MANE 3351

LECTURE 5

Classroom Management

Agenda

- Arduinos
- Reminder: Lab One - due 9/17 before 9:30 AM
- There will be a short laboratory session following the lecture

RESOURCES

Handouts

- Lecture 5 Slides
- Lecture 5 Marked Slides

Assignments

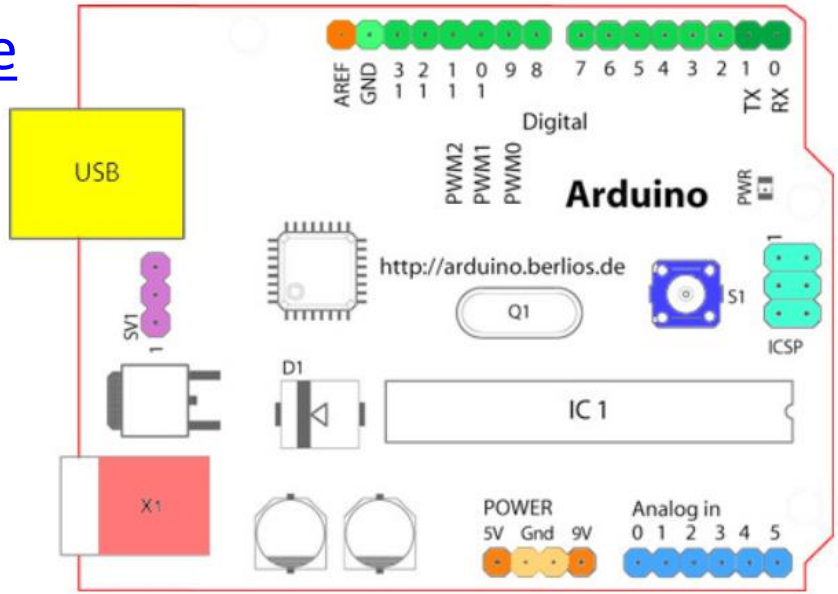
- Lab 2 (assigned 9/17/2025, due 9/24/2024 (before 9:30 AM))

Arduino Uno

Source: [Arduino Uno Website]

Arduino Uno

Source: [Arduino Website]



Arduino I/O

Input/Output

- **Voltages** - 5V pin and 9V pin are present on the board as well as ground pins (0v)
- **Digital Pins** - multiple digital pins are provided on an Arduino board that can be used for general purpose input and output via the **pinMode()**, **digitalRead()**, and **digitalWrite()** commands
- **Analog Pins**- the analog input pins suport 10-bit analog-to-digital conversion using the **analogRead()** function
- **Other Pins** - A reference voltage and reset functionality are also available

Arduino Mega

There are multiple versions of Arduino boards.

The Arduino Mega is often combined with a RAMPS shield to construct homemade Reprap 3D-printers

Source: all3dp

The RAMPS 1.4 board and the Arduino Mega. The blue USB cord goes to the PC. Source: dintid / Instructables

