

# Documentation

## INTRODUCTION

Le but de notre projet ARE est de déclencher une caméra Go Pro à l'aide d'une carte électronique Arduino dans le but de récolter des données. Les données qui seront les causes du déclenchement pourront être le pH de l'eau, la température etc. Cela appelle aussi à une adaptation de la caméra au milieu avec un processus d'éclairage la nuit, un potentiel processus de détection des espèces. Le cœur du sujet résidera donc dans le fait de déclencher la caméra sous l'action de certaines conditions nécessaires. Le dessein final serait de mettre le système dans des conditions réelles en milieu extérieur, ce pendant plusieurs jours, afin d'en observer les résultats.

Objectif : Intégration d'une carte mesures de pH + température et d'une Arduino pour la programmation du déclenchement d'une caméra Go Pro autonome.

Il s'agit d'interfaçage de l'Arduino avec les cartes de mesures de pH SENSE0161 (ou ADCM355), avec enregistrement des mesures sur carte microSD, pour pouvoir réveiller la caméra sur un seuil de mesure de température ou de pH.

Matériel fourni :

- 1 Arduino Nano
- 1 carte support microSD
- 2 Cartes mesure SENSE0161 DFRobot avec électrodes pH
- 2 cartes mesures ADUCM355 WATER QUALITY SENSOR (T°C + pH ou REDOX(ORP))
- 2 sondes T°C long. 30 cm - thermistance (Thermocoax)

## CODE

```
#include <SPI.h>

#include <WiFi101.h>

#include <GoProController.h>

char ssid[] = "VotreSSID"; // Remplacez "VotreSSID" par le nom de votre réseau Wi-Fi

char pass[] = "VotreMotDePasse"; // Remplacez "VotreMotDePasse" par le mot de passe de votre réseau Wi-Fi
```

```
WiFiClient client;
```

```
GoProController gopro(client);
```

```
const int sensorPin = A0; // Définissez le numéro de broche de votre capteur
```

```
const int seuil = 500; // Définissez le seuil de déclenchement de votre capteur
```

```
bool shouldTrigger = false;
```

```
void setup() {
```

```
    // Initialisez votre connexion Wi-Fi
```

```
    WiFi.begin(ssid, pass);
```

```
    while (WiFi.status() != WL_CONNECTED) {
```

```
        delay(1000);
```

```
    }
```

```
    // Connectez-vous à votre GoPro Hero 3
```

```
    gopro.connect();
```

```
}
```

```
void loop() {

    // Lire la valeur de votre capteur et déterminer si la GoPro doit être déclenchée ou non

    if (analogRead(sensorPin) > seuil) {

        shouldTrigger = true;

    } else {

        shouldTrigger = false;

    }


    // Si la GoPro doit être déclenchée, envoyez la commande de déclenchement

    if (shouldTrigger) {

        gopro.command(BacPac::COMMAND_SHUTTER, BacPac::VALUE_SHUTTER_START);

        delay(5000); // Attendez 5 secondes

        gopro.command(BacPac::COMMAND_SHUTTER, BacPac::VALUE_SHUTTER_STOP);

    }

}
```

## **BIBLIOGRAPHIE**

1. *Analog Input | Arduino Documentation*. <https://docs.arduino.cc/built-in-examples/analog/AnalogInput>. Consulté le 30 mars 2023.
2. *analogRead() - Arduino Reference*. <https://www.arduino.cc/reference/en/language/functions/analog-io/analogread/>. Consulté le 30 mars 2023.

3. *Nano | Arduino Documentation*. <https://docs.arduino.cc/hardware/nano>. Consulté le 30 mars 2023.
  4. *SPI - Arduino Reference*. <https://www.arduino.cc/reference/en/language/functions/communication/spi/>. Consulté le 30 mars 2023.
  5. *WiFi101 - Arduino Reference*. <https://www.arduino.cc/reference/en/libraries/wifi101/>. Consulté le 30 mars 2023.
  6. « Arduino - Temperature Sensor | Arduino Tutorial ». *Arduino Getting Started*, <https://arduinogetstarted.com/tutorials/arduino-temperature-sensor>. Consulté le 30 mars 2023.
  7. *DS18B20 Temperature Sensor For Arduino - DFRobot Wiki*. [https://wiki.dfrobot.com/Gravity\\_\\_DS18B20\\_Temperature\\_Sensor\\_\\_Arduino\\_Compatible\\_\\_V2\\_SKU\\_\\_DFR0024](https://wiki.dfrobot.com/Gravity__DS18B20_Temperature_Sensor__Arduino_Compatible__V2_SKU__DFR0024). Consulté le 30 mars 2023.
- 

Revision #1

Created 6 April 2023 12:24:06 by Thomas Adrien

Updated 6 April 2023 12:30:06 by Thomas Adrien