ( oled.nextPage() );
}
```

Done uploading.

Examples

```
void dispUpdate() {
  oled.firstPage();
  do {
    dispMsgL(20, 0, "BARSicle");
    dispBar(10, 20, 10, 50);
    dispNumL(30, 40, 7100.05, 2);
  } while ( oled.nextPage() );
}
```

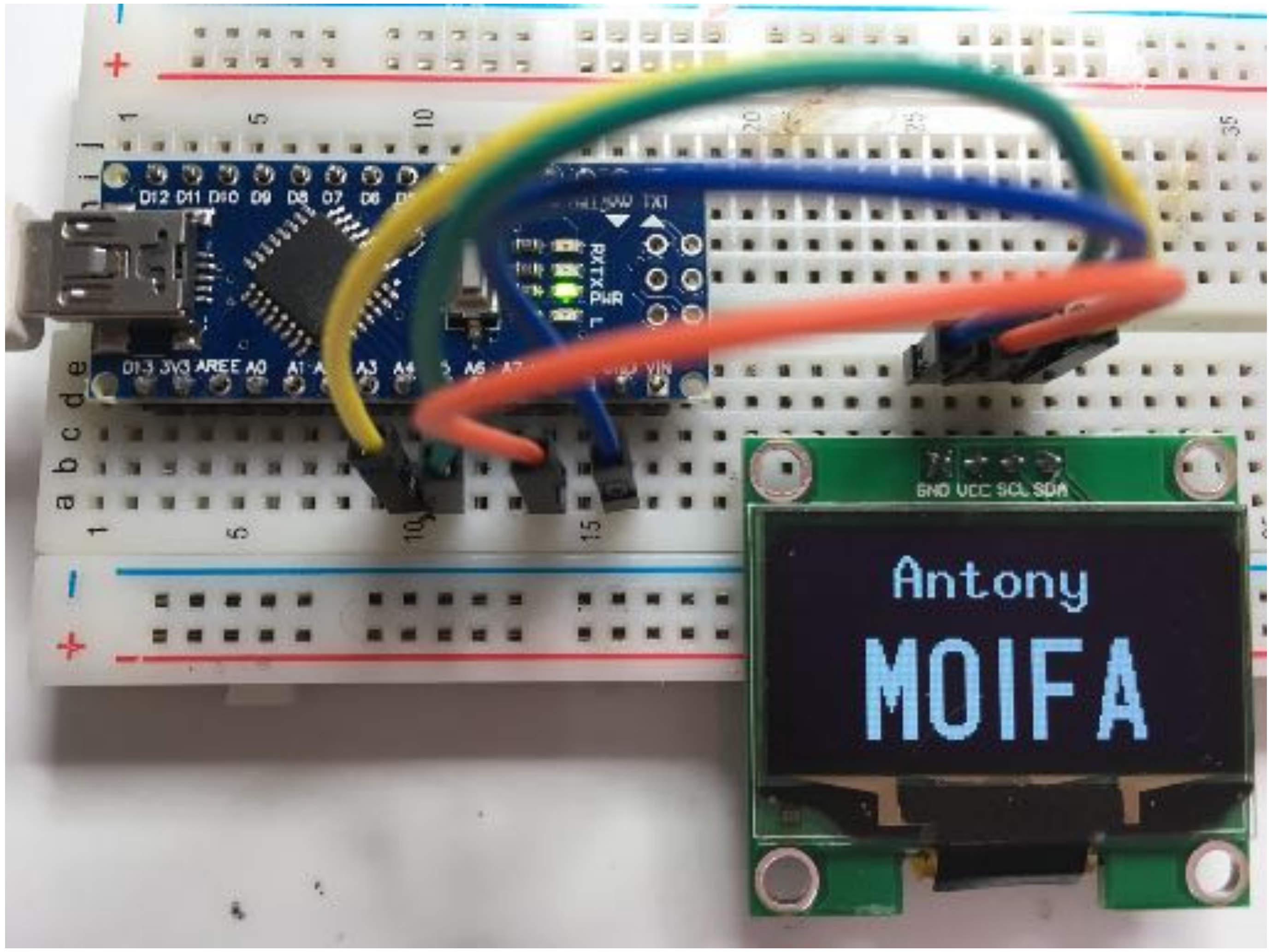


Name and call sign

- *File > Sketchbook > My_OLED_NAME_CALL*

```
}
void dispUpdate() {