Arduino Mega and Ramp

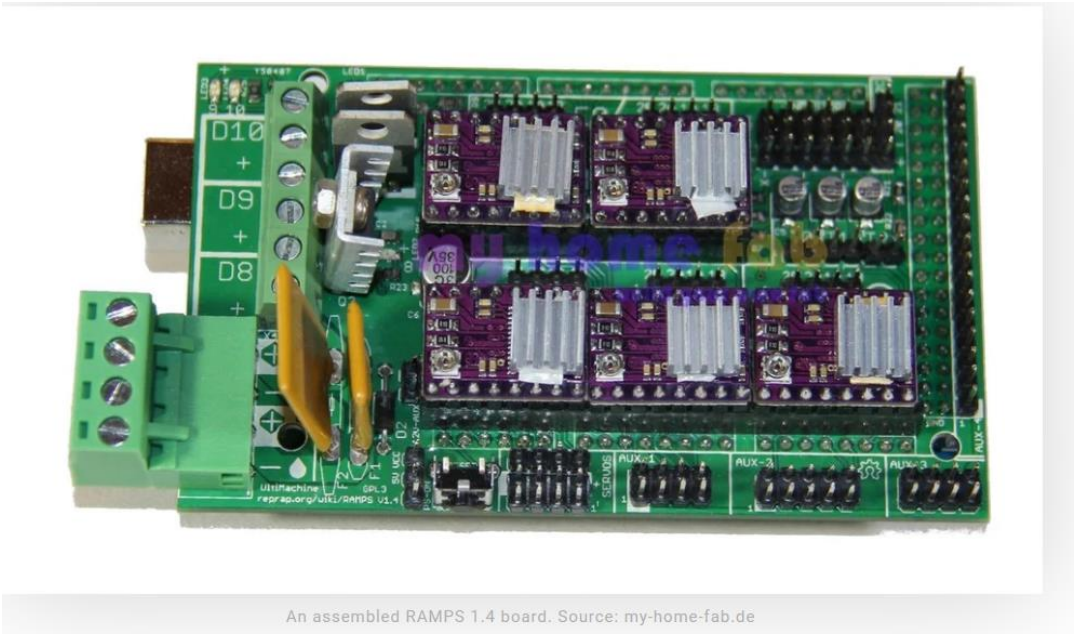
RAMPS Shield

Circuit board that plugs into Arduino Mega for build 3D printers

Typically includes drivers for stepper motors shown below

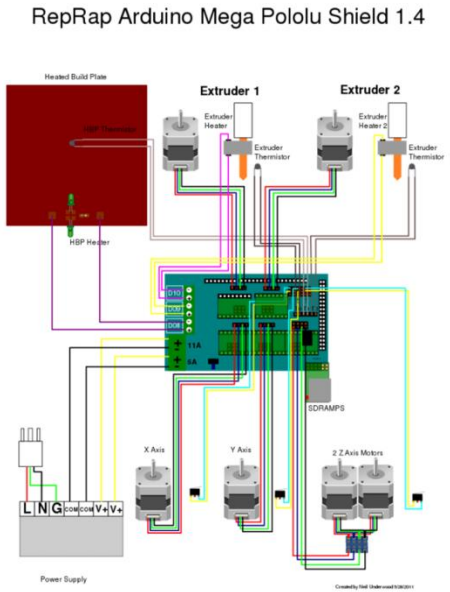
Schematic for 3D printer is also provided below

