

COMPTE RENDU DE TP: Réalisation Systemes BE - Maison connectée.



Encadrant: THIERRY PERISSE.

Année 2021/2022

Tables des matières

1.Introduction	3
2.Vue d'ensemble du BE	3
2.1.Schéma fonctionnel du TP global	3
2.2.Schéma de câblage	4
3.Connexion de la photorésistance au STM32 (mesure de la lumière)	4
3.1.Fonctionnement général	5
3.2.Vérification	5
3.3.Schéma de câblage	6
3.4.Initialisation des ports	7
4.Connexion du Adjustable PIR Motion Sensor au STM32 (détection de présence)	8
4.1.Fonctionnement général	9
4.2.Spécification	9
4.3.Schéma de câblage	9
4.4.Initialisation des ports	10
4.5.Observation	12
5.Conclusion	13

1.Introduction

Nom du projet : Maison connectée

But: Dans ce BE final, nous allons faire fonctionner deux capteurs, un capteur de luminosité et un autre capteur pour la détection de mouvements. Lorsque le PIR détecte un mouvement, il enverra l'information à la photorésistance, qui elle à son tour devra observer la luminosité, et en fonction donner l'information à une Led pour savoir s'il faut allumer la lumière ou non.

Matériel à disposition :

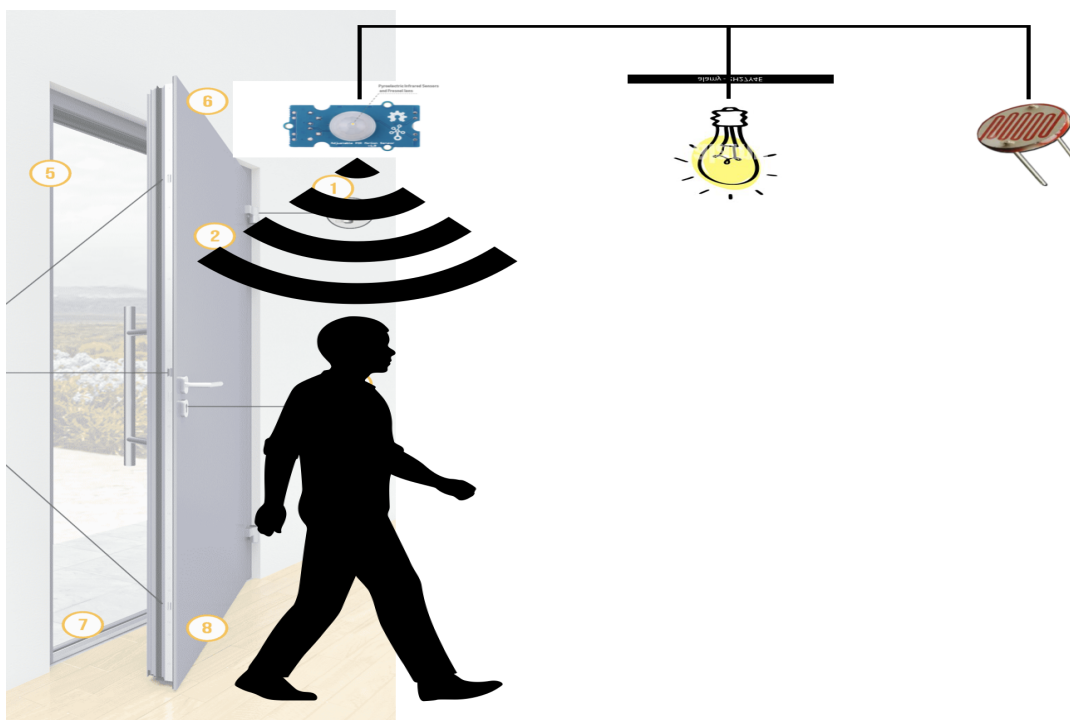
- Nucléo STM32L152RE
- LCD RGB Backlight V4.0
- Potorésistance LDR
- adjustable PIR motion sensor

Logiciel mis à disposition :

