IoT devices
Structure and programming

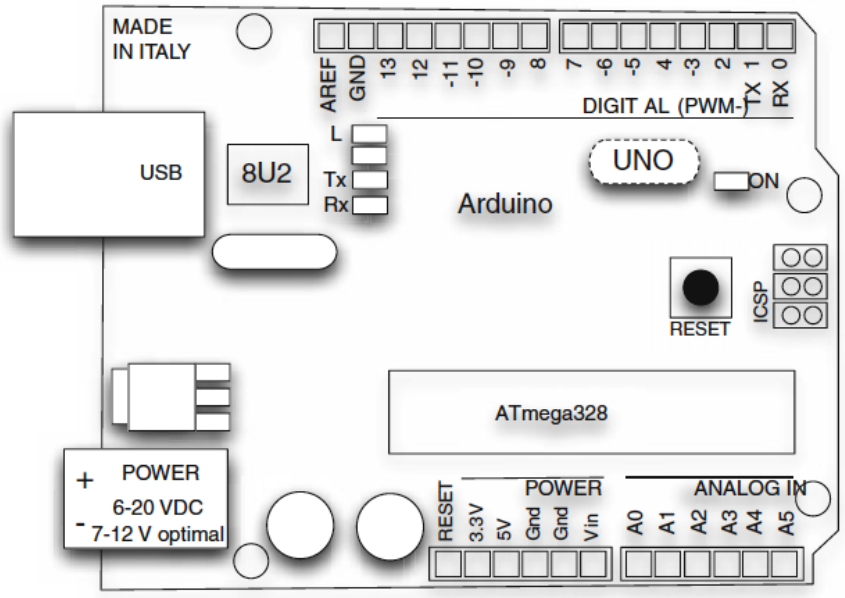
IoT devices

- **An IoT device is an embedded system characterised by**
  - Hardware: microcontroller, actuators, sensors, ...
  - Software: for control and computational intelligence
  - Internet connectivity: network controllers enabling the communication
- **The hardware and software is co-designed**
  - Requires knowledge of both the hardware and software capabilities
  - Devices are often real-time systems and are performing safety critical tasks
  - Often designed and developed for a cost-competitive market
- **Interaction with sensors and actuators**
  - Low-level interfacing: digital and analog input/output via microcontroller pins
  - Higher-level interfacing and protocols: UART, SPI, I2C,…
  - Software libraries used to hide low-level microcontroller and protocol details

Brief recap
Arduino programming

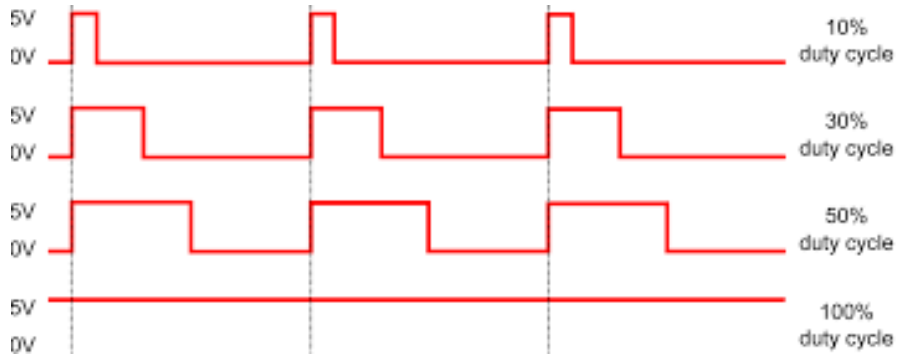
Arduino – prototyping device

- **Pins 0-13 are digital input/ouput**
  - Read/write digital output
    5v = HIGH (1) | 0v = LOW (0)



- **Pins A1-A5 are analog input**
  - A-D converted to 0 .. 1023 (10 bit)