Both images taken from All3DP website



An assembled RAMPS 1.4 board. Source: my-home-fab.de

RAMPS



3D printer schematic

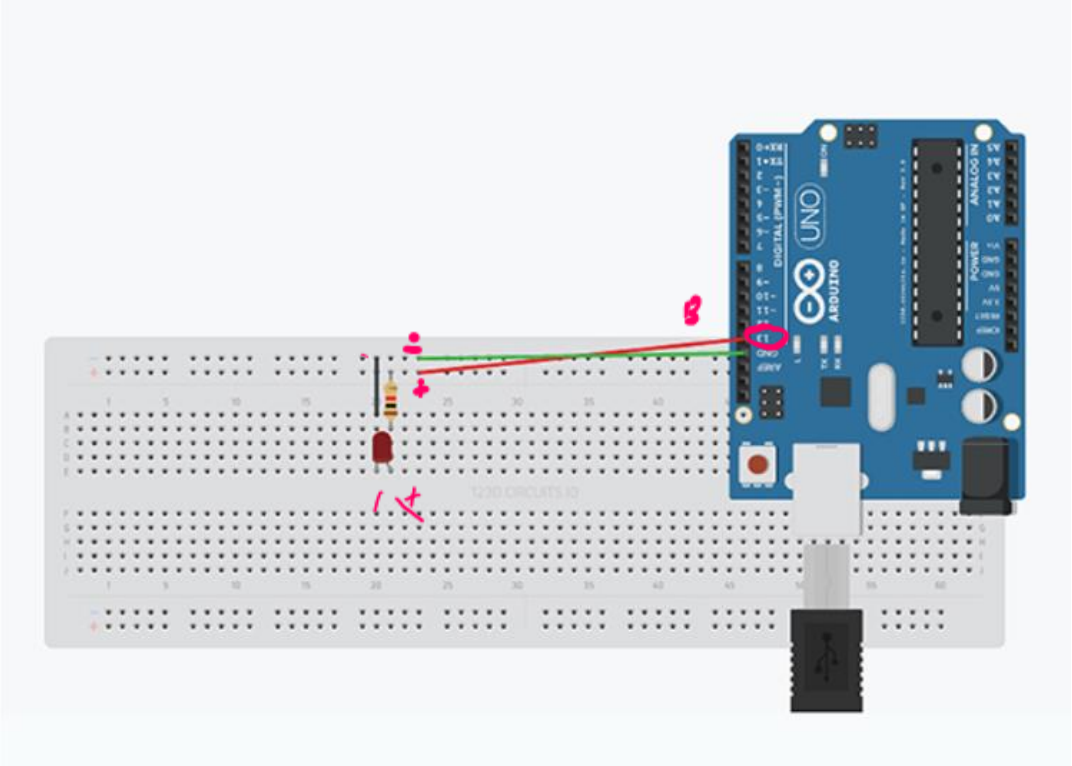
Programming Arduino

- Written in the Arduino Integrated Development Environment
- Arduino programming language is based on a very simple hardware programming language called processing, which is similar to the C language
- Programs for Arduinos are called **sketches**
- Sketches must be uploaded to Arduino (via USB cable)
- Arduino IDE installed on Raspberry Pi

Source: Programming Arduinos

Example 1: Blinking an LED

First example is to an LED (same a
Raspberry Pi)



Arduino LED Circuit

Example 1: Sketch

- Note: there is an error in the sketch. Change digitalRead to digitalWrite

```
int LED =13; // The digital pin to which the LED is connected

void setup ( )
{

pinMode (LED, OUTPUT); //Declaring pin 13 as output pin

}

void loop( ) // The loop function runs again and again
{

digitalWrite (LED, HIGH); //Turn ON the LED
delay(1000); //Wait for 1sec
digitalRead (LED, LOW); // Turn off the LED
delay(1000); // Wait for 1sec

}
```

Arduino LED Code

Code for Blinking an LED

```
int LED = 13; // the digital pin to which the LED is connected

void setup ()
{
pinMode (LED,OUTPUT); //Declaring pin 13 as output ping
}

void loop () //The loop function runs again and again
{
digitalWrite (LED,HIGH); //Turn ON the LED
delay (1000); //wait for 1 second
digitalWrite(LED,LOw); //Turn OFF the LED
delay(1000); //wait for one second
}
```

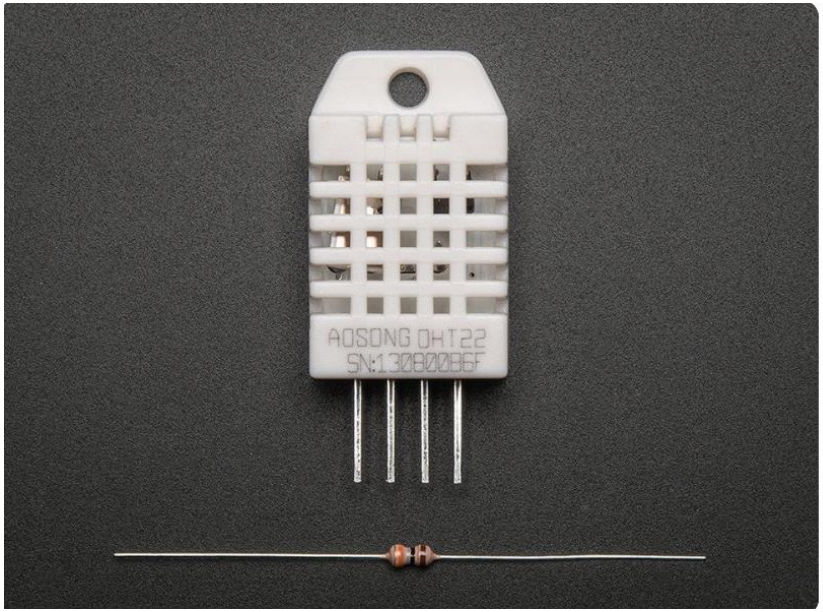
Warning

- Arduinos store the last program in EEPROM and will start running that program as soon as the power is provided
- Cautious Approach:
  - Do not connect circuit board to Arduino until after the Sketch is loaded
  - Afterwards, connect hardware

Arduino Demonstration 1

DHT22 Sensor

Source:



DHT22 temperature-humidity sensor + extras

Product ID: 385

$9.95

Discontinued

Why not check out the DHT20 - AHT20 Pin Module - I2C Temperature and Humidity Sensor?

