Internet of Things
Lab and Assignment

Dr Joshua Ellul
joshua.ellul@um.edu.mt
Department of Computer Science
University of Malta

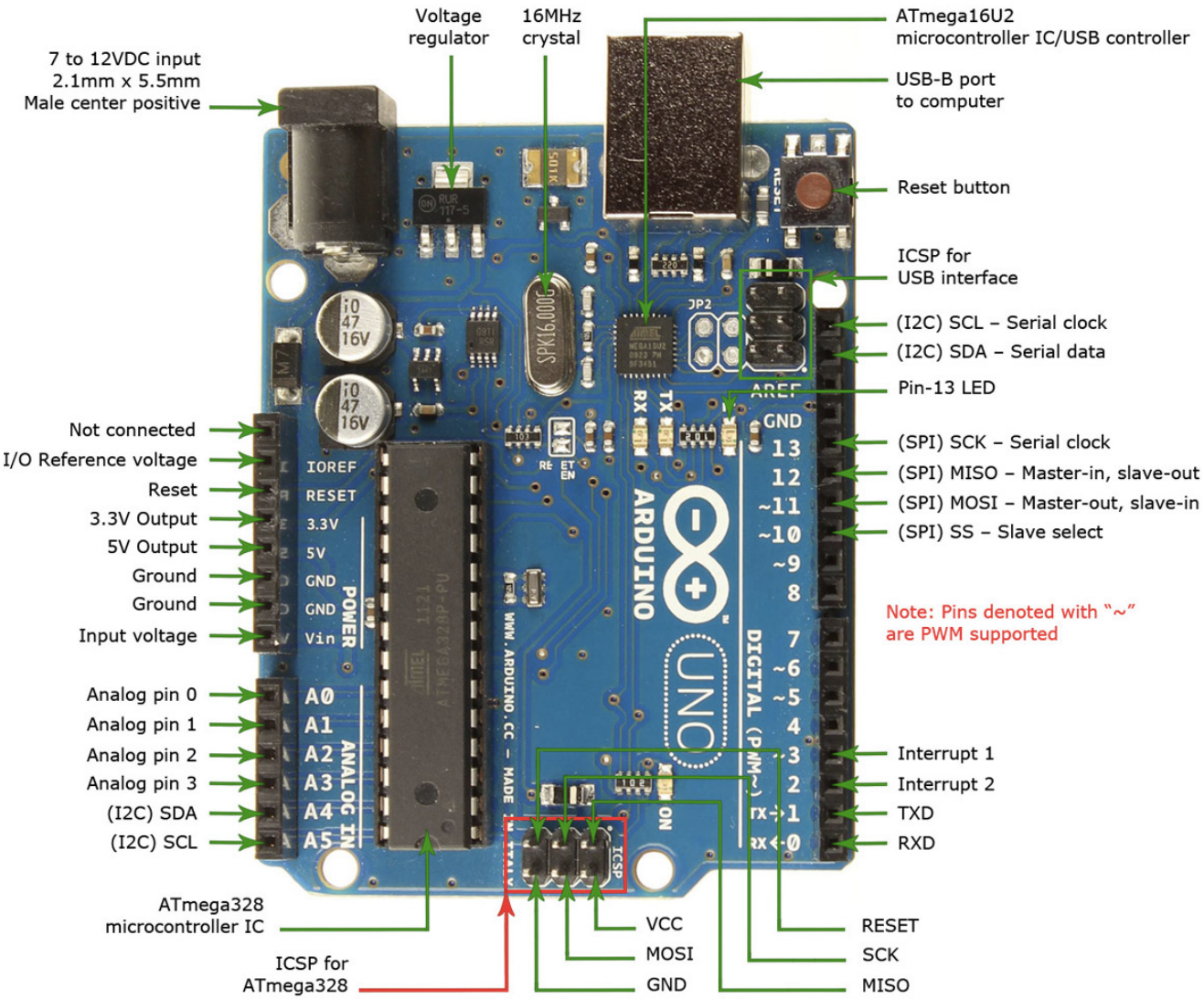
- Download Arduino IDE
  - https://www.arduino.cc/en/Main/Software

Programming an Arduino

Programming an Arduino

```
void setup()
{
      //executed once
}

void loop()
{
    //executed repeatedly
}
```

