

Components

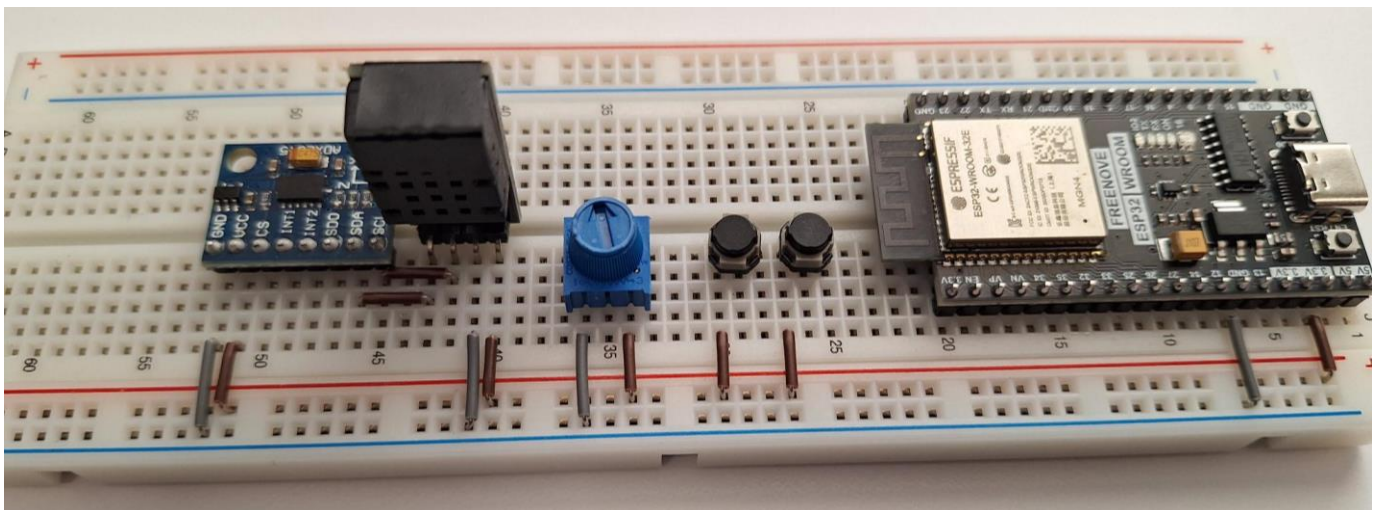
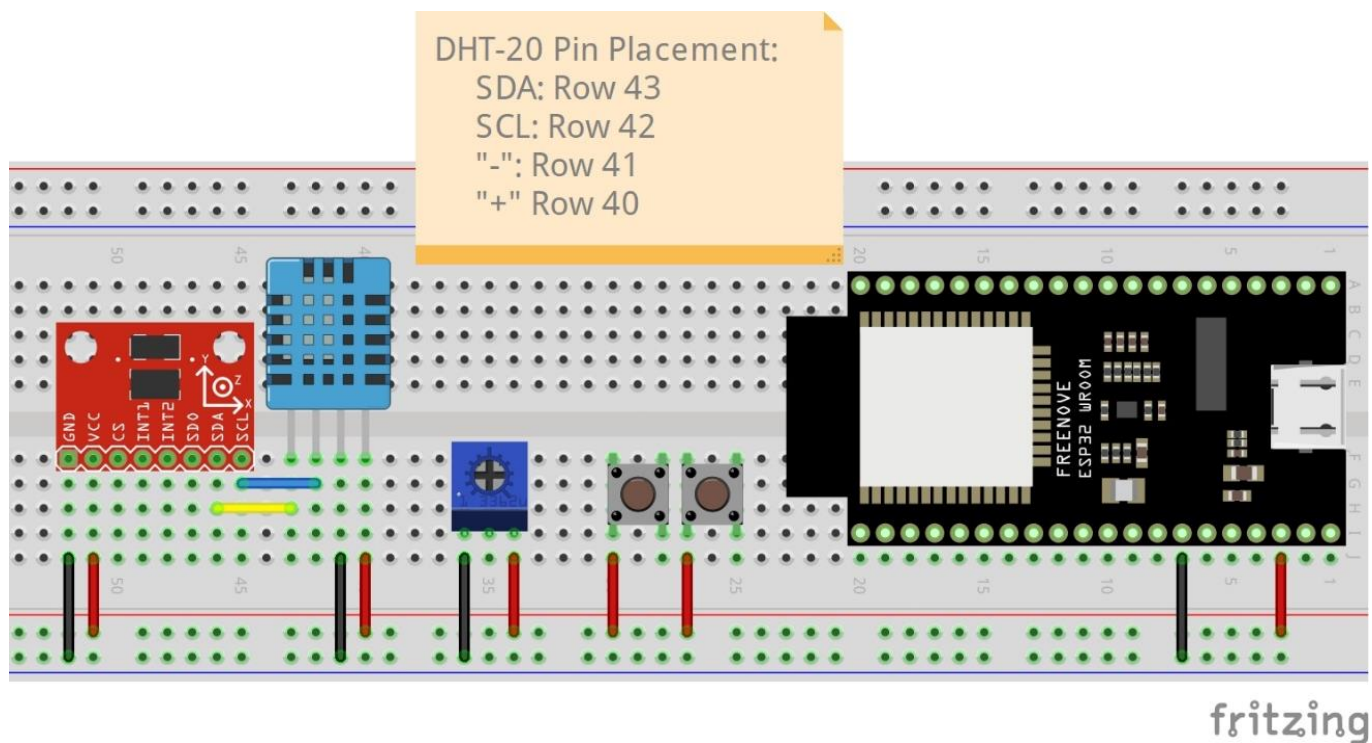
There is enough space on the breadboard to wire all the kit components together, at the same time.