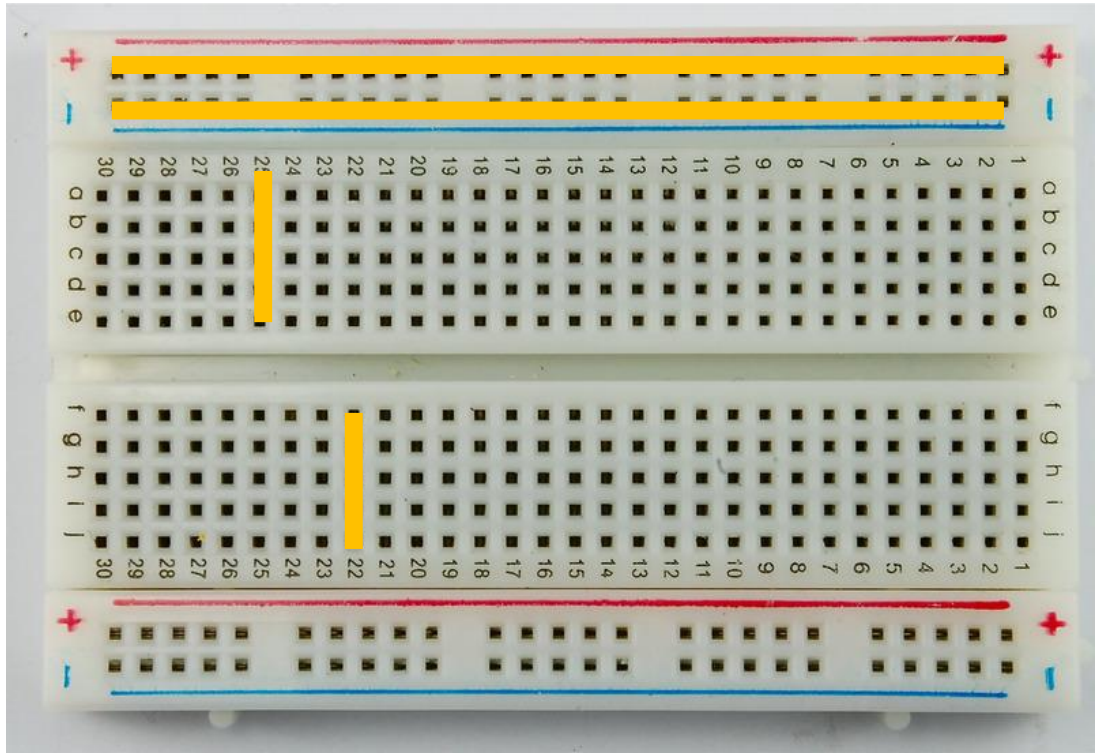
- **No analog output pins**
  - Pulse-Width Modulation (PWM) on digital output pins 3,5,6,10,11 (~) with 8-bits (0..255)
  - Duty cycle: percentage of time where the signal is high used to control perceived voltage



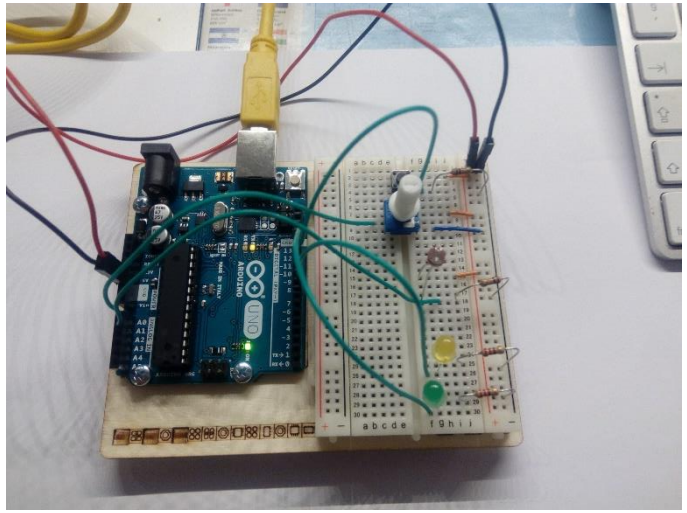
https://www.arduino.cc/en/Reference/Board

Breadboard

- **Connecting terminals of electrical components using non-permanent wiring (unlike soldering)**



- **Connect sensors, actuator, and components to the microcontroller**



Programming [https://www.arduino.cc/reference/en]