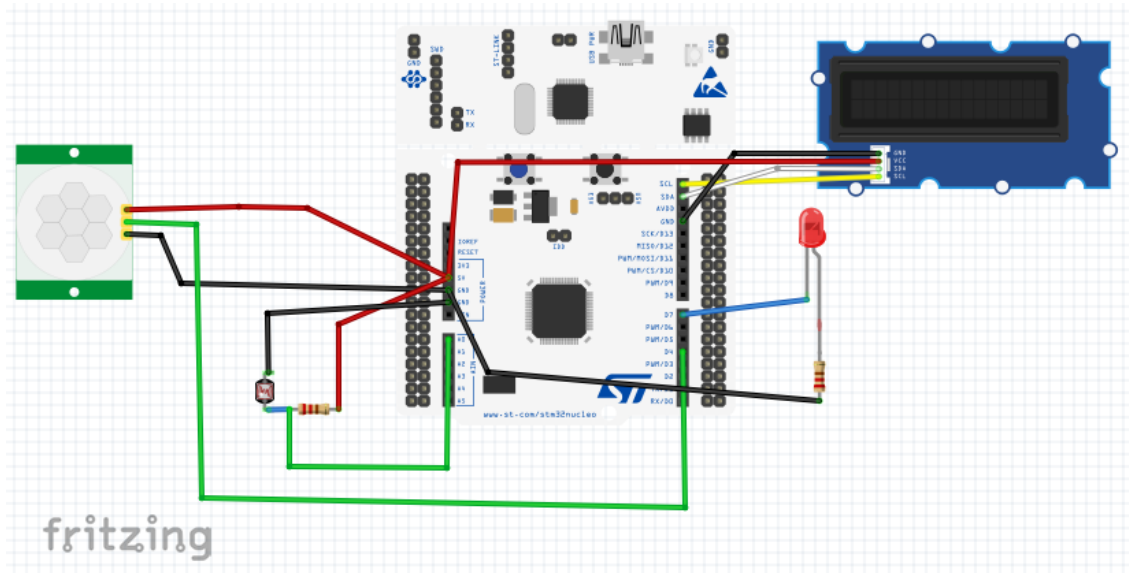
- STM32CUBEIDE

2. Vue d'ensemble du BE

2.1.Schéma fonctionnel du TP global



2.2. Schéma de câblage



3. Connexion de la photorésistance au STM32 (mesure de la lumière)



La résistance peut être utilisée pour la valeur de l'éclairage, en l'utilisant comme convertisseur primaire, mais dans notre cas nous allons simplement l'utiliser pour déterminer la présence de lumière avec un affichage sur écran Lcd du message "Il fait jour" ou "il fait nuit".

3.1. Fonctionnement général

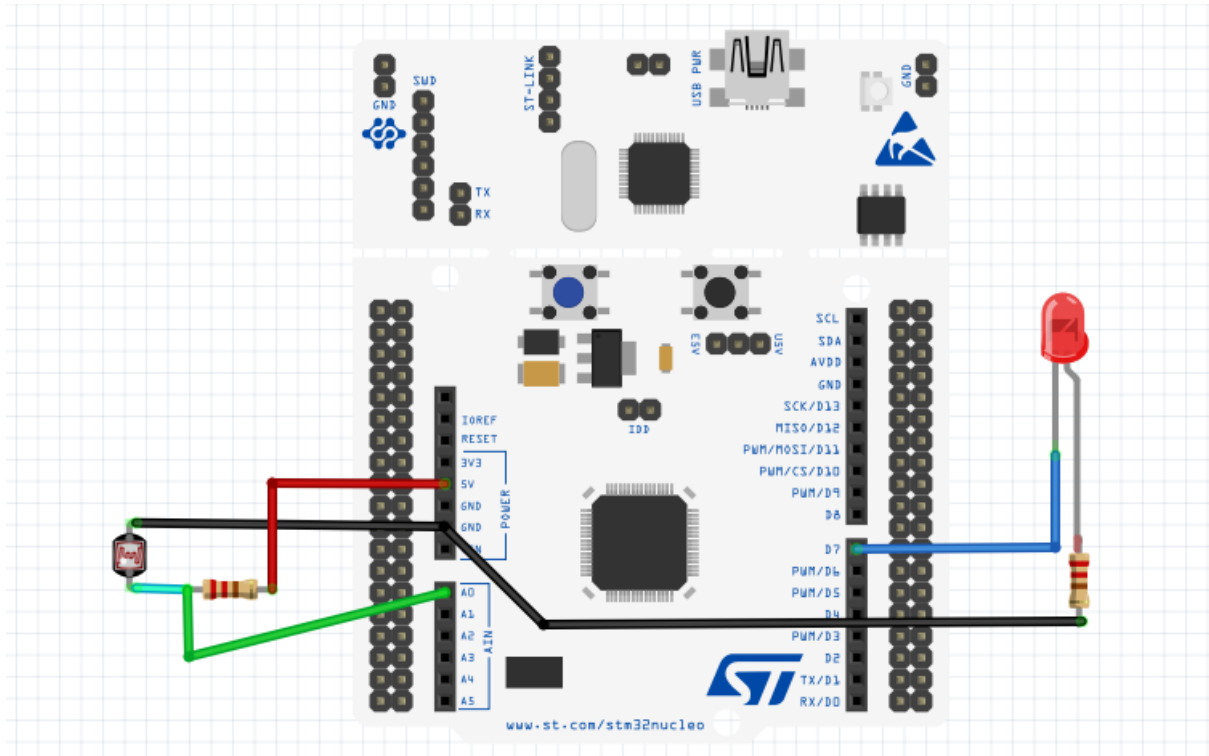
Selon le type de matériau semi-conducteur utilisé pour la photorésistance, leur plage de résistance et leur sensibilité diffèrent. En l'absence de lumière, la photorésistance peut avoir des valeurs de résistance en Méga-ohms. Et en présence de lumière, sa résistance peut diminuer jusqu'à quelques centaines d'ohms.

