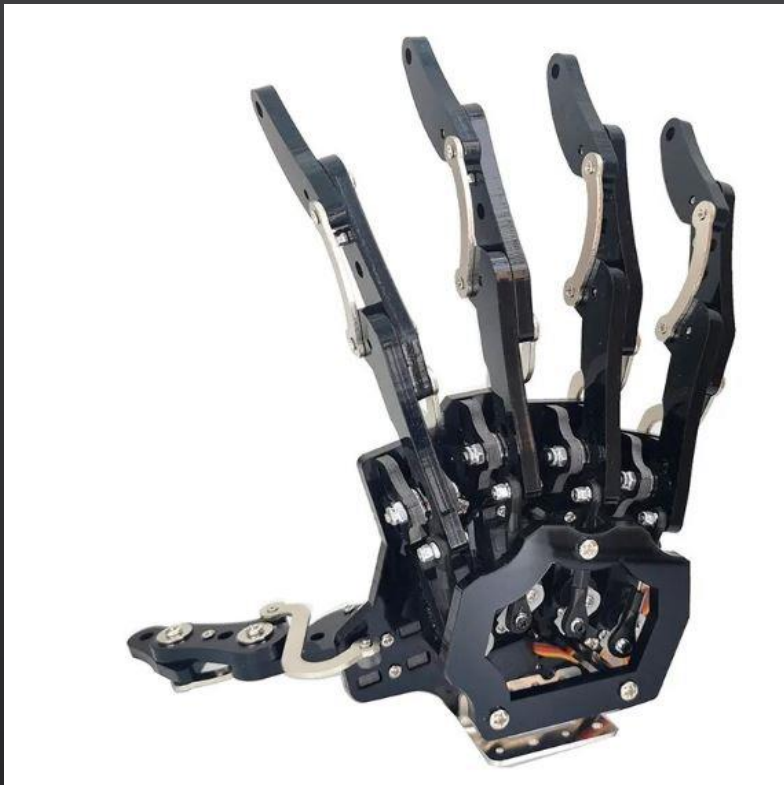
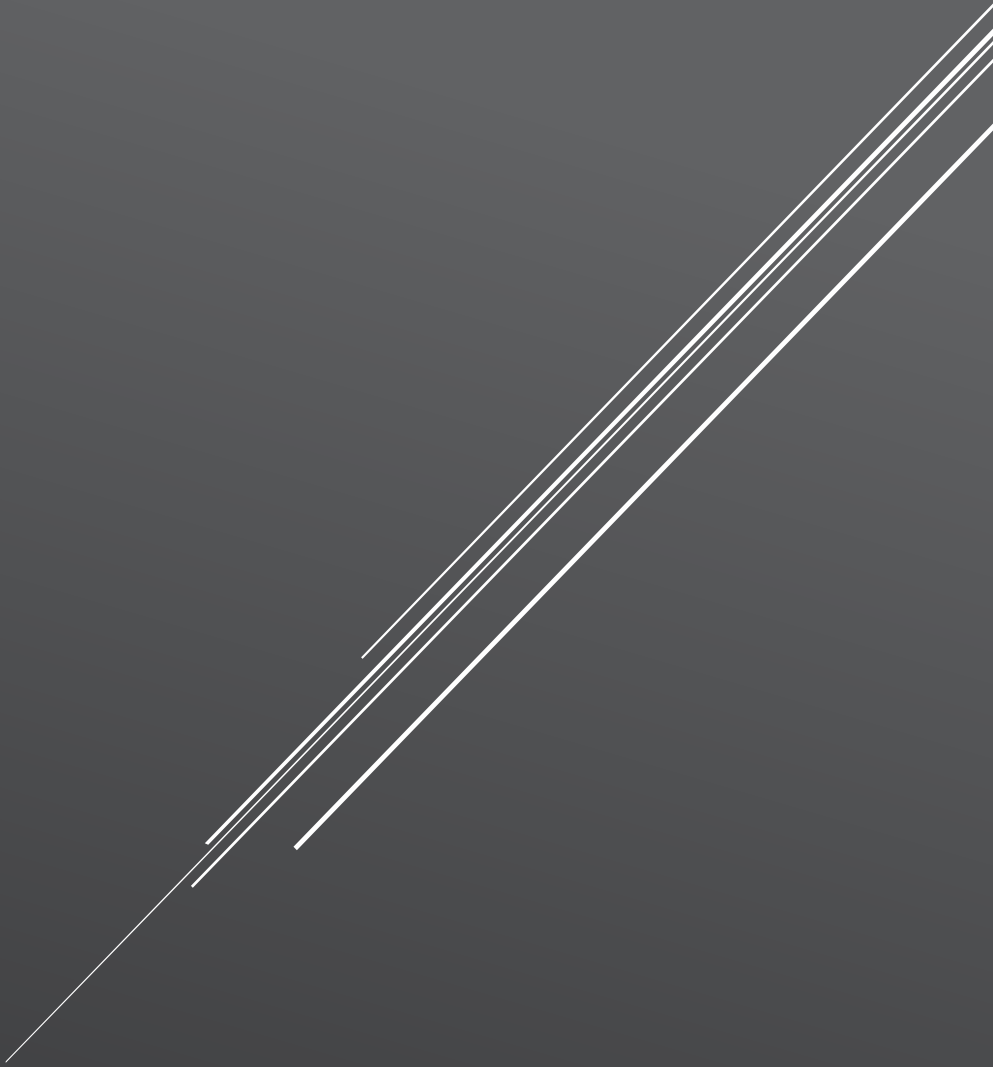


