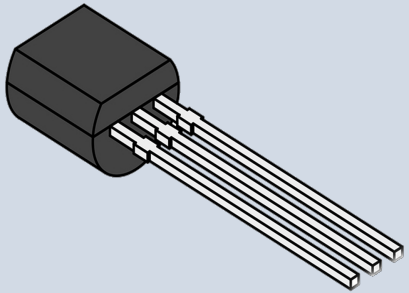
https://www.halvorsen.blog

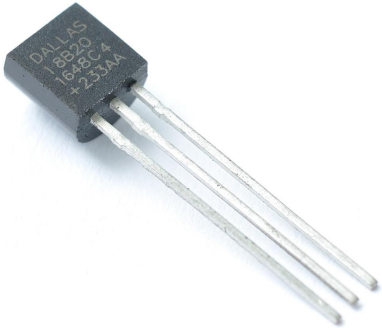


Arduino and DS18B20

1-Wire Temperature Sensor

Hans-Petter Halvorsen

Contents



Arduino

Hans-Petter Halvorsen

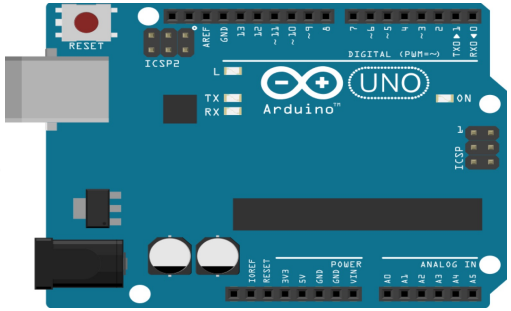
Table of Contents

Arduino

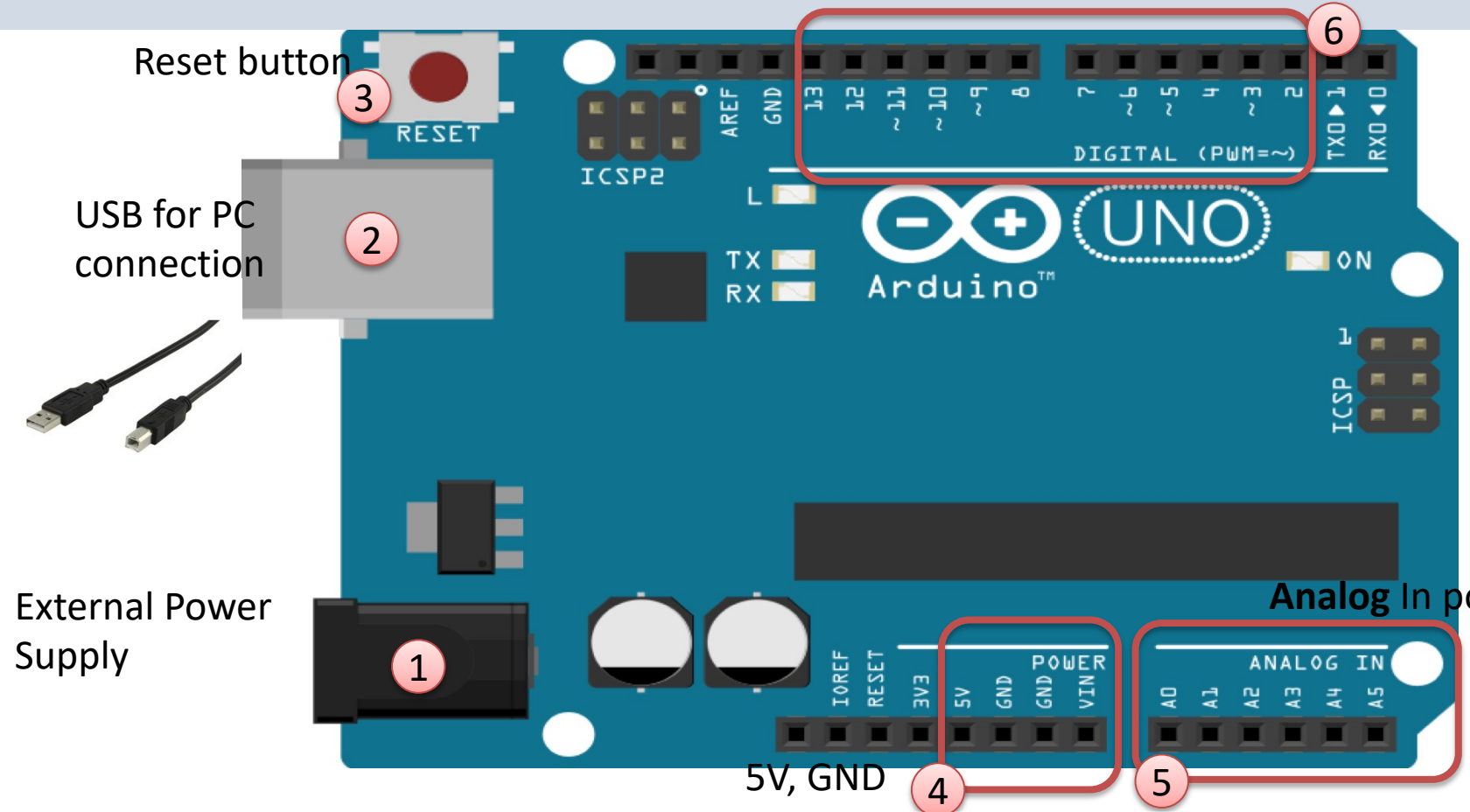
- Arduino is an open-source electronics platform based on easy-to-use hardware and software.
- It's intended for anyone making interactive projects, from kids to grown-ups.
- You can connect different Sensors, like Temperature, etc.
- It is used a lots in Internet of Things projects
- Homepage: https://www.arduino.cc

Arduino

- Arduino is a Microcontroller
- Arduino is an open-source platform with Input/Output Pins (Digital In/Out, Analog In and PWM)
- Price about $20
- Arduino Starter Kit ~$40-80
  with Cables, Wires, Resistors, Sensors, etc.