Arduino Uno

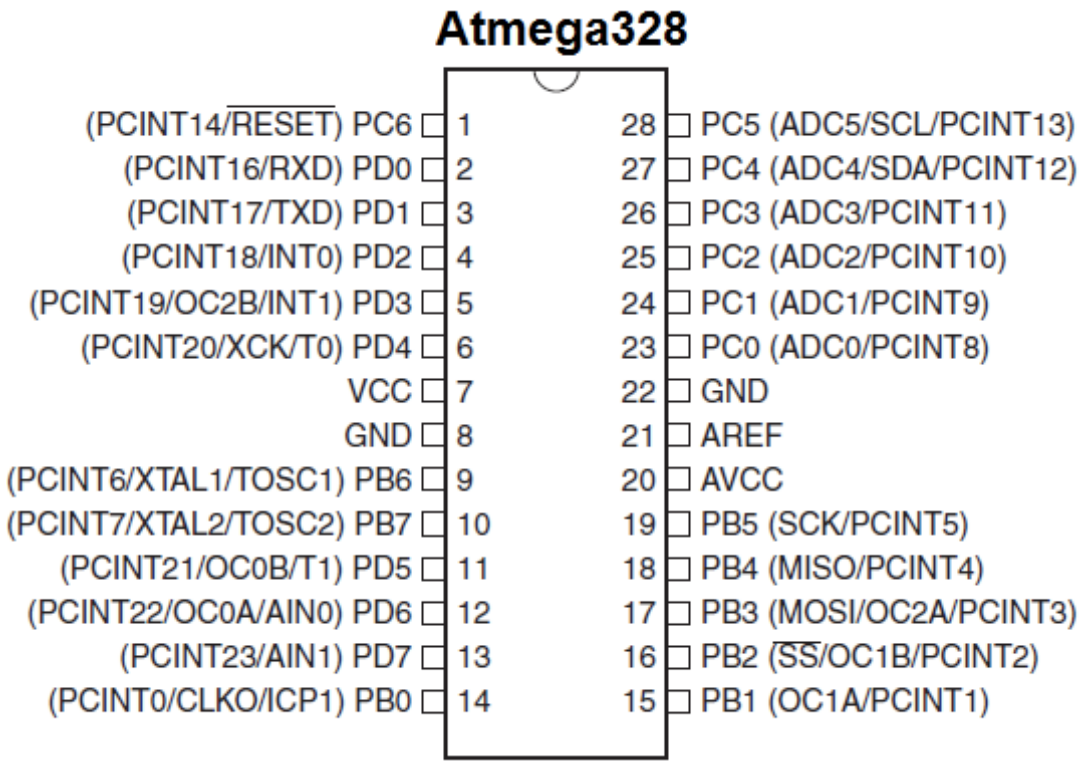


https://www.robomart.com/image/catalog/RM0058/01.jpg

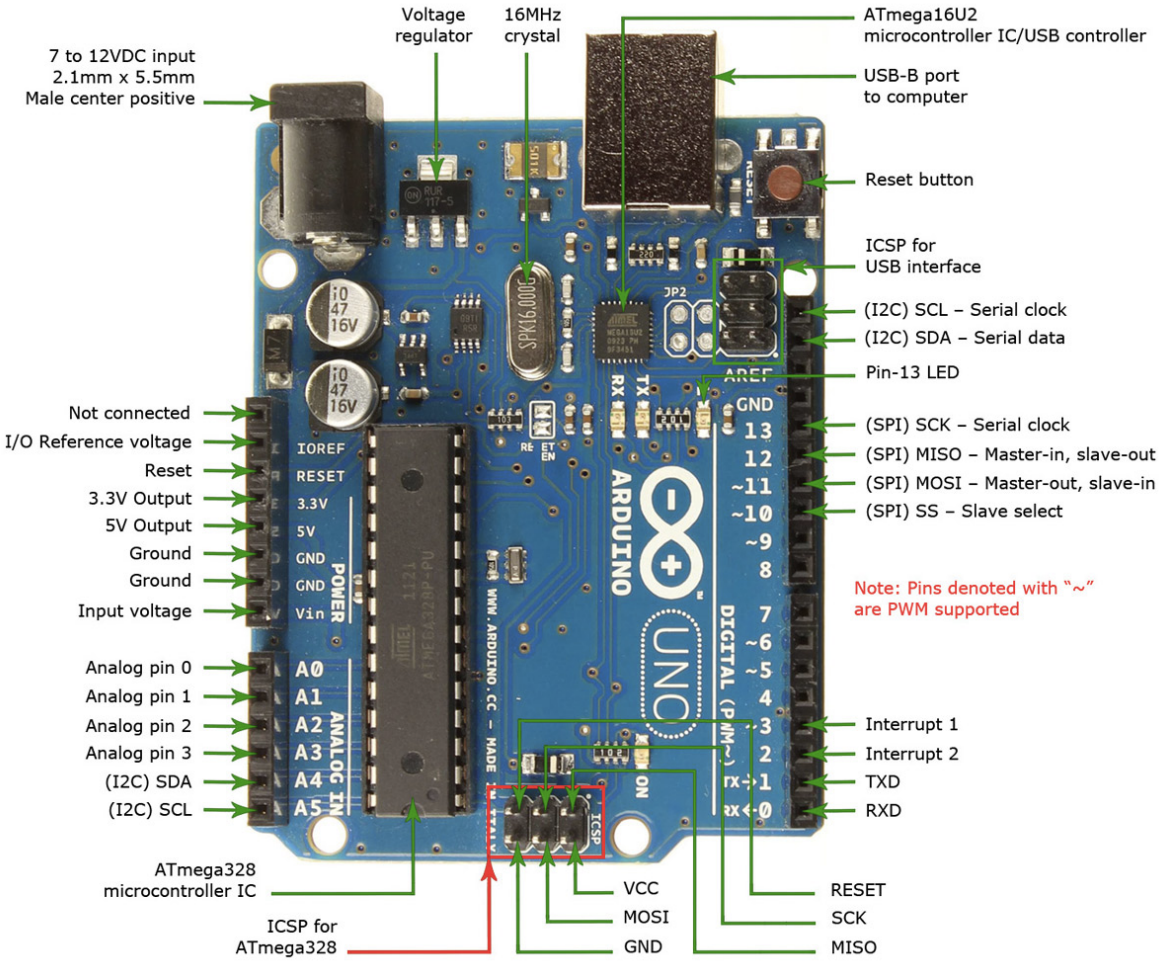
Digital Input or Output Pins

(Typicall) Each pin can be configured to be an input or an output

We need to configure whether a pin is an output



Turning on an LED



A LED is connected to PIN 13

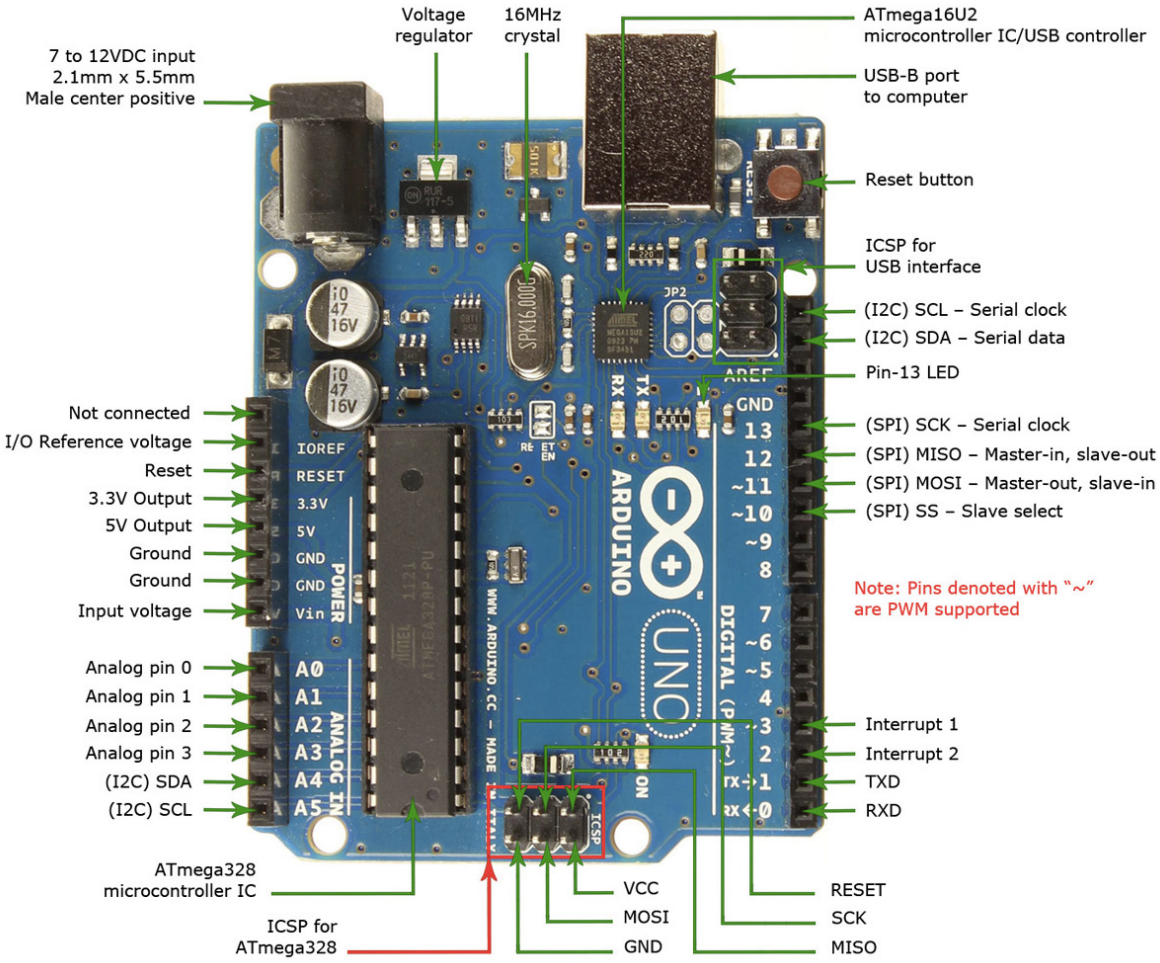
We need to configure it to be an output:

```
pinMode(13, OUTPUT);
```

Programming an Arduino

```
void setup()
{
   //configure pin 13 to be an output
   pinMode(13, OUTPUT);
}

void loop()
{
   //executed repeatedly
}
```

Turning on an LED



We need to choose to turn on or off the LED