Daoud

Oulad El Fadel

6A

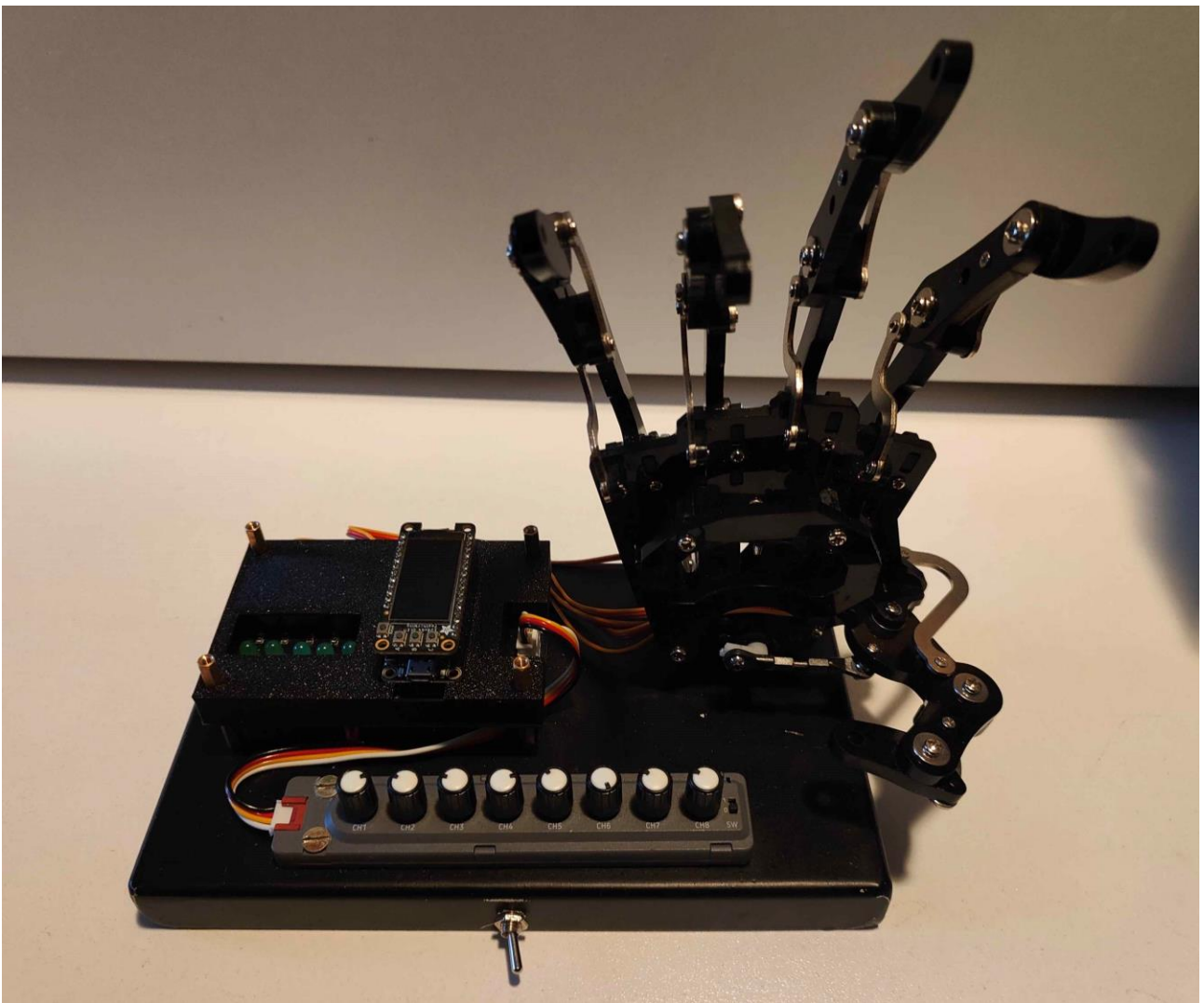
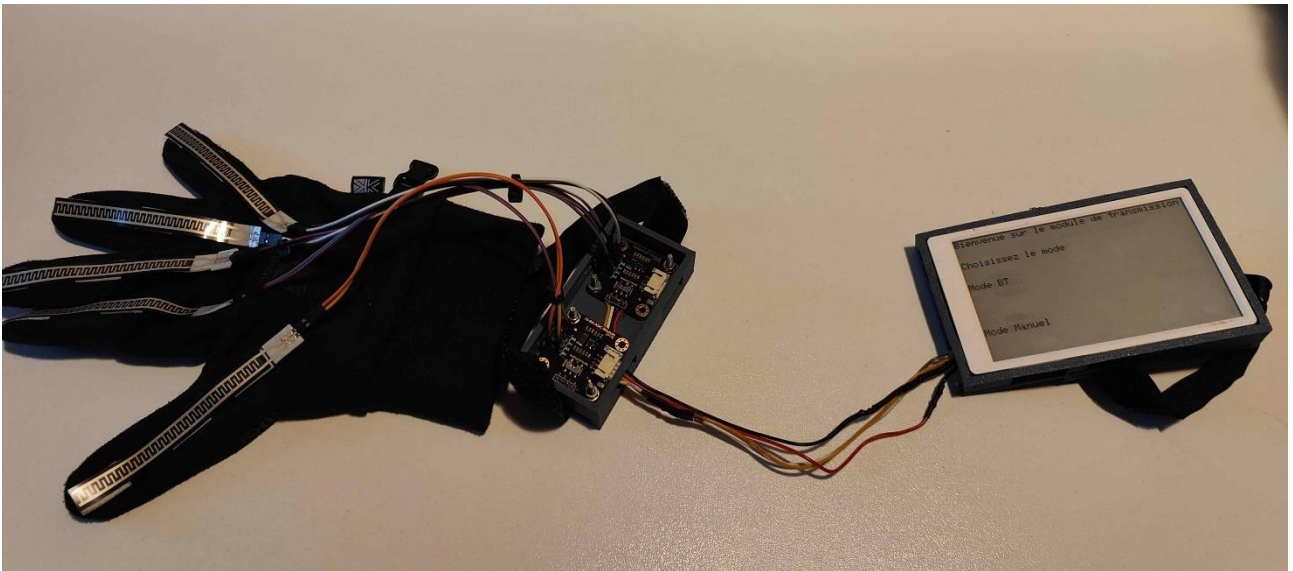


MAIN ROBOTISÉE

INRACI 2023-2024

« Si vous voulez trouver les secrets de
l'univers, pensez en termes d'énergie, de
fréquence, d'information et de vibration. »

— **Nikola Tesla**



Mes remerciements les plus sincères à mon promoteur monsieur Kapita pour sa formation et la mise à disposition de son temps et de son matériel pour mener à bien mon projet. Une personne inspirante qui a mon plus grand respect.

Mes remerciements à Thomas Giarrizzo qui a été un camarade exceptionnel et d'une grande aide. Je voudrais aussi remercier mes parents et Nathan et mes autres camarades pour le soutien qu'ils m'ont apporté.