Arduino UNO

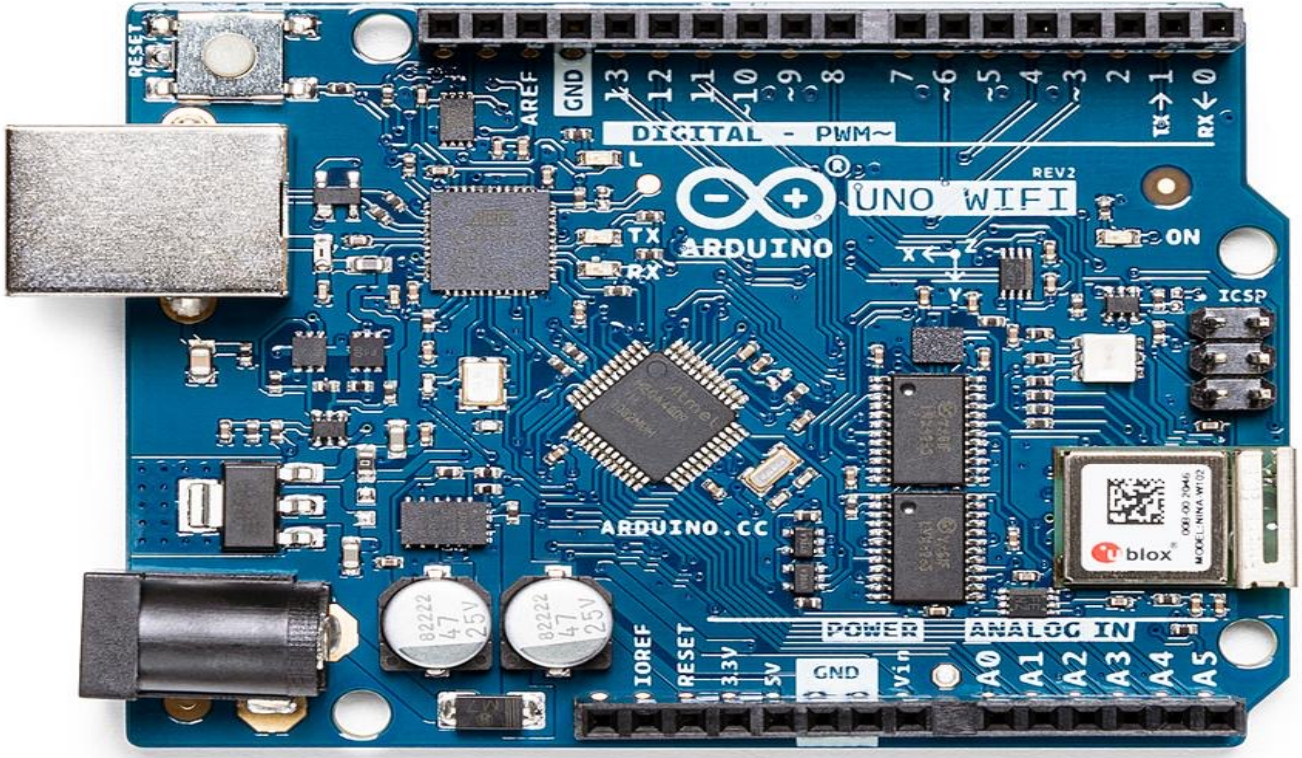


Digital ports (2-13)

Analog In ports (0-5)

Arduino UNO WiFi Rev 2

The Arduino Uno WiFi is functionally the same as the Arduino Uno Rev3, but with the addition of WiFi / Bluetooth and some other enhancements.



