

[dokumentazioa](#)

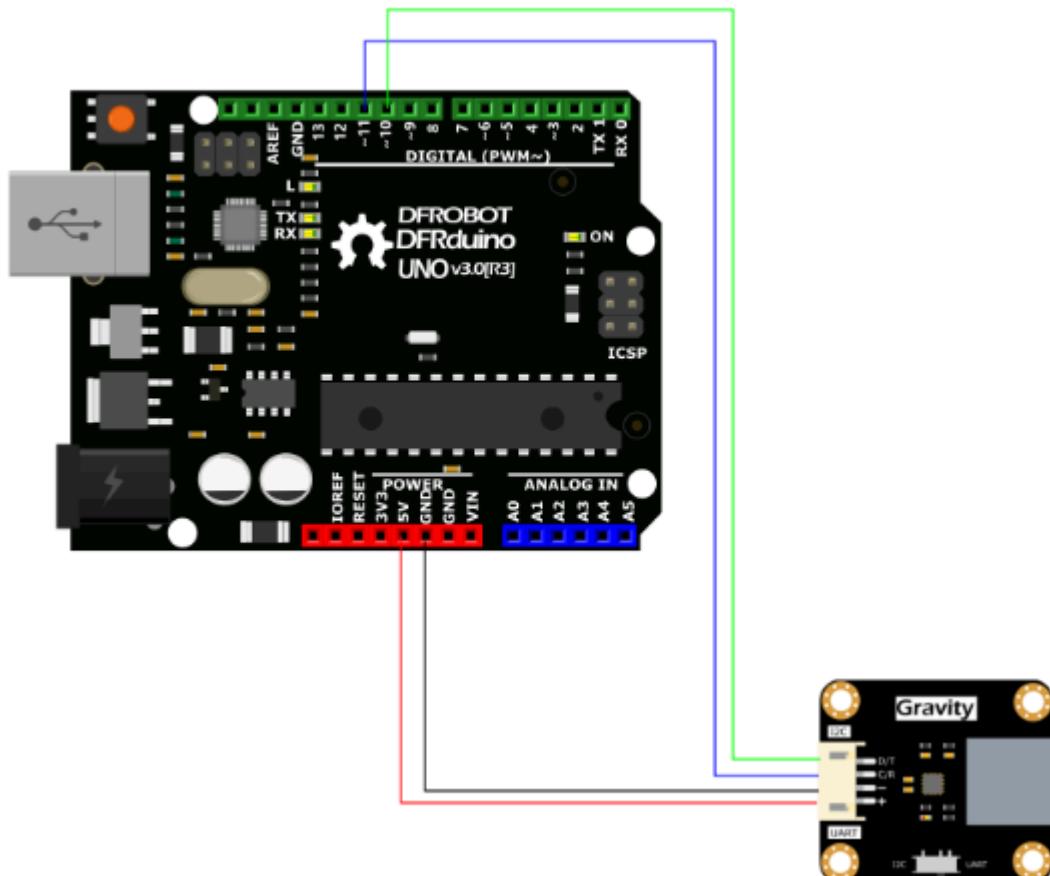
Modeloa:

SEN0575

Arduino:

-Nola konektatu

Connection Diagram



Behar den liburutegia:

Rainfall library (Githuben dago)

Programa:

```
#include "DFRobot_RainfallSensor.h"

#define MODE_UART
#ifndef MODE_UART //UART communication

#include "SoftwareSerial.h"
SoftwareSerial mySerial(/*rx */10, /*tx */11);
DFRobot_RainfallSensor_UART Sensor(/*Stream */&mySerial);

#else //I2C communication
```

```

DFRobot_RainfallSensor_I2C Sensor(&Wire);
#endif

void setup(void)
{
#ifdef MODE_UART
    mySerial.begin(9600);
#endif
    Serial.begin(115200);

    delay(1000);
    while(!Sensor.begin()){
        Serial.println("Sensor init err!!!");
        delay(1000);
    }
    Serial.print("vid:\t");
    Serial.println(Sensor.vid,HEX);
    Serial.print("pid:\t");
    Serial.println(Sensor.pid,HEX);
    Serial.print("Version:\t");
    Serial.println(Sensor.getFirmwareVersion());
    //Set the accumulated rainfall value, unit: mm
    //Sensor.setRainAccumulatedValue(0.2794);
}

void loop()
{
    // Get the sensor operating time, unit: hour
    Serial.print("Sensor WorkingTime:\t");
    Serial.print(Sensor.getSensorWorkingTime());
    Serial.println(" H");
    //Get the accumulated rainfall during the sensor operating time
    Serial.print("Rainfall:\t");
    Serial.println(Sensor.getRainfall());
    //Get the accumulated rainfall in the past 1 hour (function parameter can be 1-24)
    Serial.print("1 Hour Rainfall:\t");
    Serial.print(Sensor.getRainfall(1));
    Serial.println(" mm");
    // Get the raw data, number of tipping bucket counts
    Serial.print("rainfall raw:\t");
    Serial.println(Sensor.getRawData());
    delay(1000);
}

```

ESP-32:

Konexioa:

Sensor	ESP32
VCC	3.3V o 5V (<i>según tu módulo</i>)
GND	GND
SDA	GPIO 21
SCL	GPIO 22

Programa:

```
#include <Wire.h>
#include "DFRobot_RainfallSensor.h"

// Crear objeto del sensor por I2C
DFRobot_RainfallSensor_I2C Sensor(&Wire);

void setup() {
  Serial.begin(115200);
  delay(1000);

  // Iniciar I2C en pines del ESP32
  Wire.begin(21, 22); // SDA, SCL

  Serial.println("Iniciando sensor de lluvia...");

  while (!Sensor.begin()) {
    Serial.println("Error al iniciar el sensor !!!");
    delay(1000);
  }

  Serial.println("Sensor iniciado correctamente");

  Serial.print("VID:\t");
  Serial.println(Sensor.vid, HEX);

  Serial.print("PID:\t");
  Serial.println(Sensor.pid, HEX);

  Serial.print("Firmware:\t");
  Serial.println(Sensor.getFirmwareVersion());
}
```

```
void loop() {
    // Tiempo de funcionamiento del sensor
    Serial.print("Tiempo de trabajo:\t");
    Serial.print(Sensor.getSensorWorkingTime());
    Serial.println(" h");

    // Lluvia total acumulada
    Serial.print("Lluvia acumulada:\t");
    Serial.print(Sensor.getRainfall());
    Serial.println(" mm");

    // Lluvia en la última hora
    Serial.print("Lluvia última 1h:\t");
    Serial.print(Sensor.getRainfall(1));
    Serial.println(" mm");

    // Datos crudos (conteo del balancín)
    Serial.print("Raw data:\t");
    Serial.println(Sensor.getRawData());

    Serial.println("-----");
    delay(1000);
}
```