Table des matières

1. L'introduction	6
2. Les caractéristiques.....	7
3. Les principes mis en jeu	9
Démodulation	9
Encodeur	14
4. Le schéma bloc.....	17
Gant	17
Main	18
5. L'étude détaillée et les mesures	19
Analyse de l'écran (M5 PAPER)	19
Module ADS1115	22
Capteur de flexion	23
I ² C.....	24
Mesures sur les servomoteurs	26
Mesure sur les capteurs de flexion	29
Mesure de l'interruption	31
6. La programmation	32
L'ordinogramme de la réception	32
L'ordinogramme de la transmission	33
L'ordinogramme de l'interruption.....	33
Programme de réception	34
Programme de transmission.....	34
7. La fabrication	35
8. La mise au point	36
9. La conclusion.....	37
10. Les annexes	38
Schéma de principe sans connecteur.....	38
Dessin de PCB avec les pistes	39
Dessin de PCB sans les pistes	40
11. Vue 3D.....	41
Assemblage du gant	41
Main robotisée	42
Moteur	43
Capteur de flexion	44
ESP32	45
Capot de la main	46

Gant	47
M5 PAPER	48
Module I2C	49
12. Fiche technique des composants	50
ESP32	50
M5 PAPER	51
Module ADC	52
Encodeur	53
13. Bibliographie et webographie	54

1. L'introduction

À l'occasion de mon TFE de fin de 6-ème année secondaire. J'ai décidé de mener un projet qui va me permettre d'appliquer mes connaissances acquises grâce à mon apprentissage à l'INRACI.

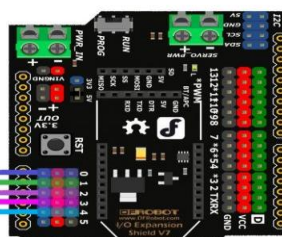
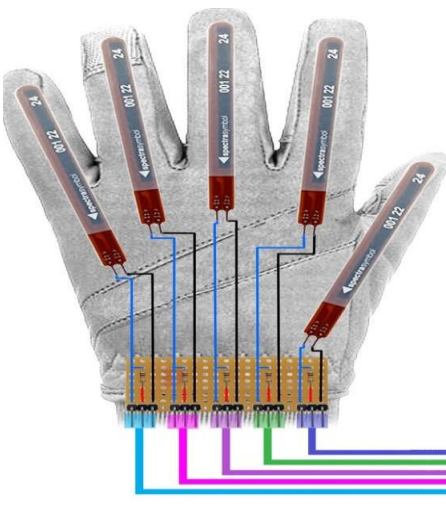
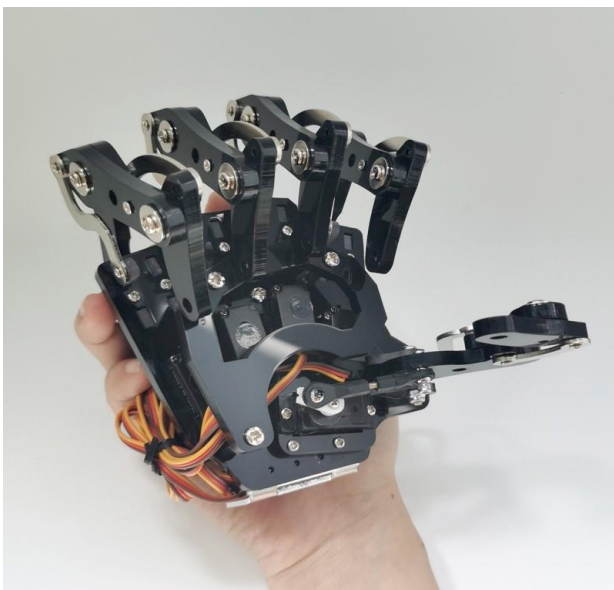
Le but de mon projet est de créer une main que l'on pourra contrôler grâce à un gant à distance.

Mes motivations pour ce projet viennent de ma passion pour la domotique et le contrôle à distance.

Depuis tout petit je suis fascinée par les entités contrôler à distance et ce projet est l'occasion de rallier ce que j'aime et l'électronique.

Rapide explication

Les servos moteurs qui seront reliés aux doigts de la main robotique vont être branchés à une première carte ESP32 qui communiquera avec la même carte. La deuxième carte sera elle branchée aux 5 capteurs de flexion qui seront collés aux gants. Les 2 cartes pourront communiquer grâce à leurs options BT intégrées.