3.2. Vérification

Il suffit de connecter n'importe quelle photorésistance à l'ohmmètre et de voir comment sa résistance change lorsque l'éclairage change.

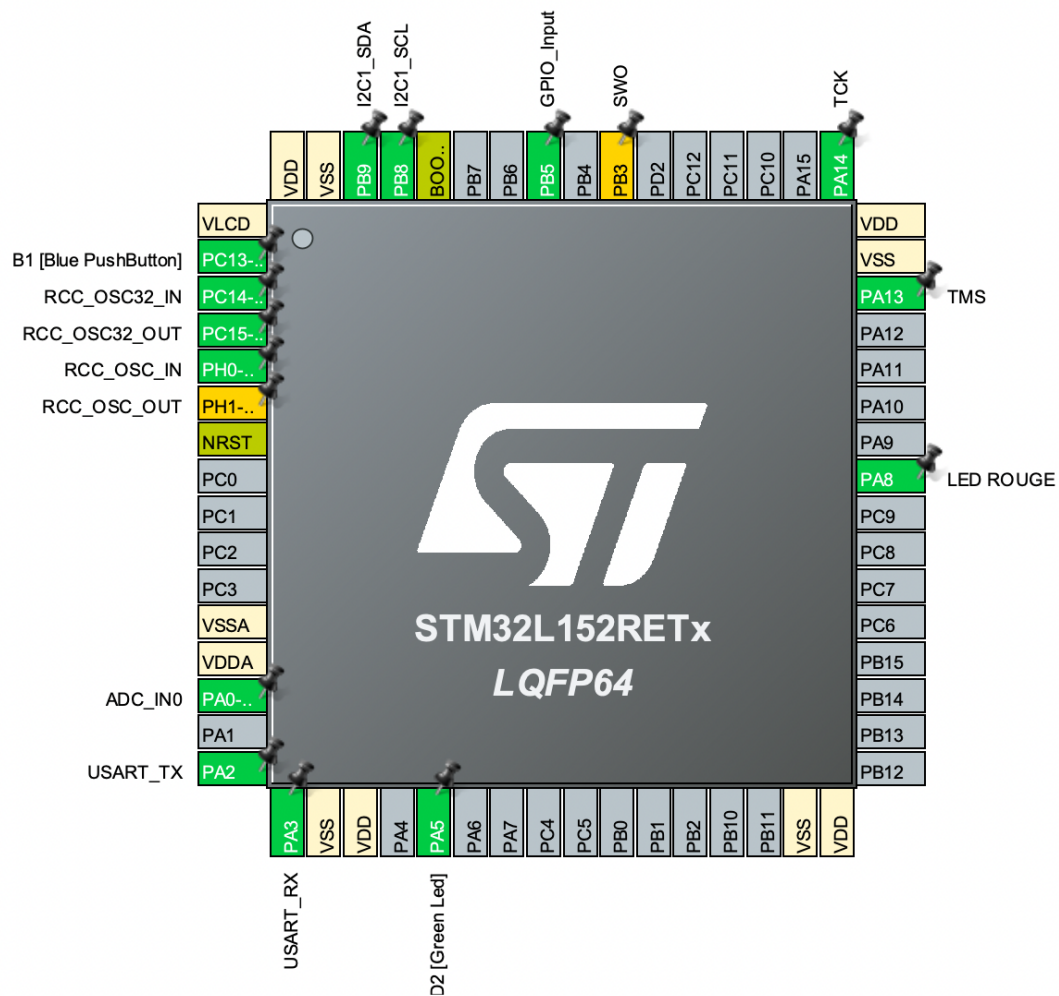
Éclairage	R , Ohm
Temps nuageux	1,5 kOhm
Obscurité (couverte à la main)	500 kOhm
Lumière vive (flashlight)	100 ohms