Arduino Software

Upload Code to Arduino Board

Save

Open Serial Monitor

Compile and Check if Code is OK

Open existing Code

Creates a New Code Window

```
sketch_nov16b
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

In this window you create your Program

Arduino/Genuino Uno on /dev/cu.usbmodem1A1231

Error Messages can be seen here

The software can be downloaded for free:

www.arduino.cc

https://www.halvorsen.blog

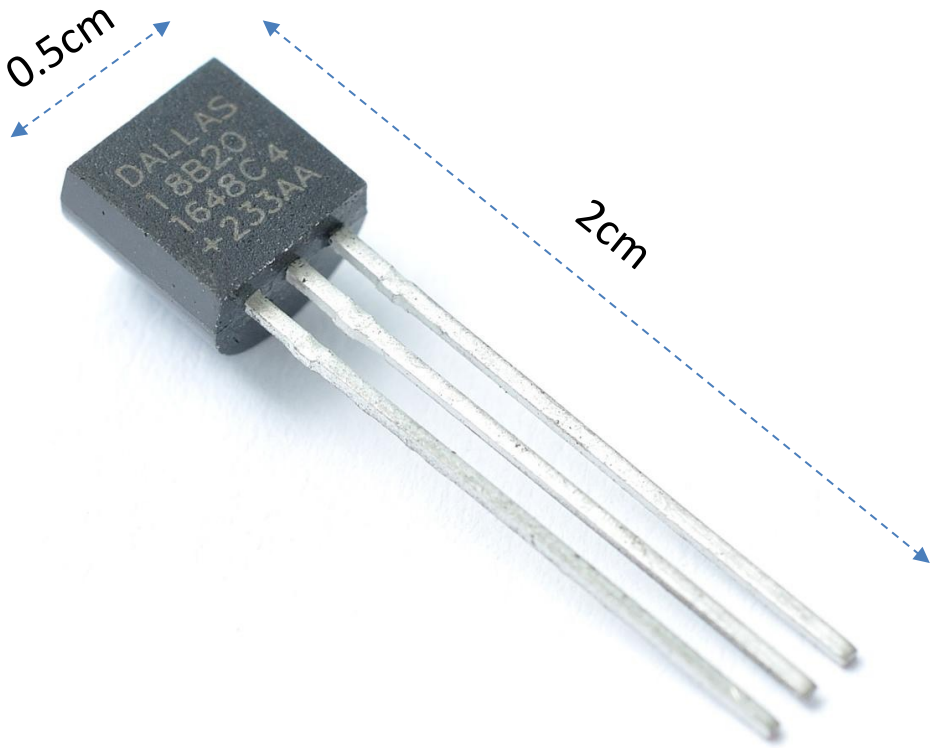
DS18B20

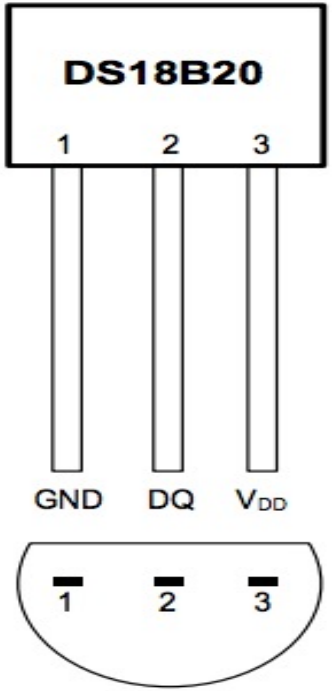
1-Wire Temperature Sensor

Hans-Petter Halvorsen

Table of Contents

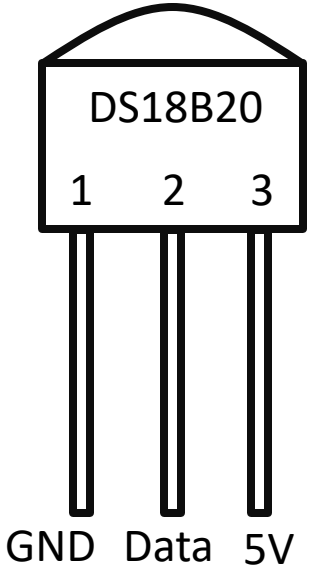
DS18B20 1-Wire Temperature Sensor





TO-92
(DS18B20)

DS18B20 1-Wire Temperature Sensor



- DS18B20 is a 1-Wire Digital Temperature Sensor, this means the Sensor only need 1 Pin for Communication (+ one pin for GND and one pin for 5V)
- **Accuracy** $\mathbf{+/-0.5}$°C
- 9 to 12-bit resolution (Programmable)
- **Temperature range** $\mathbf{-55}$**°C to** $\mathbf{+125}$**°C**
- Price: About $4
- Datasheet: https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf

The Data pin goes to a Digital Input pin on the Arduino

Digital vs. Analog Temperature Sensors

- Digital temperature sensors like the DS18B20 differ from analog thermistors in several important ways.