2. Les caractéristiques

Caractéristiques :

Général :	Main	Gants
-2 boutons-poussoirs	Sur la carte ESP32	Sur la carte ESP32
-1 Led	LED sur l'Esp32	LED sur l'Esp32
-Main robotique avec 5 servomoteurs	Servomoteur SG90	X
-Gants avec 5 capteurs de flexions (x résolutions)	X	Capteur de flexion ZD10-100

Électronique :	Main	Gants
-BT 2,4 GHz	Sur la carte ESP32	Sur la carte ESP32
-Consommation	170 et 250 mA pour les 5 servomoteurs	ESP32 (80-180 mA)
-Batteries 5 volts	X	Inclus (dans la m5 Paper)

Mécanique :	Main	Gants
Poids	289g	5g
Matière	Acrylique noir 5.0	Tissu

Liste des composants :

Électronique

- 2X Adafruit Carte Feather HUZZAH32 ESP32
- 5x Capteur de Force, ZD10-100 500g
- 2X Batteries (a determiner)
- 1X M5PAPER

Mécanique

- 1X Bras mécanique
- 5X servomoteurs (SG90)
- 1X Gants en tissus

3. Les principes mis en jeu

Démodulation

Description détaillée

Notre projet comporte 2 parties (le gant avec une M5 Paper et la main robotique avec une ESP32) qui vont se transmettre des informations via une connexion Bluetooth. Nos 2 dispositifs de communication sont dotés de ce type de transmission. Cette technologie de transmission de données nous permet d'échanger des données entre la M5 Paper et l'ESP32. Elle nous permet aussi d'avoir une communication sécurisée grâce au chiffrement de données et d'avoir un temps de connexion rapide.

Caractéristiques détaillées

1 Portée de communication

La portée de communication est de 10 mètres entre nos 2 systèmes.

2 Fréquence de communication

La fréquence utilisée est de 2,4 GHz.

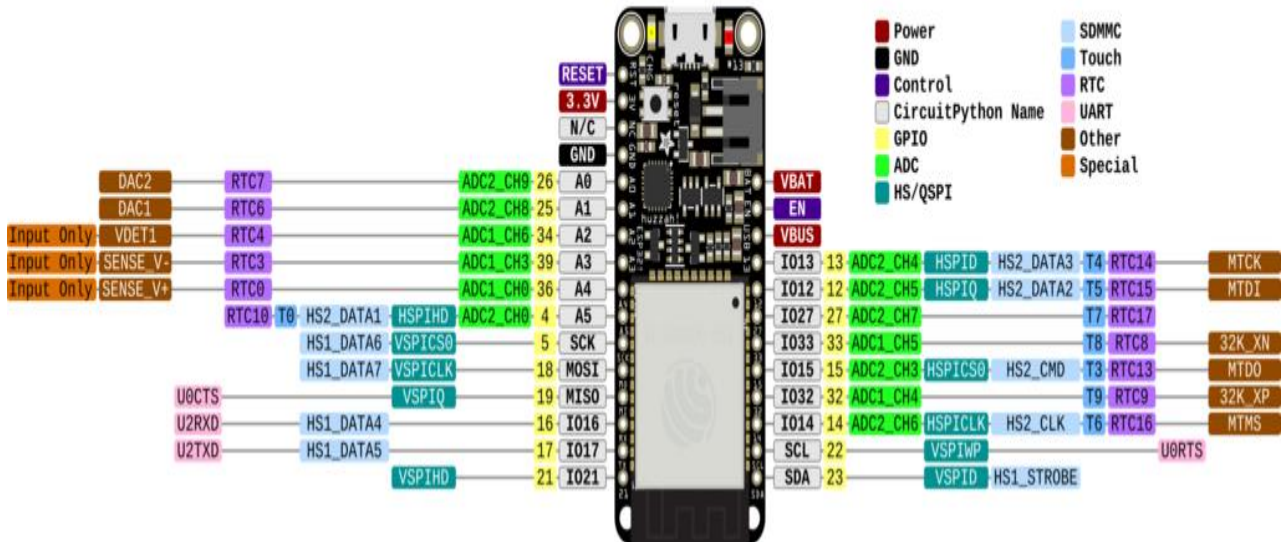
3 Maître multiple

Il peut avoir plusieurs esclaves contrôlés par 1 seul maître.