- **Reactive programming model**
  - `void setup()` function executed once on power-up and used for initialisation of input/output pins, libraries,…
  - `void loop()` function executed repeatedly (infinite loop) - read input (sense), process and update, actuate (control).
- **Programming sensor and actuator interaction**
  - Setting up modes for pins: `pinMode(pin, INPUT/OUTPUT);`
  - Digital input and output: `digitalRead(pin); digitalWrite(pin,v);`
  - Analog input (10 bit): `analogRead(pin);`
  - PWM analog output: `analogWrite(pin,v)`
- **Serial communication via USB programming interface**
  - Initialisation: `Serial.begin(baudrate)`
  - Reading: `Serial.available() [...] Serial.read();`
  - Writing: `Serial.print(..);`

Serial communication

- **Enables communication between the target platform (device) and other devices / components / controllers.**

- **Universal Asynchronous Receiver-Transmitter (UART)**
  - Designed for long-distance asynchronous communication (no shared clock)
  - Sequential bidirectional transfer of bytes with flow control
  - Low hardware overhead: 1 pin on each device / 1 bit transmitted at a time
  - Most microcontroller have built-in support for UART communication

- **Can also be used over USB for input/output and for debugging purposes in conjunction with a serial monitor.**

Example: debugging

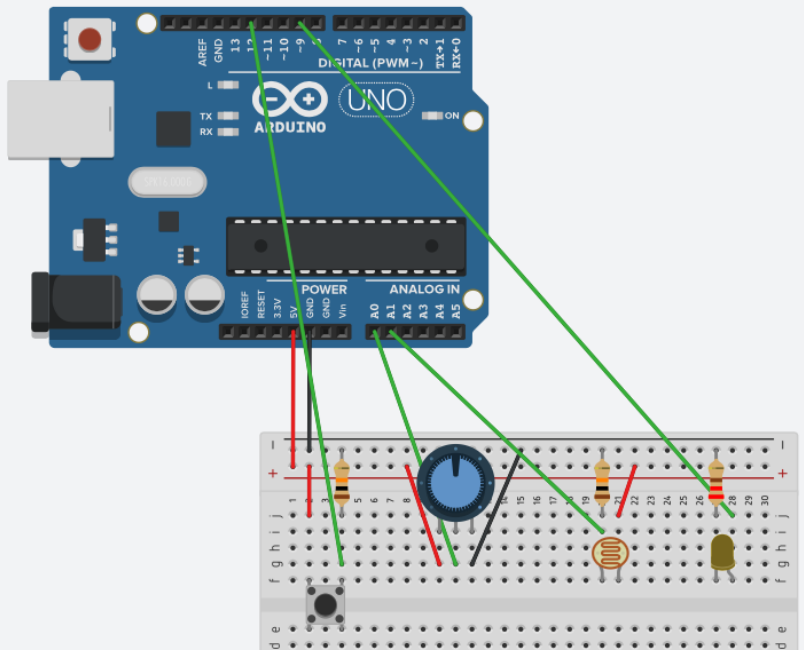
- **Configuring and controlling debugging output using serial communication**

```
void setup()
{
  Serial.begin(9600);
  pinMode(12, INPUT);
  pinMode(13, OUTPUT);
  pinMode(9, OUTPUT);
}

byte debug = 0;
```

```
if (Serial.available() > 0) {

   byte input = Serial.read();
   debug = (input == 49);

}

if (debug) {
  Serial.print(photoval);
  Serial.print("<");
  Serial.print(potval);
  Serial.print(":");
  Serial.print(ledbrightness);
  Serial.println();
}
```

TinkerCad [https://www.tinkercad.com/]

- **Web application for circuit design and simulation**



Example

- **A push button is used to activate / deactivate the systems the system.**
- **A on-board / green LED indicate if the system is active.**
- **A light sensor is used to sense light intensity.**
- **A potentiometer is used to set a threshold for turning on an LED with a certain brightness.**

- **Circuit design and code**
  - https://www.tinkercad.com/things/ki35AdXHyJz-dat159-lecture-1-example
  - https://github.com/lmkr/dat159/blob/master/iotdevices/arduino/lecture1/InputOutput/InputOutput.ino

Now: mandatory hands-on - A

An access control device

To be designed and implemented in TinkerCAD

Can be done in groups of maximum 3

https://github.com/lmkr/dat159/blob/master/week1.md