- In thermistors, changes in temperature cause changes in the resistance of a ceramic or polymer semiconducting material.
- Usually, the thermistor is set up in a voltage divider, and the voltage is measured between the thermistor and a known resistor.
- The voltage measurement is converted to resistance and then converted to a temperature value by the microcontroller.
- Digital temperature sensors are typically silicon based integrated circuits.
- They contain the temperature sensor, an analog to digital converter (ADC), memory to temporarily store the temperature readings, and an interface that allows communication between the sensor and a microcontroller.
- Unlike analog temperature sensors, calculations are performed by the sensor, and the output is an actual temperature value (in degrees Celsius) – so no conversion is needed.
- The DS18B20 communicates with the “One-Wire” communication protocol, a proprietary serial communication protocol that uses only one wire to transmit the temperature readings to the microcontroller.

https://www.circuitbasics.com/raspberry-pi-ds18b20-temperature-sensor-tutorial/

https://www.halvorsen.blog

Arduino Examples

Hans-Petter Halvorsen

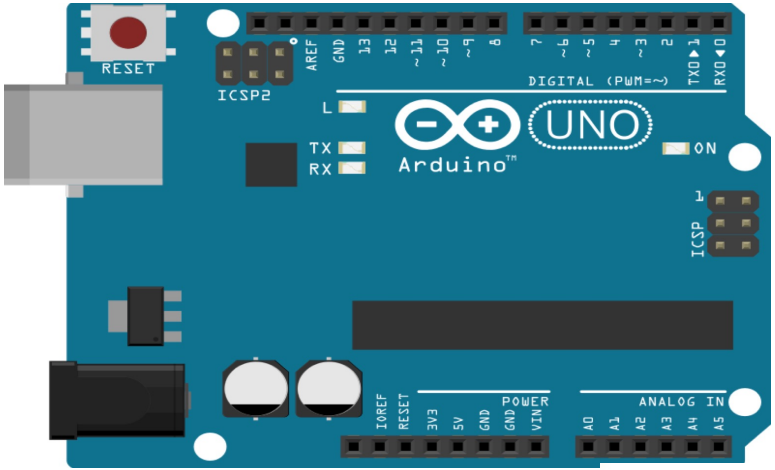