3.3. Schéma de câblage



La photorésistance lorsqu'il fait nuit, fera appel à la Led Rouge et lui ordonnera de s'allumer, dans le cas contraire elle restera éteinte si la photorésistance reçoit assez de lumière.

3.4. Initialisation des ports



Il nous faut ici mettre le PA0 en mode analogique ADC,
La Led rouge sera elle initialisée au PA8 en Output.

Créer des variables globales :

```
/* USER CODE BEGIN 1 */
uint16_t readValue;
char msg[10];
/* USER CODE END 1 */
```

Fonction readValue, qui va lire la valeur en décimale de la conversion qui à été réalisé:

```

HAL_ADC_Start(&hadc);
HAL_ADC_PollForConversion(&hadc, HAL_MAX_DELAY);
readValue = HAL_ADC_GetValue(&hadc);

sprintf(msg, "%hu\r\n", readValue);
HAL_UART_Transmit(&huart2, (uint8_t*)msg, strlen(msg), HAL_MAX_DELAY);

```

Ci dessous, un code pour afficher sur le Lcd un message en fonction de la valeur renvoyée par la photorésistance. En dessous d'un seuil de la valeur renvoyée par la photorésistance en hexadécimal, le message "IL FAIT NUIT" sera affiché. Dans le cas inverse, si on est au dessus du seuil, on affichera "IL FAIT JOUR" sur l'écran LCD".

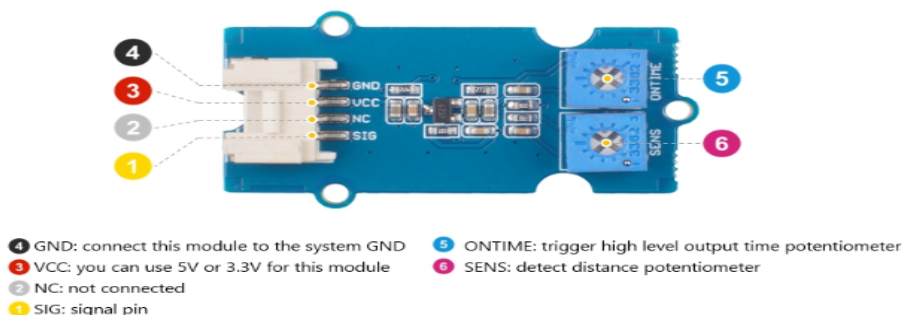
```

HAL_Delay(500);
char res [10];

if (readValue>3000)
{
    lcd_position(&hi2c1,2,1);
    lcd_print(&hi2c1," IL FAIT JOUR");
}else
{
    lcd_position(&hi2c1,2,1);
    lcd_print(&hi2c1,"IL FAIT NUIT\n");
}

```

4.Connexion du Adjustable PIR Motion Sensor au STM32 (détection de présence)



La liaison entre notre détecteur de présence et le microcontrôleur du STM32 se fait en 1-wire (on a pris la broche A1 du STM32 avec SIG du PIR pour sa connexion). Dans notre BE, le principe est de détecter s'il y a présence ou pas (si oui on reçoit sur le moniteur série le mot "motion", cela veut dire qu'on a eu détection et on envoie

l'information à la photorésistance pour qu'elle voit s'il fait jour ou nuit et voir si la lumière doit s'allumer ou pas).

4.1. Fonctionnement général

Grove - Adjustable PIR Motion Sensor est un capteur de mouvement infrarouge passif facile à utiliser, qui peut détecter le mouvement d'un objet infrarouge jusqu'à 3 mètres. Tout objet infrarouge se déplace dans sa plage de détection, le capteur émet HIGH sur sa broche SIG. Et on peut régler le temps SIG HIGH jusqu'à 130s via le potentiomètre, de plus, on peut régler la plage de détection via l'autre potentiomètre.