Turn on (drive pin low):

```
digitalWrite(13, HIGH);
```

Turn off (drive pin low):

```
digitalWrite(13, LOW);
```

If we want to 'pause':

```
delay(<milliseconds>);
```

Programming an Arduino

```
void setup() {
  pinMode(13, OUTPUT); //configure pin 13 as output
}

void loop() {
  digitalWrite(13, HIGH); //turn led on
  delay(1000); //wait 1 second
  digitalWrite(13, LOW); //turn led off
  delay(1000); //wait 1 second
}
```

Communicating to the outside World… Write Data over Serial (via USB)

- Initialise serial port:
  - Serial.begin(9600);
- Write to serial:
  - Serial.println(<message>);

Communicating to the outside World… Write Data over Serial (via USB)

```
void setup() {
  Serial.begin(9600);
  Serial.println("Serial started!");
}

void loop() {
  Serial.println("Debug info");
  delay(1000);
}
```

Reading Analog Sensor Input

analogRead(<pin>);

https://www.arduino.cc/en/Tutorial/ReadAnalogVoltage

Reading Analog Sensor Input and Communicating to the Outside World

```
void setup() {
  Serial.begin(9600);
}

void loop() {
  int sensorValue = analogRead(A0);
  float voltage = sensorValue * (5.0 / 1023.0);
  Serial.println(voltage);
  delay(30000);
}
```

https://www.arduino.cc/en/Tutorial/ReadAnalogVoltage

Programming Considerations

- Serial.println(”This string is in RAM”);
- Serial.println(F(”This string is in flash”));