However, if you are also planning to use additional sensors, some kit sensors may need to be disconnected.

Wiring

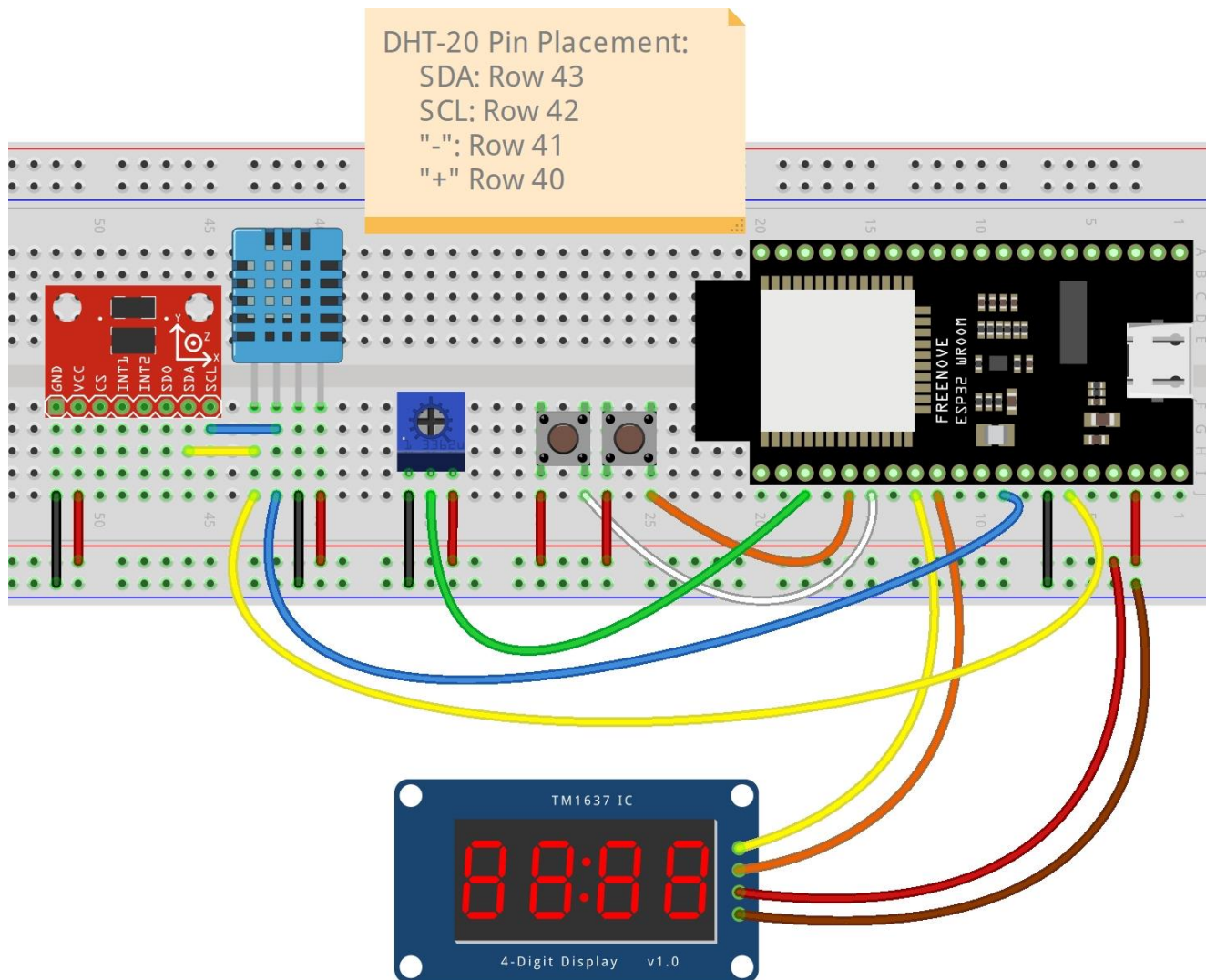
Start by wiring power and ground to all sensors. Your instructor can supply you with pre-made short-length jumpers to make the wiring neater. The 4-digit LED is mounted off the breadboard.