Table of Contents

Read Temperature Data

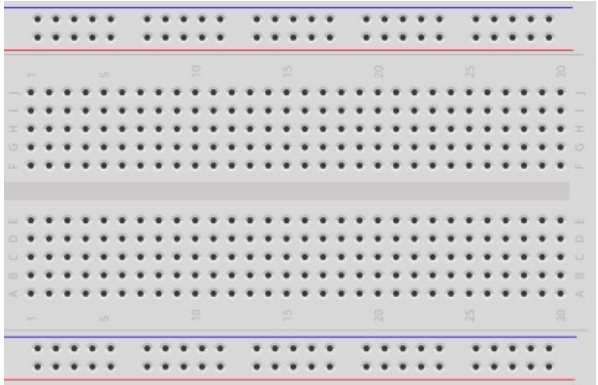
Hans-Petter Halvorsen

Table of Contents

Equipment

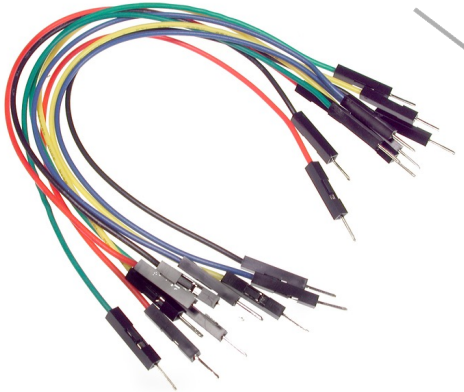


Arduino

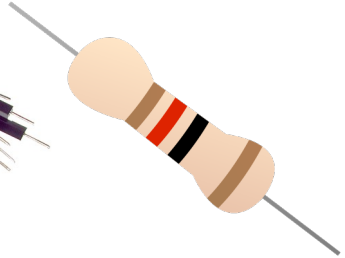


Breadboard

DS18B20



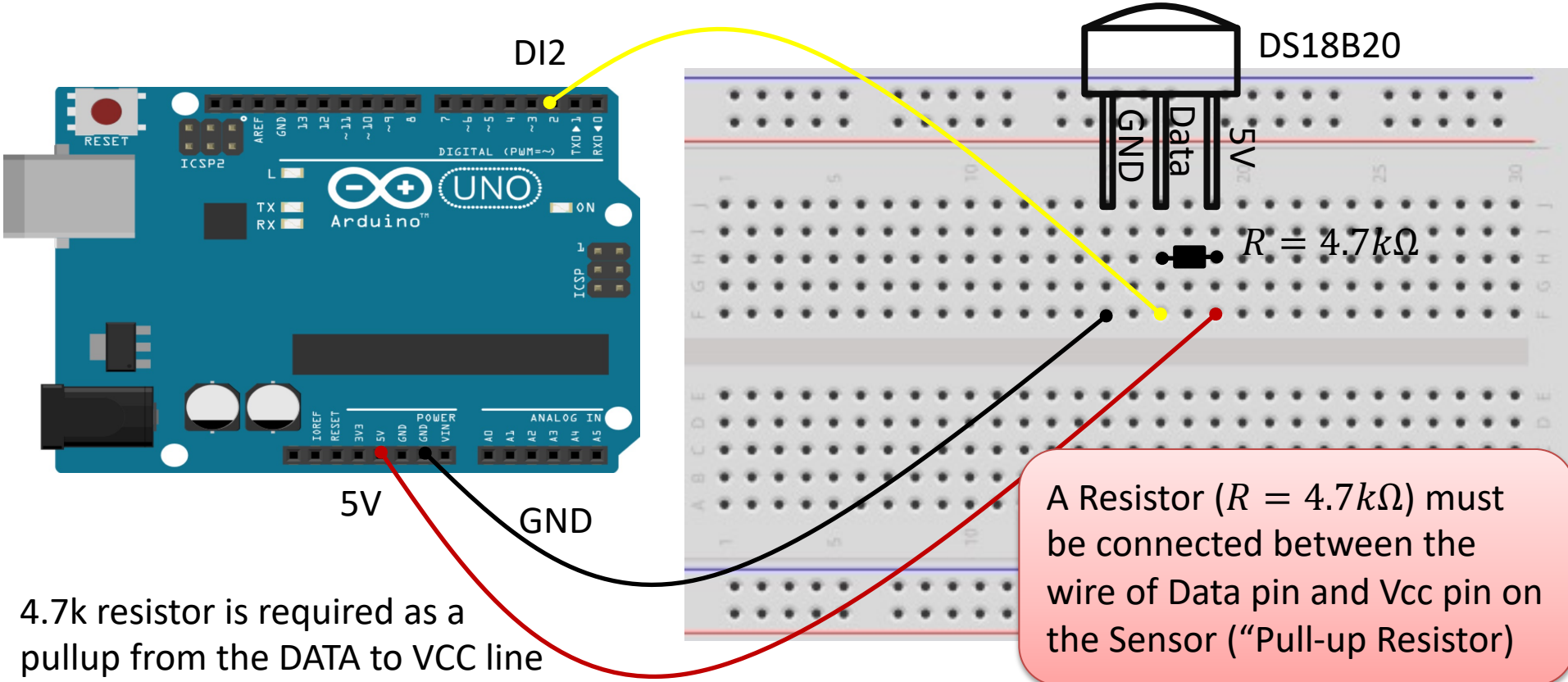
Wires



Resistor $R = 4.7\mathrm{k}\Omega$

Wiring

Be careful to get the DS18B20 the right way around. The curved edge should be to placed as shown in the figure below. If you put it the wrong way around, it will get hot and then break.



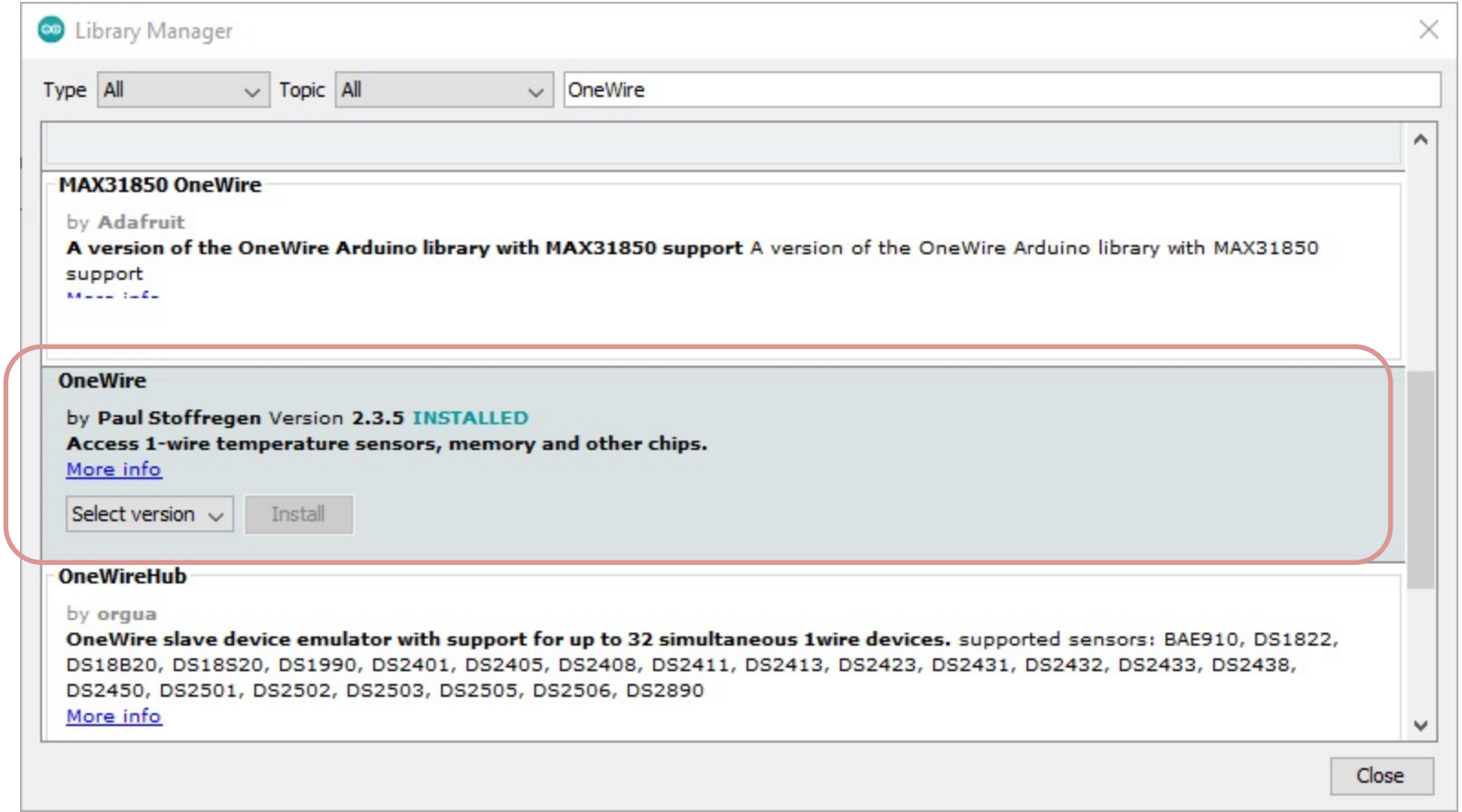
4.7k resistor is required as a pullup from the DATA to VCC line

A Resistor ($R = 4.7k\Omega$) must be connected between the wire of Data pin and Vcc pin on the Sensor ("Pull-up Resistor)

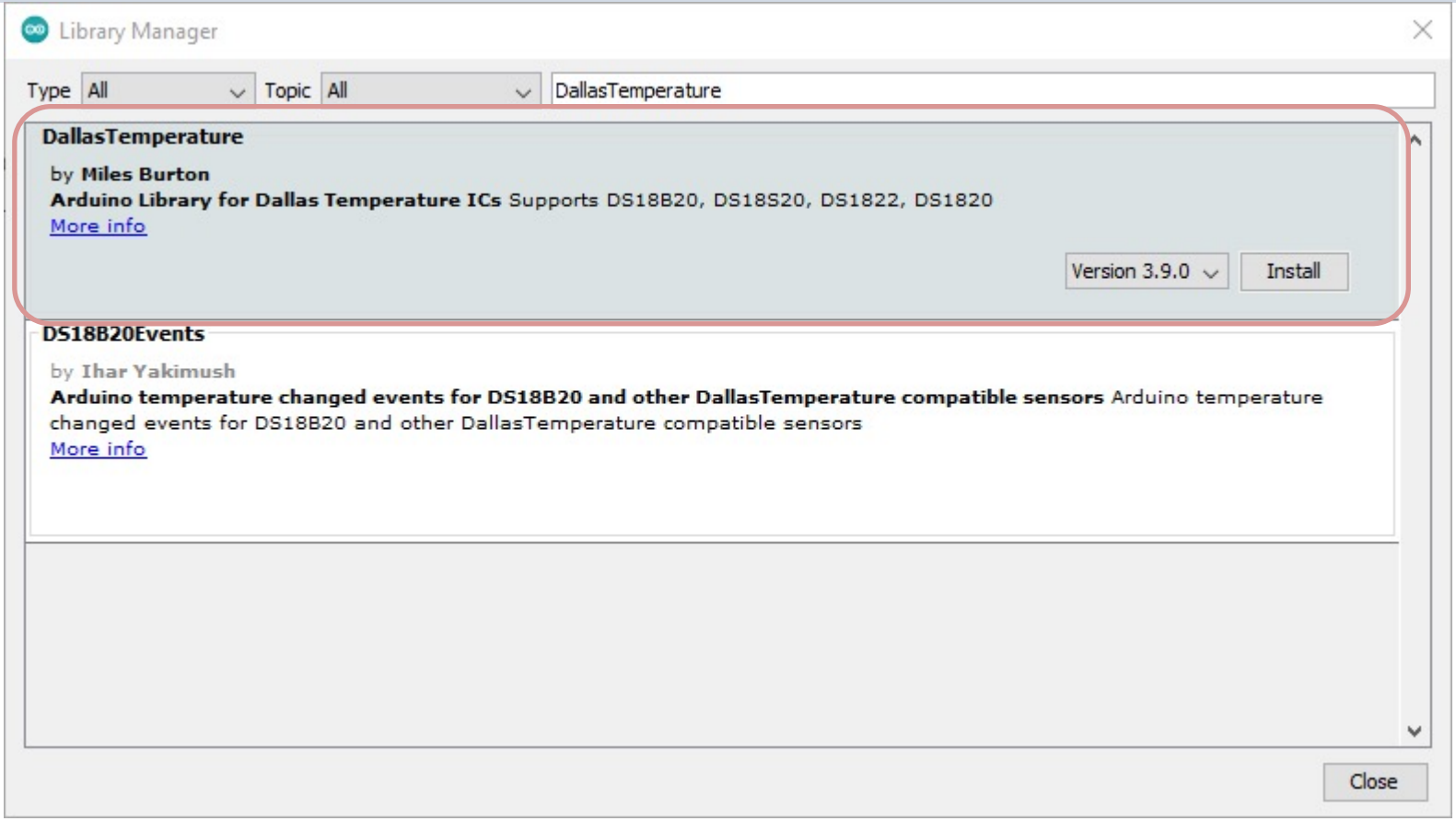
Arduino Code Example

- We use some existing Libraries
  - **OneWire** Library
  - **DallasTemperature** Library
- By using these Libraries, it requires just a few lines of code in order to get the Temperature Value in degrees Celsius

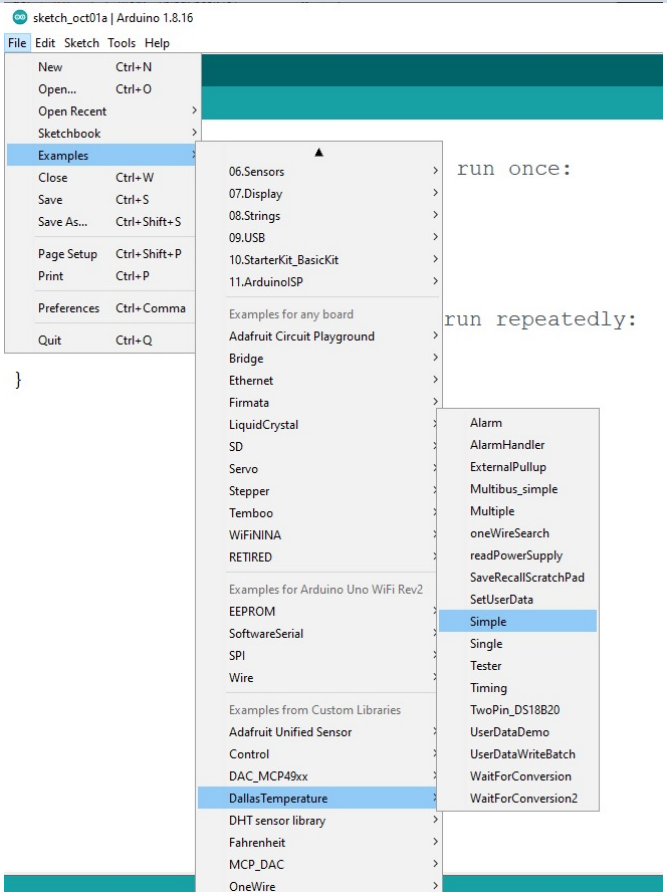
OneWire Library

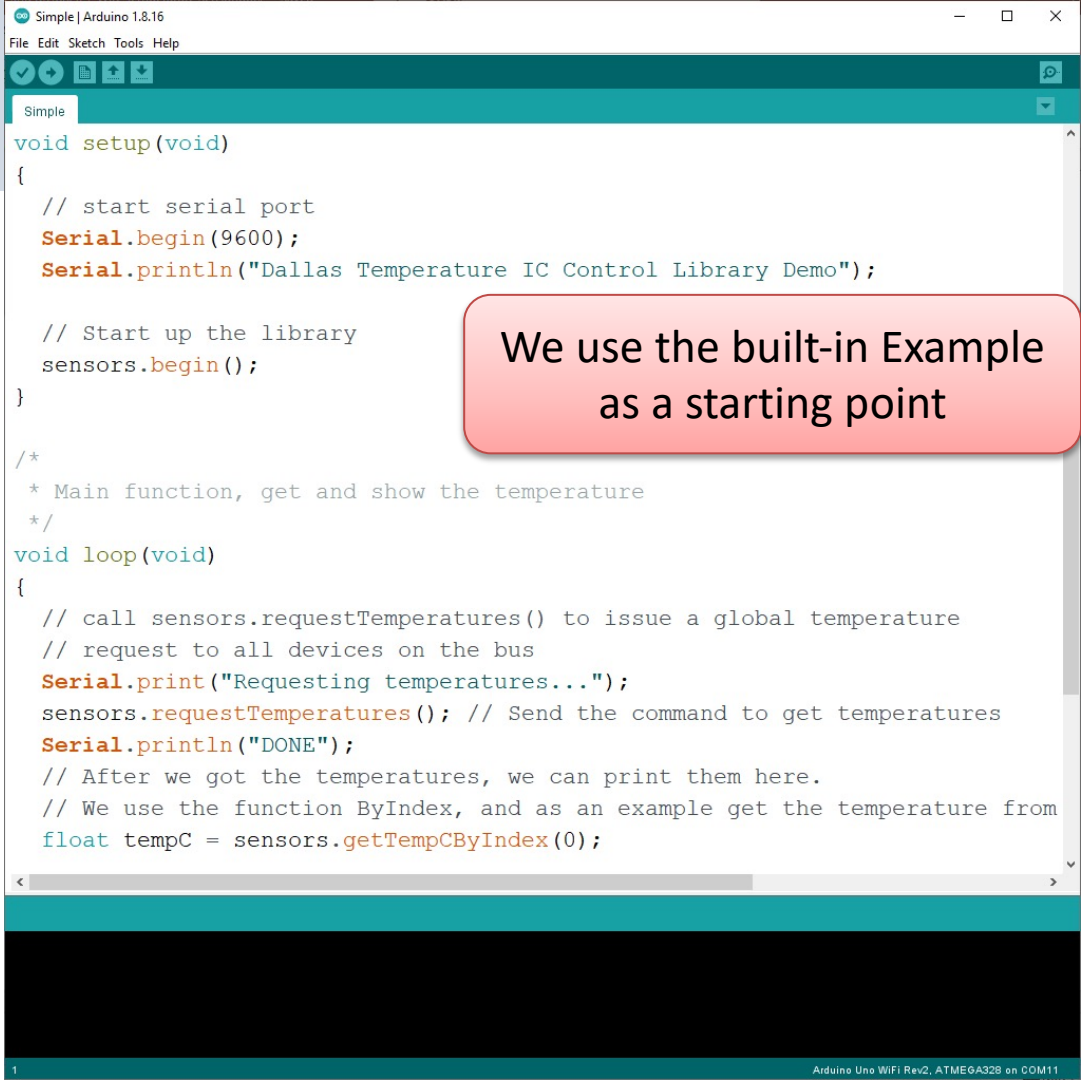


DallasTemperature Library



Example





We use the built-in Example as a starting point

Arduino Code