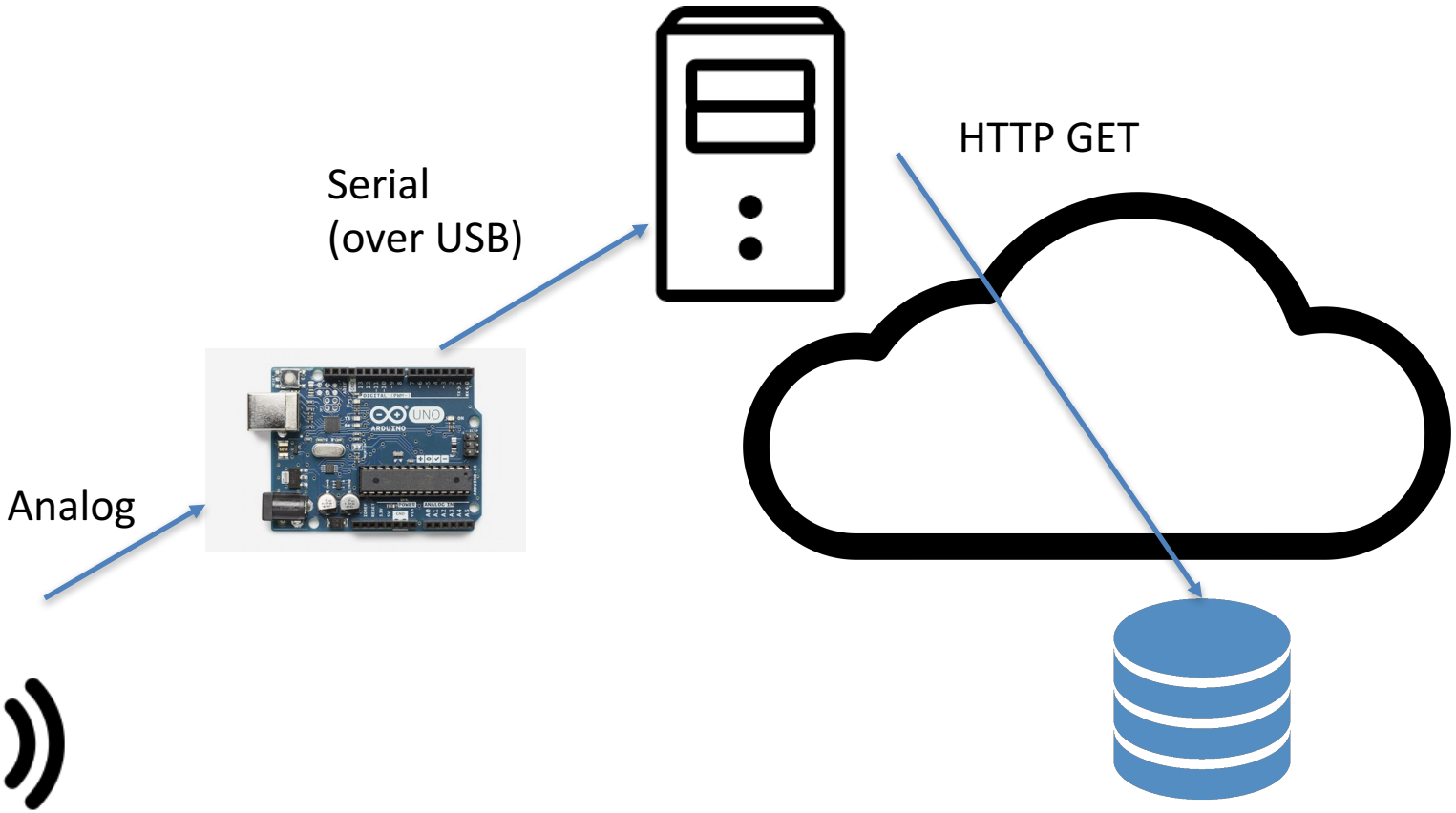
- Download and Install NodeJS
  - https://nodejs.org/en/download/

Programming The IoT Edge

- For resource constrained IoT devices, often a gateway or edge node is required to interact with the outside world

- Raspberry Pi's, or traditional computing platforms often used

Programming The IoT Edge



Example: Edge Node App

- Node.js example app (output data from serial port to console)