4.2. Spécification

Notre PIR fonctionne selon les informations données dans le tableau ci-dessous :

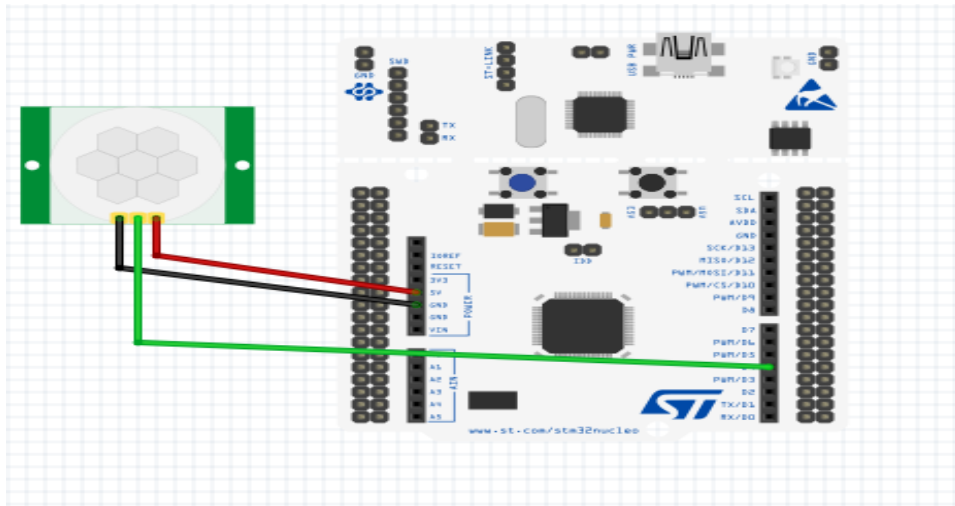
Item	Value
Operating Voltage	3.3V / 5V
Operating Temperature	-30°C ~ 70°C
Viewing Angel	Horizontal placement 80° Vertical placement 55°
Detction Spectral Response(λ)	5nm ~ 14nm
Detction Distance	0 ~ 3m
High Level On Time	<130s
Interface	Digital