Description

Technical Details

dht2sensor

Technical Details of DHT11 Sensor

Source:

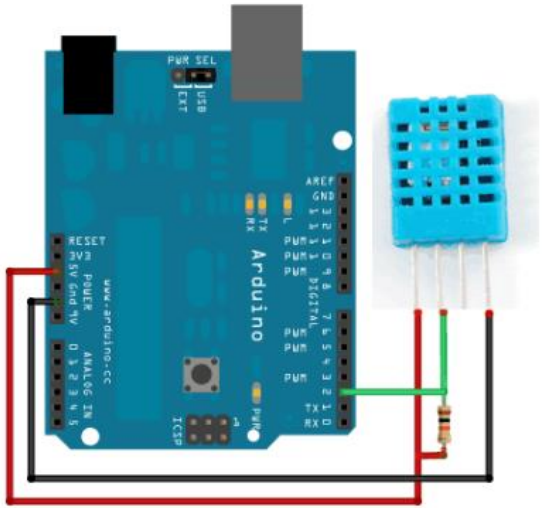
Technical Details

- Low cost
- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 0-100% humidity readings with 2-5% accuracy
- Good for -40 to 80°C temperature readings ±0.5°C accuracy
- No more than 0.5 Hz sampling rate (once every 2 seconds)
- Body size 27mm x 59mm x 13.5mm (1.05" x 2.32" x 0.53")
- 4 pins, 0.1" spacing
- Weight (just the DHT22): 2.4g

dht22 details

Arduino Circuit

Source:



arduino

Arduino Sketch

Source: Example Code

DHTtester

```
// Example testing sketch for various DHT humidity/temperature sensors
// Written by ladyada, public domain

// REQUIRES the following Arduino libraries:
// - DHT Sensor Library: https://github.com/adafruit/DHT-sensor-library
// - Adafruit Unified Sensor Lib: https://github.com/adafruit/Adafruit_Sensor

#include "DHT.h"

#define DHTPIN 2     // Digital pin connected to the DHT sensor
// Feather HUZZAH ESP8266 note: use pins 3, 4, 5, 12, 13 or 14 --
// Pin 15 can work but DHT must be disconnected during program upload.

// Uncomment whatever type you're using!
//#define DHTTYPE DHT11   // DHT 11
#define DHTTYPE DHT22   // DHT 22  (AM2302), AM2321
//#define DHTTYPE DHT21   // DHT 21 (AM2301)

// Connect pin 1 (on the left) of the sensor to +5V
// NOTE: If using a board with 3.3V logic like an Arduino Due connect pin 1
// to 3.3V instead of 5V!
// Connect pin 2 of the sensor to whatever your DHTPIN is
// Connect pin 3 (on the right) of the sensor to GROUND (if your sensor has 3 pins)
// Connect pin 4 (on the right) of the sensor to GROUND and leave the pin 3 EMPTY (if your sensor has 4 pins)
// Connect a 10K resistor from pin 2 (data) to pin 1 (power) of the sensor

// Initialize DHT sensor.
// Note that older versions of this library took an optional third parameter to
// tweak the timings for faster processors.  This parameter is no longer needed
// as the current DHT reading algorithm adjusts itself to work on faster procs.
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));

  dht.begin();
}
```