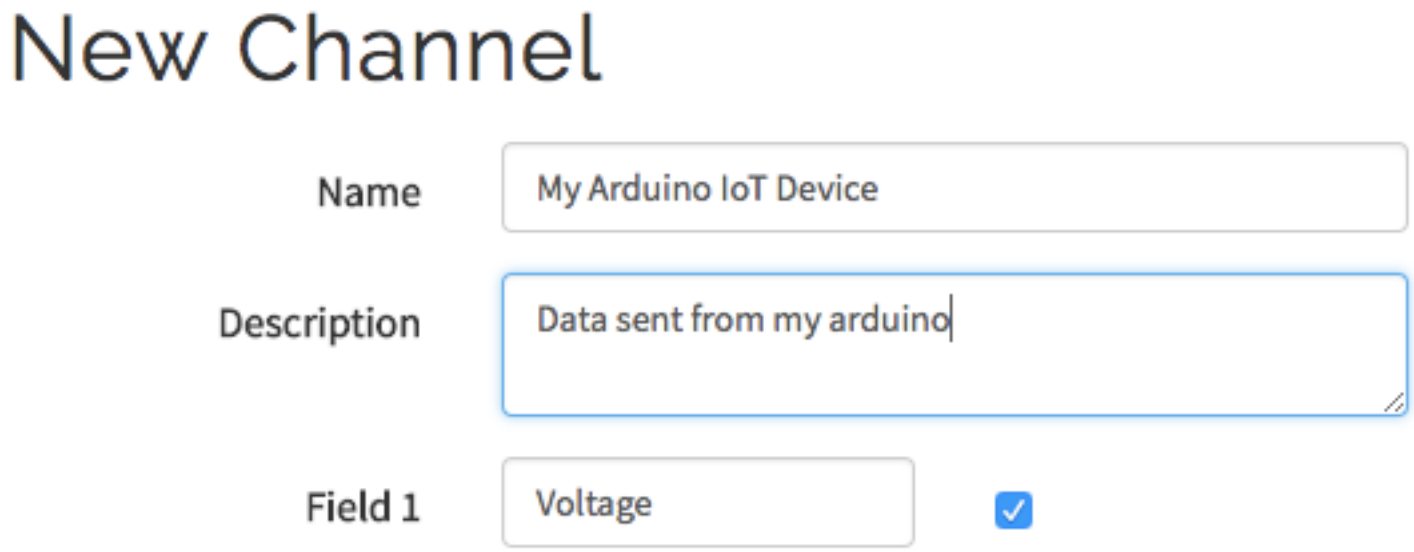
```
var SerialPort = require('serialport');
const Readline = SerialPort.parsers.Readline;
var port = new SerialPort('/dev/cu.usbmodem1421', {
  baudRate: 9600
});
const parser = port.pipe(new Readline());

parser.on('data', function(data) {
    console.log(data);
});
```

Instead, send into the Cloud!

Cloud Based IoT Analytics Platform

- Many, as a use case, we'll use ThingSpeak.com

- ThingSpeak allows you to create different 'channels' for different sensors



Example: Edge Node App

```
parser.on('data', function(data) {
    console.log(data);
    var options = {
      host: 'api.thingspeak.com',
      port: 80,
      path: '/update?api_key=M8QVD63DFEXYOEH&field1=' + encodeURIComponent(data)
    };

    http.get(options, function(resp){
      resp.on('data', function(chunk){
        console.log('sent to thingspeak');
      });
    }).on("error", function(e){
      console.log("error sending to thingspeak: " + e.message);
    });
});
```

Assignment/Lab:

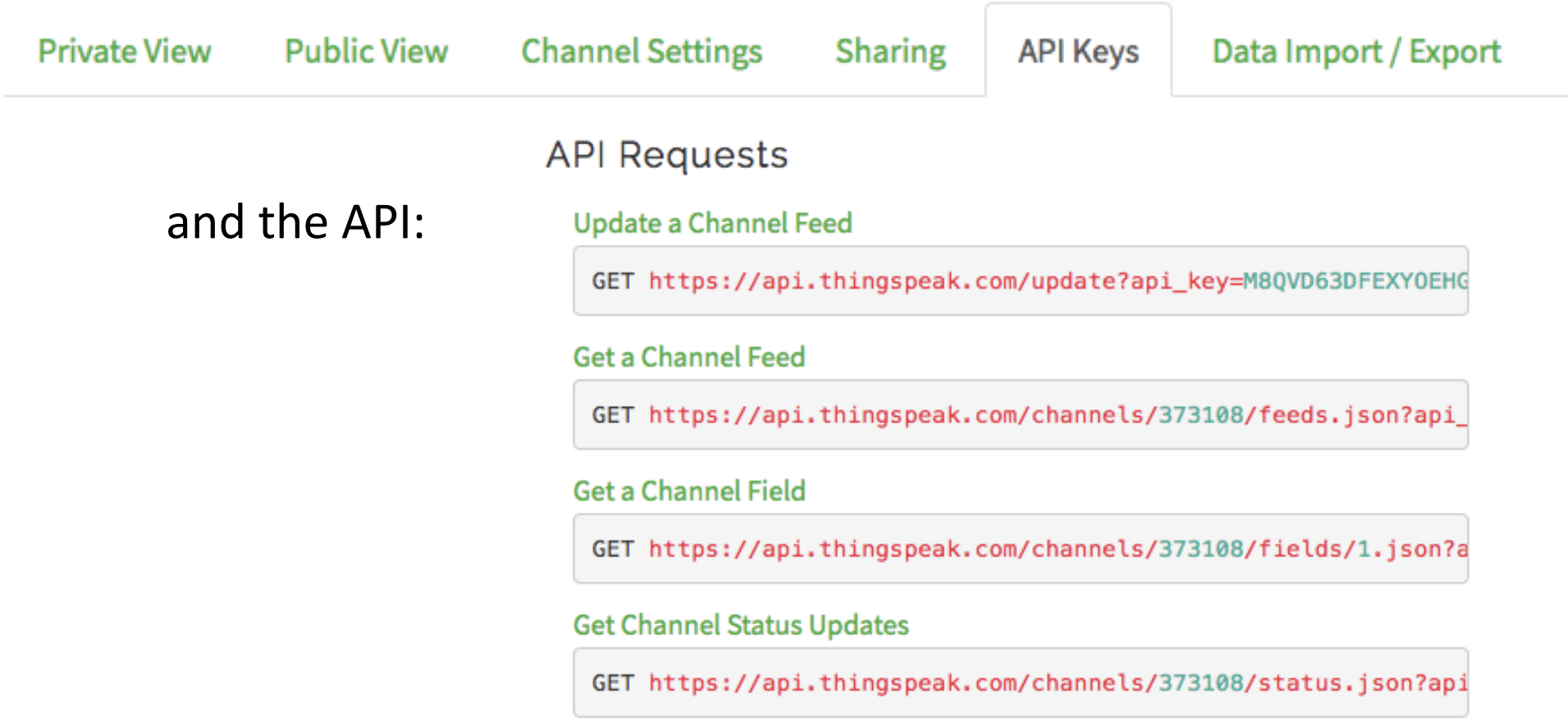
- In class, a demonstration of how a sensor node can be developed to send readings to an edge device was shown.
  - The use case: a data collection system needs to be developed to help scientists understand a particular environment. An analog sensor will be used to detect some phenomenon within the environment (emulated by nodeJS code).
  - In your lab and for your assignment, since you do not have access to a sensor node,
    - 1. Create a nodeJS timer event, that will: after one second, generate a random floating point number between 0 and 5 (to represent an analog input of between 0 and 5).
    - 2. Create a function called *processReadingOnIoTDevice* which will: take as input a floating point number. The random number generated in part 1, should be fed into the function defined in this part.
      The *processReadingOnIoTDevice* function should: send the reading to the edge node, only if the reading has changed by a value differing more than 0.1 since the last reading.
      Provide a function *sendToEdgeNode* that represents sending the value from the IoT device to the edge node. (There's no need to create a separate event to handle sending data to the edge node, a function call is fine).

Assignment/Lab:

- 3. There are (at least) two problems with the approach outlined in 2. One to do with unexpected behaviour in terms of what can be monitored; and another to do with energy consumption. Explain further what the problems might be.
- 4. Alter the *processReadingOnIoTDevice* function to: calculate a simple moving average (SMA) of the last 10 readings. If the last SMA **sent to the edge node** differs to the current SMA by at least 0.5, then send the new SMA to the edge node.

Assignment/Lab (cont)

- 5. Create an account on thingspeak.com
  > Create a ‘channel’ in your thingspeak.com account
  > Familiarise yourself with thingspeak.com and how you can manipulate the channel; particularly by looking at the channel tabs:



and the API:

Assignment/Lab (cont)

- 6. In the *sendToEdgeNode* function, implement functionality to send the received SMA to the channel created in part 4.