```
#include <OneWire.h>
#include <DallasTemperature.h>

#define ONE_WIRE_BUS 2

OneWire oneWire(ONE_WIRE_BUS);

DallasTemperature sensors(&oneWire);

float tempCelcius=0;

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

void loop(void)
{
  sensors.requestTemperatures();
  tempCelcius = sensors.getTempCByIndex(0);
  Serial.print("T = ");
  Serial.print(tempCelcius);
  Serial.println("°C");

  delay(1000);
}
```

Serial Monitor

COM6

Send

```
T = 24.25°C
T = 24.25°C
T = 24.31°C
T = 24.25°C
T = 24.31°C
T = 24.31°C
T = 24.25°C
T = 24.31°C
T = 24.31°C
T = 24.31°C
T = 25.25°C
T = 27.12°C
T = 28.25°C
T = 28.31°C
```

Autoscroll Show timestamp Newline 9600 baud Clear output

https://www.halvorsen.blog

Log Sensor Data to ThingSpeak

Hans-Petter Halvorsen

Table of Contents

Log Data to ThingSpeak

- In this Example we will read Temperature data from the DS18B20 Sensor
- Then we will the Temperature data to the ThingSpeak Cloud Service

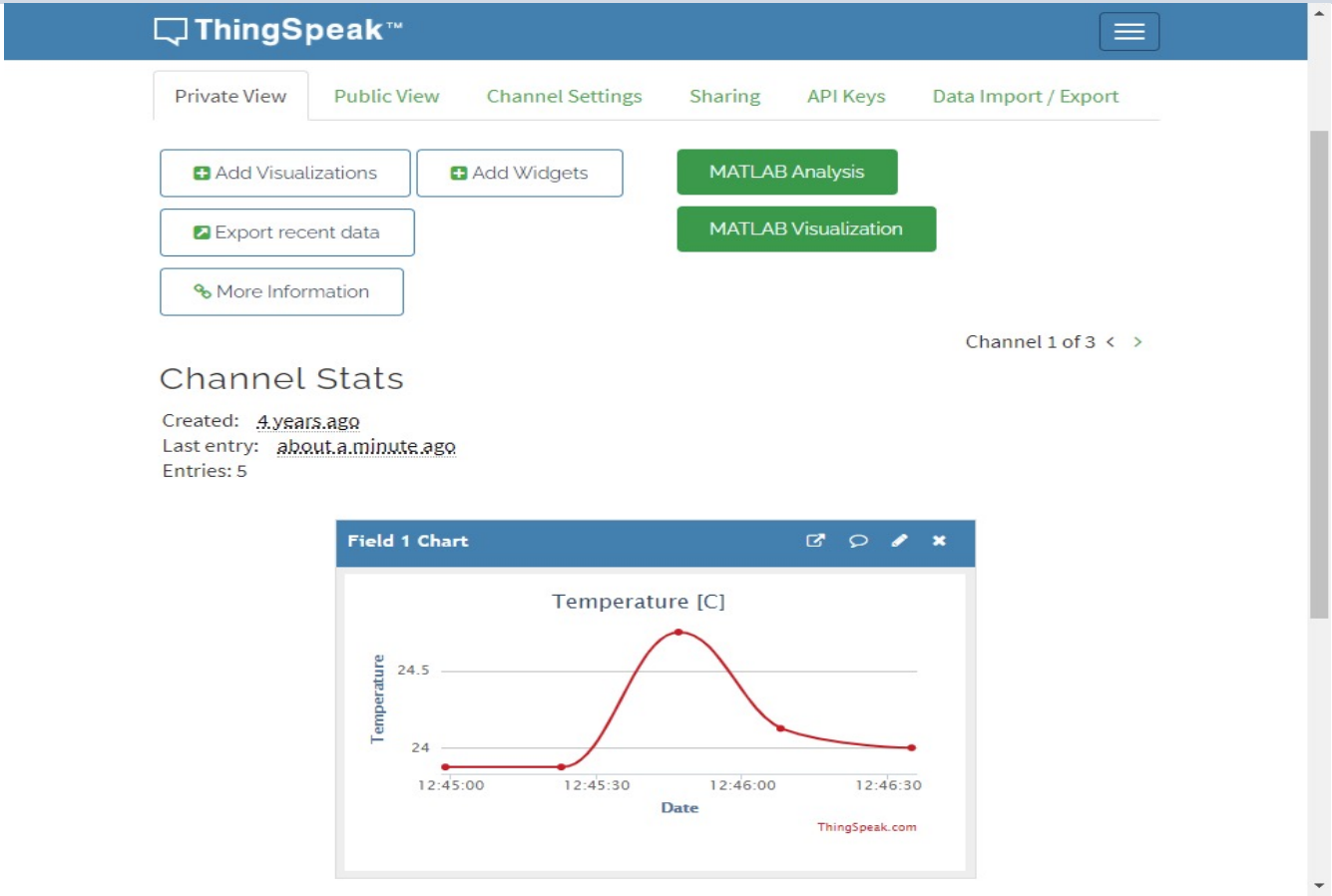
ThingSpeak

- ThingSpeak is an IoT analytics platform service that lets you collect and store sensor data in the cloud and develop Internet of Things (IoT) applications.
- ThingSpeak has a free Web Service (REST API) that lets you collect and store sensor data in the cloud and develop Internet of Things applications.
- It works with Arduino, Raspberry Pi, MATLAB and LabVIEW, Python, etc.

https://thingspeak.com

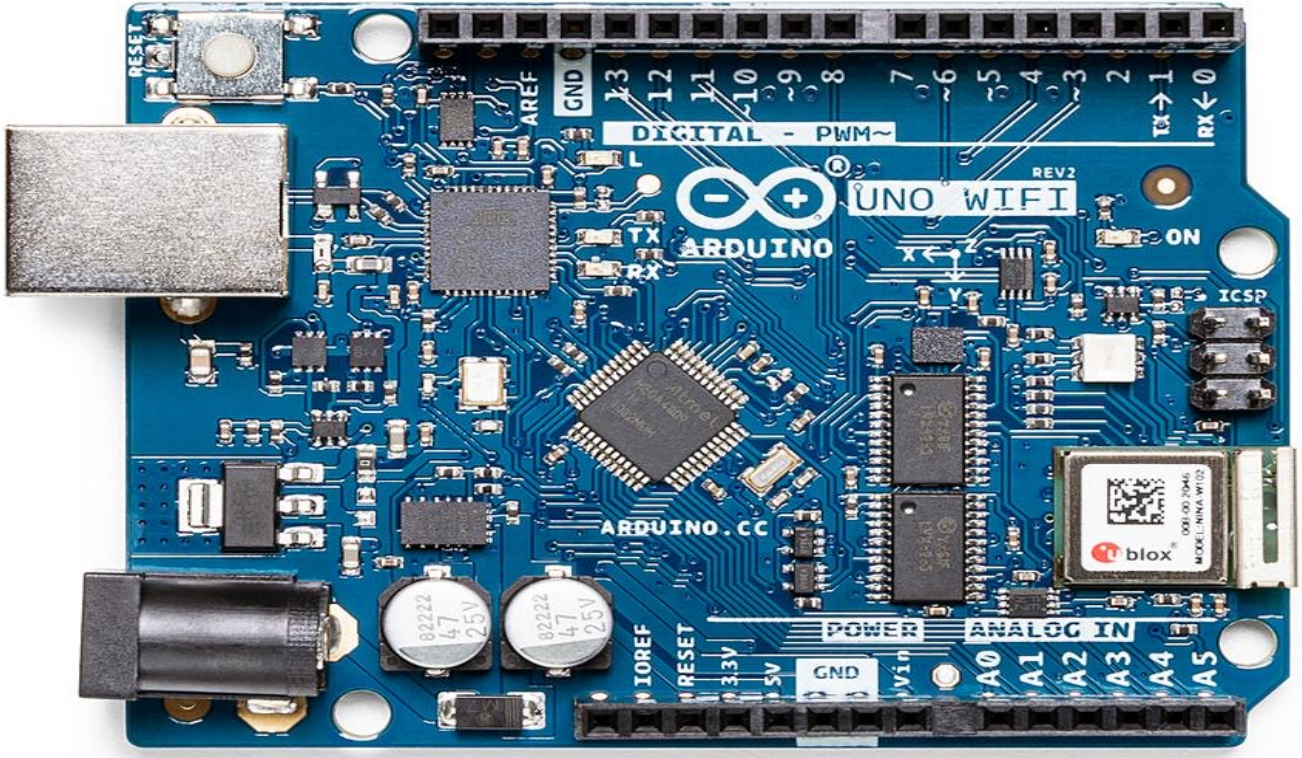
ThingSpeak

https://thingspeak.com

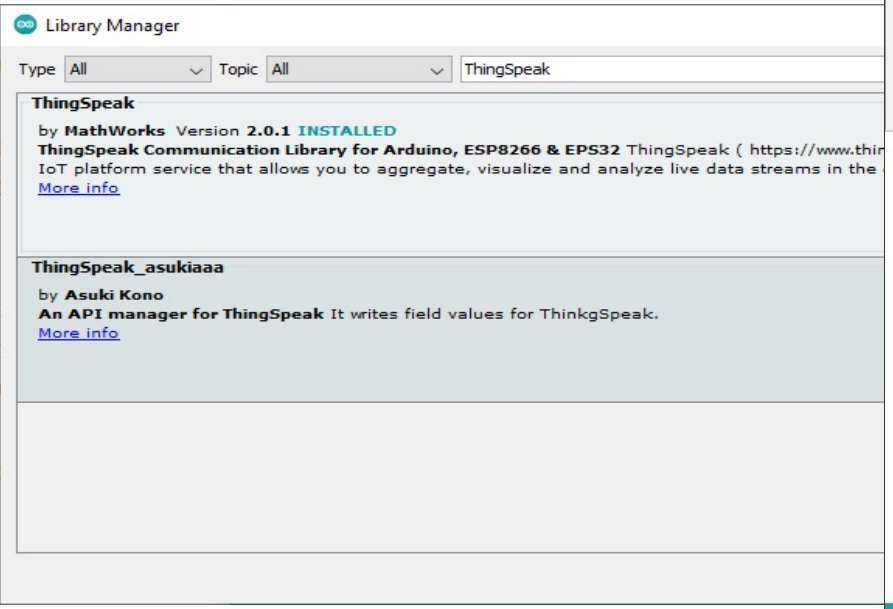


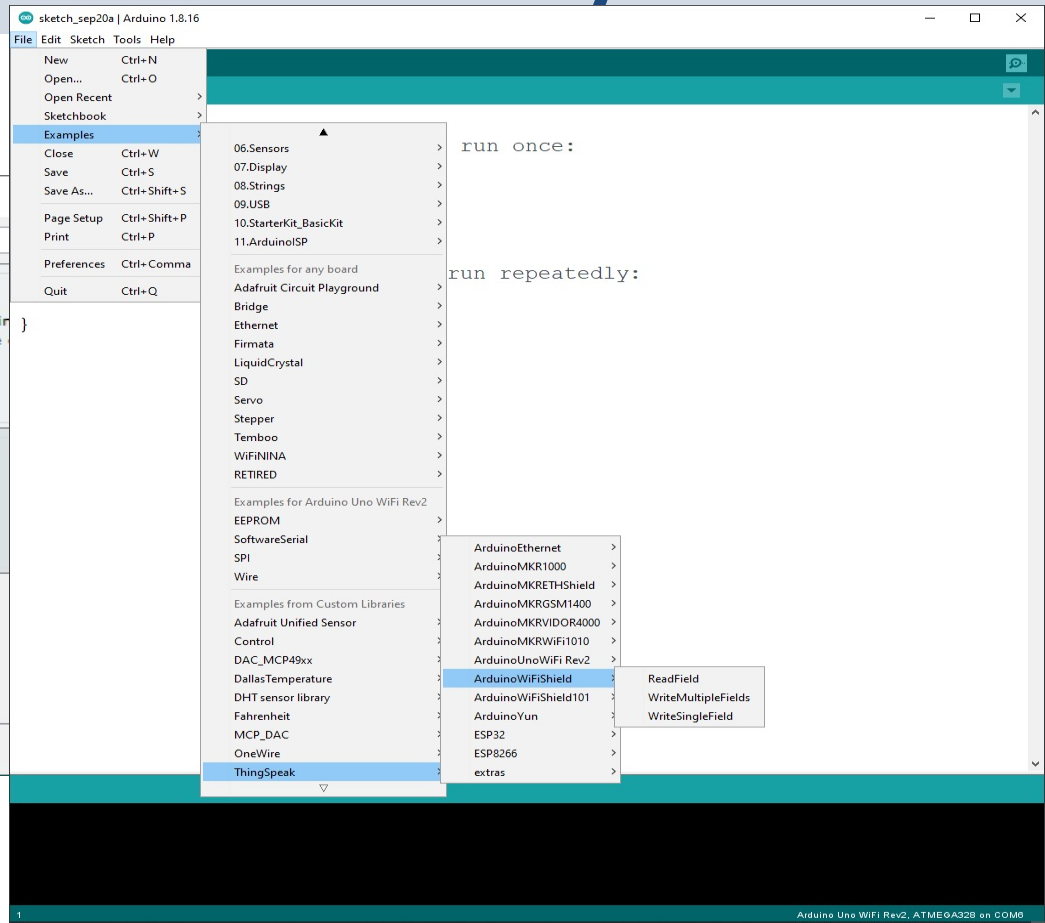
Arduino UNO WiFi Rev 2

The Arduino Uno WiFi is functionally the same as the Arduino Uno Rev3, but with the addition of WiFi / Bluetooth and some other enhancements.



ThingSpeak Library





Arduino Example

WriteSingleField | Arduino 1.8.16

File Edit Sketch Tools Help

WriteSingleField § secrets.h