part 1

Arduino Sketch, continued

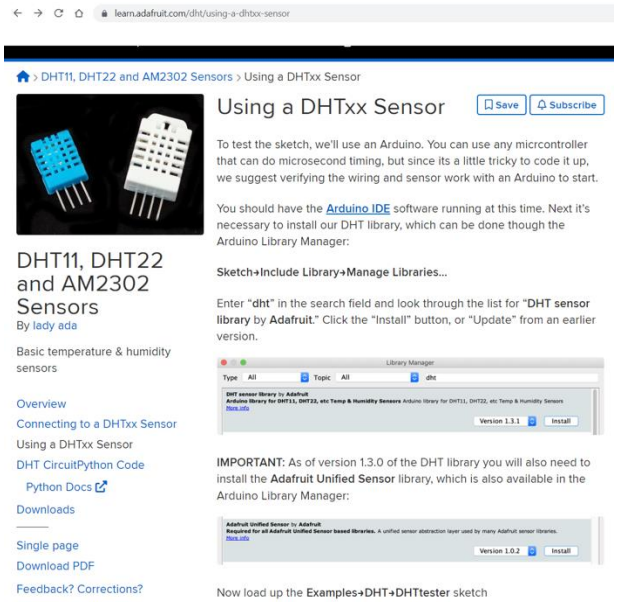
Source: Example Code

```
void loop() {
  // Wait a few seconds between measurements.
  delay(2000);

  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
  }

  // Compute heat index in Fahrenheit (the default)
  float hif = dht.computeHeatIndex(f, h);
  // Compute heat index in Celsius (isFahreheit = false)
  float hic = dht.computeHeatIndex(t, h, false);

  Serial.print(F("Humidity: "));
  Serial.print(h);
  Serial.print(F("%  Temperature: "));
  Serial.print(t);
  Serial.print(F("°C "));
  Serial.print(f);
  Serial.print(F("°F  Heat index: "));
  Serial.print(hic);
  Serial.print(F("°C "));
  Serial.print(hif);
  Serial.println(F("°F"));
}
```

part 2

DHT Library

Source:



dht Sensor Library

DHTest Code

```
// Example testing sketch for various DHT humidity/temperature sensors
// Written by ladyada, public domain

// REQUIRES the following Arduino libraries:
// - DHT Sensor Library: https://github.com/adafruit/DHT-sensor-library
// - Adafruit Unified Sensor Lib: https://github.com/adafruit/Adafruit_Sensor

#include "DHT.h"

#define DHTPIN 2     // Digital pin connected to the DHT sensor
// Feather HUZZAH ESP8266 note: use pins 3, 4, 5, 12, 13 or 14 --
// Pin 15 can work but DHT must be disconnected during program upload.

// Uncomment whatever type you're using!
//#define DHTTYPE DHT11   // DHT 11
#define DHTTYPE DHT22   // DHT 22  (AM2302), AM2321
//#define DHTTYPE DHT21   // DHT 21 (AM2301)

// Connect pin 1 (on the left) of the sensor to +5V
// NOTE: If using a board with 3.3V logic like an Arduino Due connect pin 1
// to 3.3V instead of 5V!
// Connect pin 2 of the sensor to whatever your DHTPIN is
// Connect pin 3 (on the right) of the sensor to GROUND (if your sensor has 3 pins)
// Connect pin 4 (on the right) of the sensor to GROUND and leave the pin 3 EMPTY (if your sensor has 4 pins)
// Connect a 10K resistor from pin 2 (data) to pin 1 (power) of the sensor

// Initialize DHT sensor.
// Note that older versions of this library took an optional third parameter to
// tweak the timings for faster processors.  This parameter is no longer needed
// as the current DHT reading algorithm adjusts itself to work on faster procs.
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));

  dht.begin();
}

void loop() {
  // Wait a few seconds between measurements.
  delay(2000);

  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
  }

  // Compute heat index in Fahrenheit (the default)
  float hif = dht.computeHeatIndex(f, h);
  // Compute heat index in Celsius (isFahreheit = false)
  float hic = dht.computeHeatIndex(t, h, false);

  Serial.print(F("Humidity: "));
  Serial.print(h);
  Serial.print(F("%  Temperature: "));
  Serial.print(t);
  Serial.print(F("°C "));
  Serial.print(f);
  Serial.print(F("°F  Heat index: "));
  Serial.print(hic);
  Serial.print(F("°C "));
  Serial.print(hif);
  Serial.println(F("°F"));
}
```

Demonstration 2 - Temperature/Humidity Sensor