4.3. Schéma de câblage

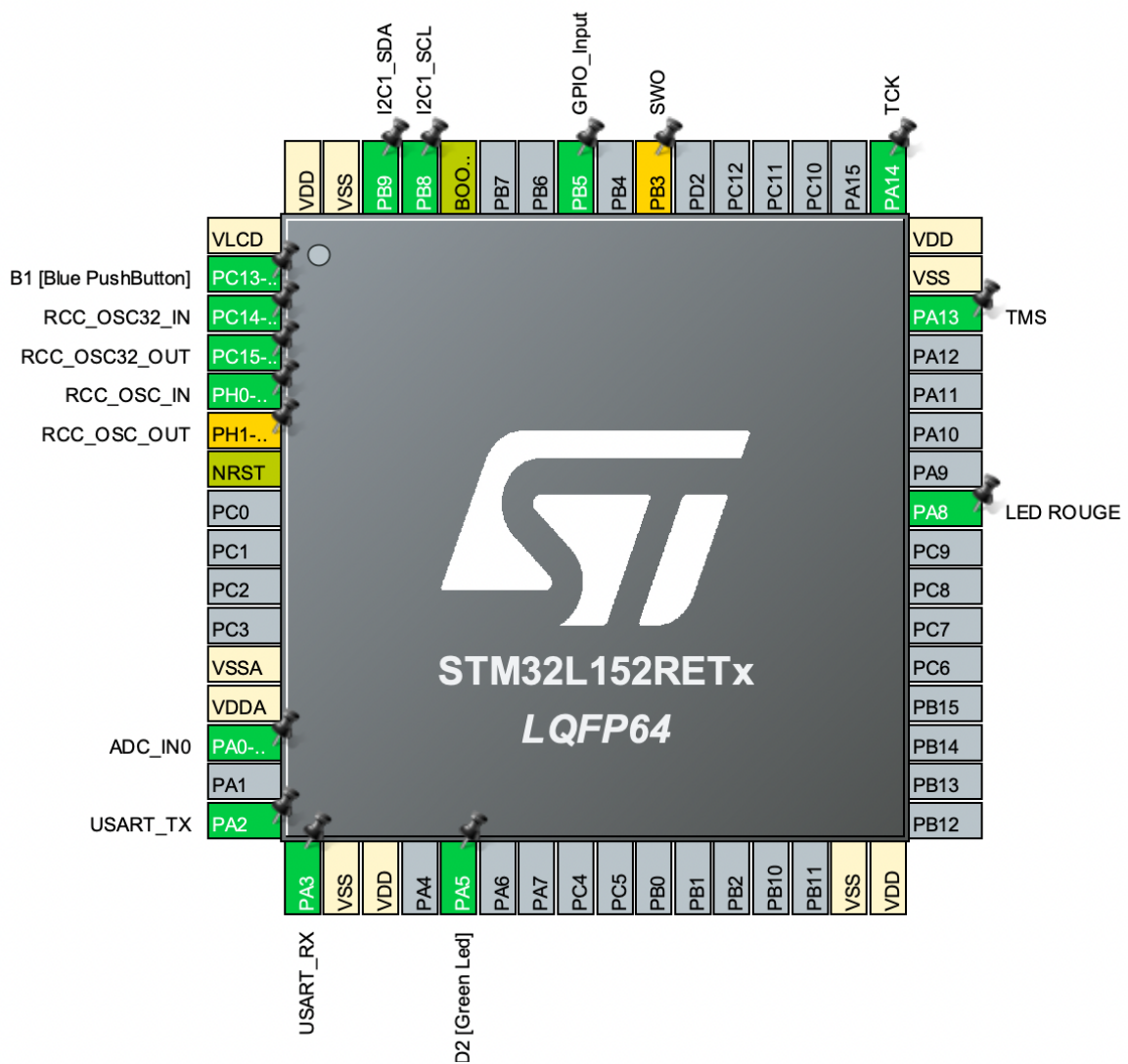
Noir : GND

Rouge : VCC

Vert : SIG Data



4.4. Initialisation des ports



Le capteur PIR est un capteur infrarouge passif qui détecte les rayonnements infrarouges (comme son nom l'indique). Lorsqu'un mouvement est détecté, le capteur envoie un signal élevé (1)

à partir de sa broche OUT, l'état haut persistera environ 5 secondes, puis passera à l'état bas.

Le STM32 détectera ces niveaux élevés avec sa broche IN et allume la LED bleue (PB15) pendant quelques secondes.

L'état de la LED sera indiqué sur le terminal, lorsqu'il est allumé (Mouvement détecté) lorsqu'il est OFF (Aucun mouvement détecté).

Pour le fonctionnement du PIR, on a simplement configuré ici le Pin NAME "D7" qui est le Pin sur STM32 "PB5" en INPUT.

Ensuite on a observé à la fois à l'oscilloscope et au picoscope le signal de la détection pour savoir de quel signal il s'agit.