Step 1 – Power Wiring



Step 2 – Add Signal Wiring

Here is the completed Wiring diagram. Notice that the ADXL345 and DHT-20 share the I2C communication bus.



Libraries

- Adafruit AHTX0 by Adafruit
- TM1637 by Avishay
- Adafruit ADXL345 by Adafruit
- Freenove WS2812 Lib for ESP32 by Freenove (**MUST USE VERSION 1.0.6**).

Check for the following entries in your **platformIO.ini** file after adding these libraries:

```
lib_deps =  
adafruit/Adafruit AHTX0@^2.0.5  
smougenot/TM1637@0.0.0-alpha+sha.9486982048  
adafruit/Adafruit ADXL345@^1.3.4  
freenove/Freenove WS2812 Lib for ESP32@1.0.6
```

Sample Code

```
// *****
// COMP-10184 - IoT Programming
// Mohawk College
//
// Sample code for all devices/sensors in the 2025 Lab Kit
//

#include <Arduino.h>
#include <Wire.h>
#include <Adafruit_AHTX0.h>
#include <Adafruit_ADXL345_U.h>
#include <TM1637Display.h>
#include "Freenove_WS2812_Lib_for_ESP32.h"

// *****
// GPIO definitions
#define PIN_BTN1      35
#define PIN_BTN2      34
#define PIN_BLUE_LED  2
#define PIN_POT        36
#define PIN_TOUCH7    27
#define PIN_CLK        33    // CLK pin connection to the display
#define PIN_DIO        25    // DIO pin connection to the display
#define PIN_RGB_LED    16    // RGB LED

// we need to use non-standard I2C pins..
#define I2C_SDA        13
#define I2C_CLK        14

// RGB LED
#define RGB_COUNT      1    // number of RGB LED's
#define RGB_CHANNEL    0    // RGB LED Channel

// *****
// devices and sensors...
Adafruit_AHTX0      aht;    // DHT-20
// Accelerometer sensor. assign a unique ID. Helpful if you are using

// more than sensor at a time.
Adafruit_ADXL345_Unified adxl = Adafruit_ADXL345_Unified(12345);

// 4-digit display
TM1637Display      tm(PIN_CLK, PIN_DIO);
```

```
// RGB LED (WS2812)
Freenove_ESP32_WS2812 rgb_led = Freenove_ESP32_WS2812(RGB_COUNT, PIN_RGB_LED, RGB_CHANNEL,
TYPE_GRB);

// *****
void setup() {
  // init serial monitor
  Serial.begin(115200);

  // set pins for I2C communications.. Used for DHT20 and ADXL345
  Wire.begin(I2C_SDA, I2C_CLK);

  // configure DHT-20
  if (!aht.begin()) {
    Serial.println("No DHT-20 detected. Check wiring..");
    while (1);
  }
  Serial.println("DHT20 found!");

  // init adxl345
  if(!adxl.begin())
  {
    Serial.println("No ADXL345 detected. Check wiring..");
    while(1);
  }
  /* Set the range to whatever is appropriate for your project */
  adxl.setRange(ADXL345_RANGE_16_G);
  // accel.setRange(ADXL345_RANGE_8_G);
  // accel.setRange(ADXL345_RANGE_4_G);
  // accel.setRange(ADXL345_RANGE_2_G);

  // 4-DIGIT LED setup. brightness=0 (dim), 7=max
  tm.setBrightness(2);
  // clear display
  tm.clear();

  // init RGB LED
  rgb_led.begin();
  // set initial brightness. Use 0-255 here.
  rgb_led.setBrightness(20);

  // set GPIO's
  pinMode(PIN_BTN1, INPUT_PULLDOWN);
```

```

    pinMode(PIN_BTN2, INPUT_PULLDOWN);
    pinMode(PIN_BLUE_LED, OUTPUT);

}

// *****
// print push button states
void doBtns() {
    int iBtn1, iBtn2;

    iBtn1 = digitalRead(PIN_BTN1);
    iBtn2 = digitalRead(PIN_BTN2);

    Serial.printf("Btn1 = %d, Btn 2 = %d\n", iBtn1, iBtn2);
}

// *****
// print variable resistor value..
void doPot() {
    int iPot;

    iPot = analogRead(PIN_POT);

    Serial.printf("Variable Resistor = %d\n", iPot);
}

// *****
// BLUE LED
void doBlueLED() {
    static boolean bState = LOW;

    digitalWrite(PIN_BLUE_LED, bState ? HIGH : LOW);

    // flip state
    bState = !bState;
}

// *****
void doRGBLED() {
    static int iMode = 0;

    // WS2812 device has 3 internal LED's (red, green blue).
    // Each led intensity can vary from 0-255 for a total
    // of 255x255x255 = ~16 million colours.
    //
    switch (iMode) {
        case 0:

```

```

    // red led...
    // Set color data:      r      g      b
    rgb_led.setLedColorData(0, 0xff, 0x00, 0x00);
    break;
case 1:
    // green led...
    // Set color data:      r      g      b
    rgb_led.setLedColorData(0, 0x00, 0xff, 0x00);
    break;
case 2:
    // red led...
    // Set color data:      r      g      b
    rgb_led.setLedColorData(0, 0x00, 0x00, 0xff);
    break;
case 3:
    // all leds (white)...
    // Set color data:      r      g      b
    rgb_led.setLedColorData(0, 0xff, 0xff, 0xff);
    break;
default:
    // no leds (black)...
    // Set color data:      r      g      b
    rgb_led.setLedColorData(0, 0x00, 0x00, 0x00);
    break;
}

// update LED
rgb_led.show();

iMode = (iMode+1) %5;
}

// *****
// print touch sensor value..
void doTouch() {
    int iTouch;

    iTouch = touchRead(PIN_TOUCH7);

    Serial.printf("Touch sensor = %d\n", iTouch);
}

// *****
// print Hall-effect sensor value..
void doHall() {
    int iHall;

```



```

    iHall = hallRead();

    Serial.printf("Hall sensor = %d\n", iHall);
}

// *****
// print DHT-20 values..
void doDHT20() {
    sensors_event_t humidity, temp;

    // request DH20 sensor values
    aht.getEvent(&humidity, &temp);

    // print temp/humidity
    Serial.printf("Temperature = %.2f deg C, Humidity = %.1f %%rH\n", temp.temperature,
humidity.relative_humidity);
}

// *****
// get and print accelerometer values
void doADXL345() {
    sensors_event_t accel;

    // request ADXL345 Sensor values
    adxl.getEvent(&accel);

    // display results
    Serial.printf("X = %.2f, Y = %.2f, Z = %.2f m/s^2\n", accel.acceleration.x,
accel.acceleration.y, accel.acceleration.z);
}

// *****
// display test
void doTM1637() {
    static int i;

    tm.showNumberDec(i, true, 4, 0);
    ++i;
}

// *****
void loop() {

    Serial.println("\nSENSOR VALUES");
    Serial.println("=====");

    // deal with individual components..

```

```
doBtns();  
doPot();  
doBlueLED();  
doRGBLED();  
doTouch();  
doHall();  
doDHT20();  
doADXL345();  
doTM1637();  
  
delay(500);  
}
```