```
#include <WiFiNINA.h>
#include "secrets.h"
#include "ThingSpeak.h" // always include thingspeak header file after other header f

char ssid[] = SECRET_SSID;    //  your network SSID (name)
char pass[] = SECRET_PASS;   // your network password
int keyIndex = 0;            // your network key Index number (needed only for WEP)
WiFiClient  client;

unsigned long myChannelNumber = SECRET_CH_ID;
const char * myWriteAPIKey = SECRET_WRITE_APIKEY;

int number = 0;

void setup() {
  Serial.begin(115200);  // Initialize serial
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo native USB port only
  }

  // check for the WiFi module:
```

1 Arduino Uno WiFi Rev2, ATMEGA328 on COM11

We use the built-in Example as a starting point

Arduino Code

Here you see the main code structure:

We have created separate Functions for:

- CheckWiFi()
- ConnectWiFi()
- ReadSensorData()
- ThingSpeakWrite()

The Functions are presented on the next pages.

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include "ThingSpeak.h"
#include <WiFiNINA.h>
#include "secrets.h"

#define ONE_WIRE_BUS 2
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature
sensors(&oneWire);

float tempCelcius=0;
WiFiClient  client;
int wait = 20000;

void setup(void)
{
  Serial.begin(9600);
  sensors.begin();
  CheckWiFi();
  ThingSpeak.begin(client);
}

void loop(void)
{
  ConnectWiFi();
  ReadSensorData();
  ThingSpeakWrite();
  delay(wait);
}
```