Ci-dessous le code pour le fonctionnement du PIR. Ici, on a mis un if pour lier la détection et la fonction readValue de la photorésistance, pour allumer la lumière que lorsque les deux conditions sont respectées. C'est à dire qu'il faut tout d'abord que le détecteur renvoie la valeur "1" et qu'ensuite la valeur de la photorésistance soit inférieure au seuil pour pouvoir allumer la LED.

```
/* USER CODE END WHILE */
    strcpy((char*)buf, "NONE\r\n");

    HAL_UART_Transmit(&huart2, buf, strlen((char*)buf), HAL_MAX_DELAY);

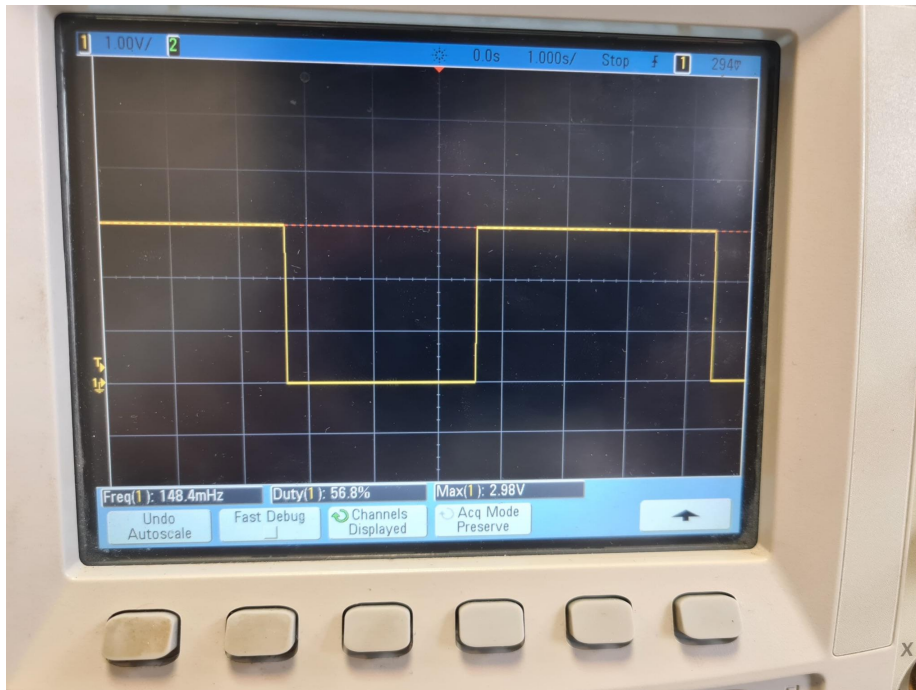
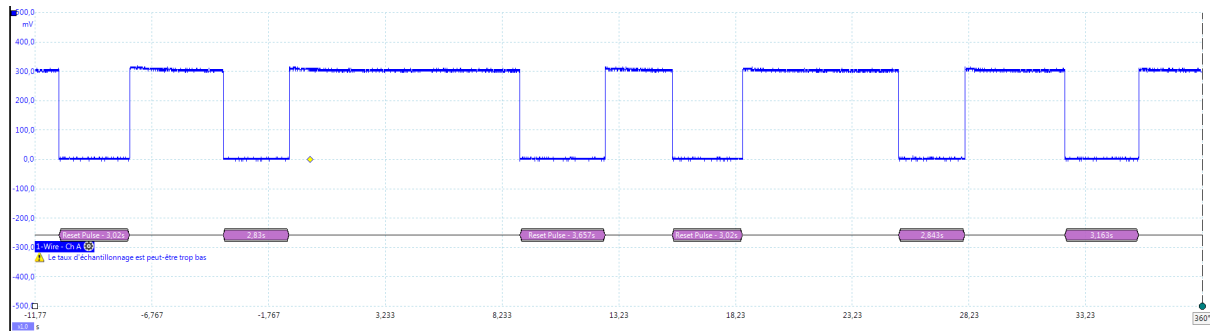
    if (HAL_GPIO_ReadPin(GPIOB, GPIO_PIN_5) && readValue < 3000) // PIN HIGH
    {
        strcpy((char*)buf, "MOTION\r\n");

        HAL_UART_Transmit(&huart2, buf, strlen((char*)buf), HAL_MAX_DELAY);
        // HAL_GPIO_WritePin (GPIOB, GPIO_PIN_15, 1); // LED ON
        HAL_GPIO_WritePin(GPIOA, LD2_Pin|LED_ROUGE_Pin, 1);
        HAL_Delay (1000);
        HAL_GPIO_WritePin(GPIOA, LD2_Pin|LED_ROUGE_Pin, 0);

        // HAL_GPIO_WritePin (GPIOB, GPIO_PIN_15, 0); // LED OFF
        // while (HAL_GPIO_ReadPin(GPIOB, GPIO_PIN_5)); // Wait for PB5 to go low
    }
}
```

4.5.Observation

On a ensuite essayer d'apercevoir le signal du PIR sensor ci dessous grâce au picoscope et à l'oscilloscope pour déterminer de quelle type de signal il s'agit pour ce capteur.



On peut clairement voir qu'il s'agit ici d'un signal numérique, donc lorsqu'il y a détection, le signal passe à "1" et redescend à "0" lorsque le PIR sensor ne détecte plus rien. C'est un signal "tout ou rien".

Le temps de réponse pour la détection est instantanée, et la détection est maintenue sur 3s environ.

Au niveau de la portée, nous l'avons réglé au maximum, c'est-à-dire de 3 mètres.

5.Conclusion

Après avoir testé chaque capteur indépendamment de l'autre, nous avons pu relier ensuite les deux pour les faire communiquer entre eux et pouvoir arriver au bout de notre idée de départ. Finalement, ce BE à été une réussite car nous avons pu réaliser jusqu'au bout le projet voulu et nous avons aussi pu nous familiariser beaucoup plus avec les protocoles de communications.