  oled.firstPage();
  do {
    dispMsgL(30, 5, "Antony");
    dispMsgUL(20,30, "M0IFA");
  } while ( oled.nextPage() );
}
```

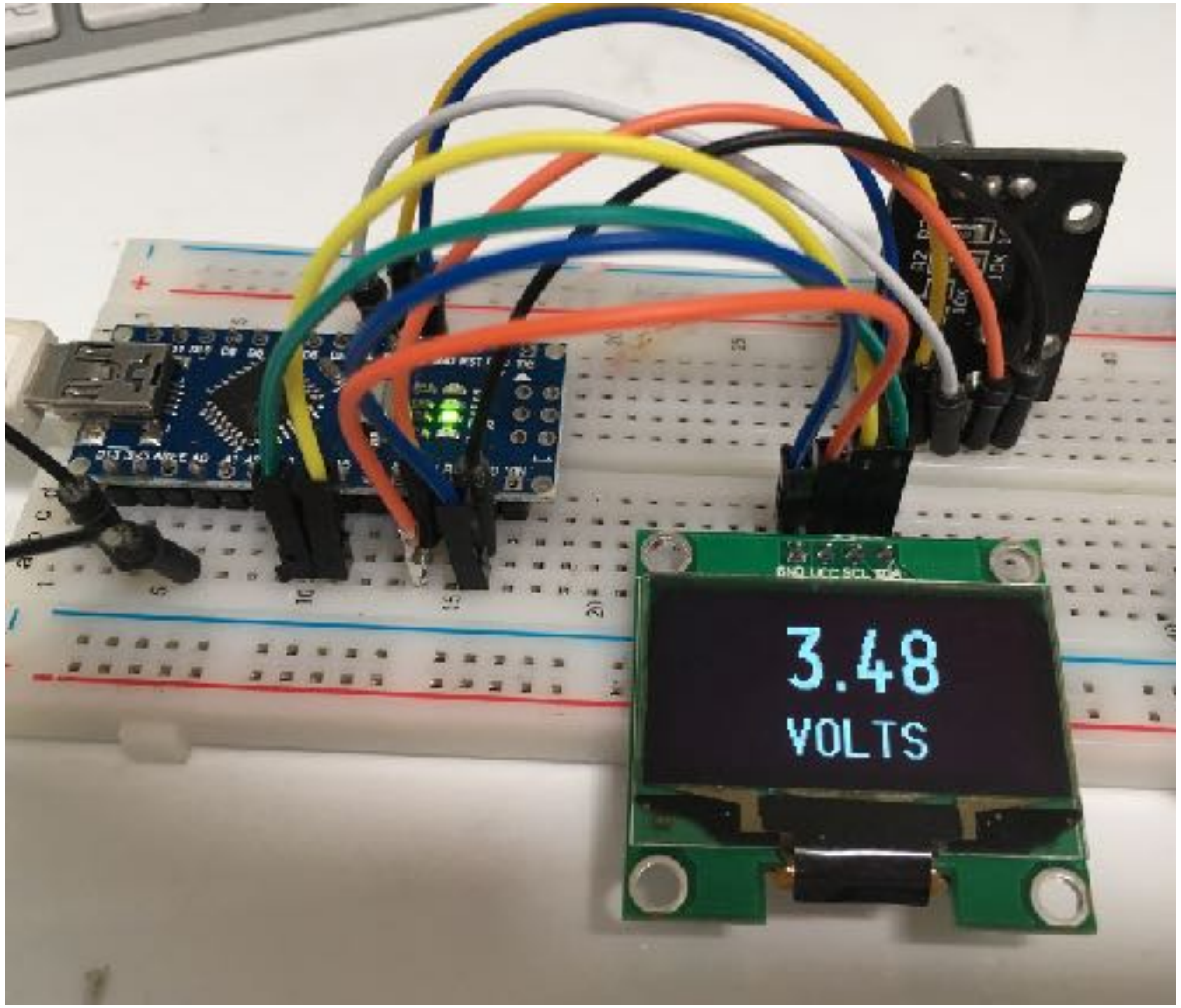
Edit to insert your name and callsign



Volts with OLED

- *File > Sketchbook > My_VOLTS_OLED*

- A version of the voltmeter but with OLED display



Now you can use an OLED display

Home work: look up u8g2 on Github