Arduino Code

Secrets.h

```
#define SECRET_SSID "xxxxxx"
#define SECRET_PASS "xxxxxx"

#define SECRET_CH_ID xxxxxx
#define SECRET_WRITE_APIKEY "xxxxxx"
```

```
void CheckWiFi()
{
  // check for the WiFi module:
  if (WiFi.status() == WL_NO_MODULE) {
    Serial.println("Communication with WiFi module failed!");
    // don't continue
    while (true);
  }

  String fv = WiFi.firmwareVersion();
  if (fv != "1.0.0") {
    Serial.println("Please upgrade the firmware");
  }
}

void ConnectWiFi()
{
  char ssid[] = SECRET_SSID;
  char pass[] = SECRET_PASS;

  if(WiFi.status() != WL_CONNECTED)
  {
    Serial.print("Attempting to connect to SSID: ");
    Serial.println(SECRET_SSID);
    while(WiFi.status() != WL_CONNECTED)
    {
      WiFi.begin(ssid, pass);
      Serial.print(".");
      delay(5000);
    }
    Serial.println("\nConnected.");
  }
}
```

Arduino Code

```
void ReadSensorData()
{
  sensors.requestTemperatures();
  tempCelcius = sensors.getTempCByIndex(0);
  Serial.print("T = ");
  Serial.print(tempCelcius,1);
  Serial.println("°C");
}
```

Arduino Code

Secrets.h

```
#define SECRET_SSID "xxxxxx"
#define SECRET_PASS "xxxxxx"

#define SECRET_CH_ID xxxxxx
#define SECRET_WRITE_APIKEY "xxxxxx"
```

```
void ThingSpeakWrite()
{
  unsigned long myChannelNumber = SECRET_CH_ID;
  const char * myWriteAPIKey = SECRET_WRITE_APIKEY;

  int channelField = 1;

  int x = ThingSpeak.writeField(myChannelNumber, channelField, tempCelcius, myWriteAPIKey);
  if(x == 200){
    Serial.println("Channel update successful.");
  }
  else{
    Serial.println("Problem updating channel. HTTP error code " + String(x));
  }
}
```

Serial Monitor

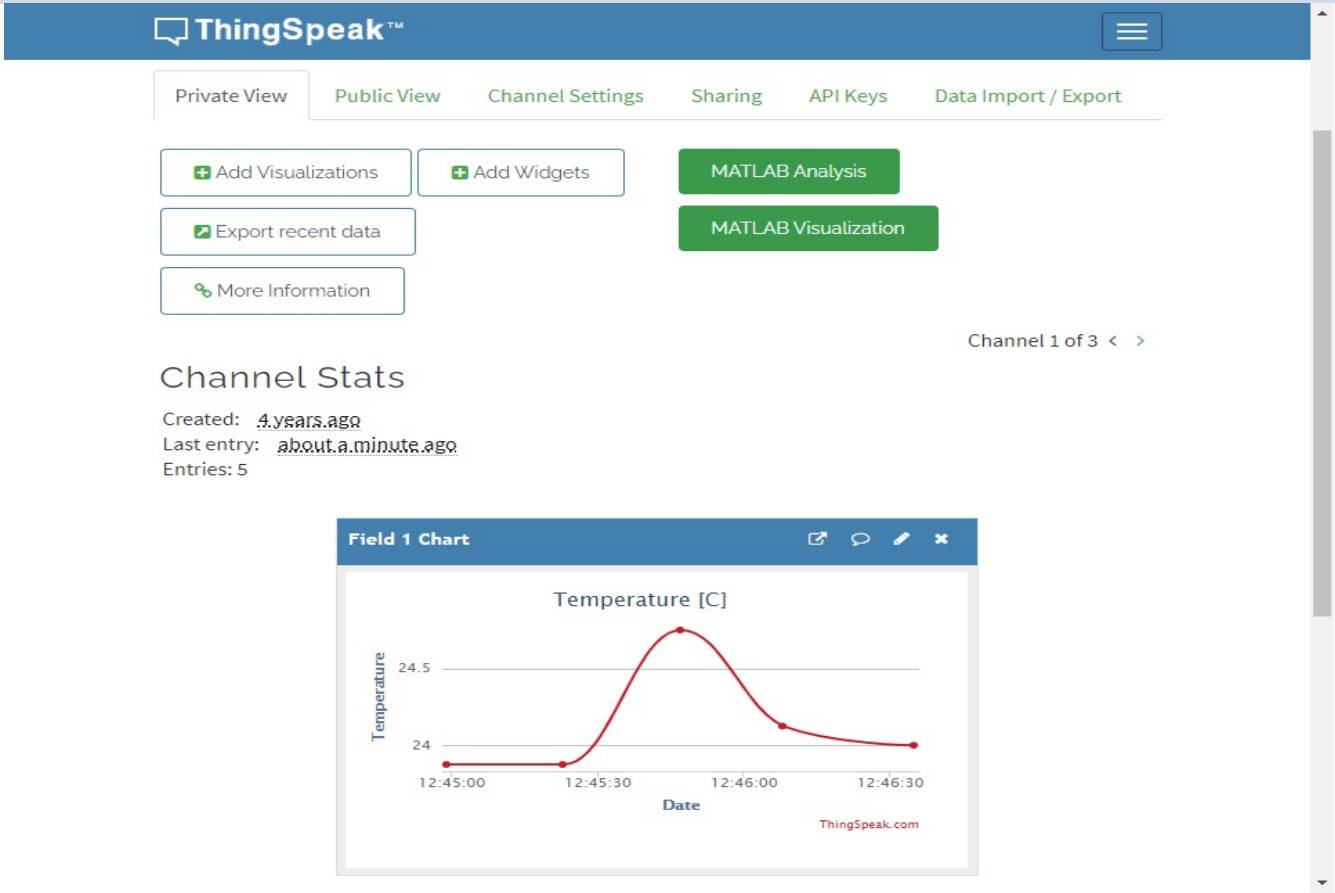
COM11

Send

```
Channel update successful.
T = 23.9°C
Channel update successful.
T = 24.7°C
Channel update successful.
T = 24.1°C
Channel update successful.
T = 24.0°C
Channel update successful.
T = 23.9°C
Channel update successful.
T = 23.9°C
Channel update successful.
T = 23.9°C
Channel update successful.
T = 23.9°C
```

Autoscroll Show timestamp Newline 9600 baud Clear output

ThingSpeak


ThingSpeak™
Private View
Public View
Channel Settings
Sharing
API Keys
Data Import / Export
Add Visualizations
Add Widgets
MATLAB Analysis
Export recent data
MATLAB Visualization
More Information
Channel 1 of 3
Channel Stats
Created: 4 years ago
Last entry: about a minute ago
Entries: 5
Field 1 Chart
Temperature [C]
Temperature
24.5
24
12:45:00
12:45:30
12:46:00
12:46:30
Date
ThingSpeak.com

References

- https://create.arduino.cc/projecthub/TheGadgetBoy/ds18b20-digital-temperature-sensor-and-arduino-9cc806
- https://create.arduino.cc/projecthub/iotboys/how-to-use-ds18b20-water-proof-temperature-sensor-2adecc
- https://lastminuteengineers.com/ds18b20-arduino-tutorial/

Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

E-mail: hans.p.halvorsen@usn.no

Web: https://www.halvorsen.blog