Brochure :

Adafruit HUZZAH32 ESP32 Feather

<http://www.adafruit.com/products/3405>



M5 PAPER

Do not expose the device in ultraviolet rays for a long time

TOUCH SCREEN

4.7" 960x540 16-Level EInk

Anti-glare Tempered Glass

USB PWR IN

BATTERY

RST

T&H STH30

ON G38

MICRO SD

PORT.A

PORT.B

PORT.C

BUILT-IN MAGNET

Li-Po 1150mAh 4.35V

M5STACK

ESP32 D0W/D06-V3 16M-FLASH 8M-PSRAM

T&H@SHT30

RTC@BM8563

CAP.TOUCH@GT911

4.7" 960x540 16-Level

BATTERY:LiPo-1150mAh

LOW POWER CONSUMPTION

UIFlow

Arduino

MicroPython

118x67x10mm @ 86g



Étude théorique du principe physique de la réception (démodulation) Bluetooth.

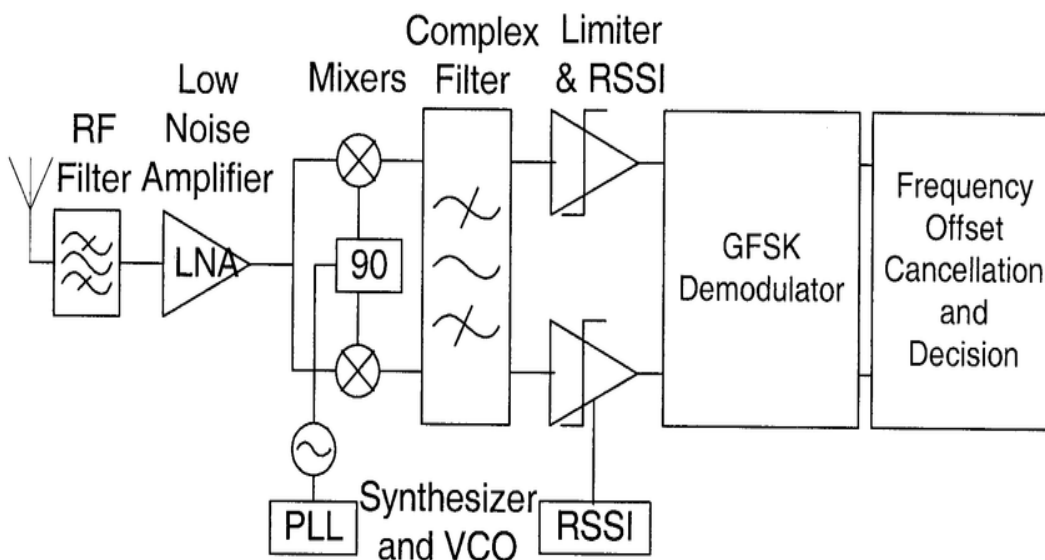
Le principe de modulation utilisé est le GFSK qui est une technique de variation de fréquences qui est utilisée pour transmettre des données.

L'émetteur (M5 Paper) va envoyer les données moduler en GFSK via une radiofréquence vers l'esp32.

L'ESP32 grâce à une antenne va capter l'onde et acheminer le signal vers le circuit de réception.

Avant d'arriver à l'étape de démodulation le signal va être filtré, le bruit capté va être éliminé et le signal utile va être renforcé.

Le circuit de réception va démoduler le signal reçu en traduisant les variations de fréquence pour récupérer les données originales.



Programmes de test

```
char receiveData[7] //Tableau pour stocker les données reçues
```

```
// Initialisation de la communication série Bluetooth  
SerialBT.begin(device_name);  
// Définition du code PIN pour le couplage Bluetooth  
SerialBT.setPin(pin);  
Serial.println("Using PIN");
```

```
if (SerialBT.available()) {  
    SerialBT.write(Serial.read());  
    DATA = SerialBT.read(); // Lire le caractère envoyé via Bluetooth  
}
```

```
// Affiche les données reçues  
Serial.println(receiveData[0]);  
Serial.printf("a:%d\n", receiveData[1]);  
Serial.printf("b:%d\n", receiveData[2]);  
Serial.printf("c:%d\n", receiveData[3]);  
Serial.printf("d:%d\n", receiveData[4]);  
Serial.printf("e:%d\n", receiveData[5]);  
Serial.printf("f:%d\n", receiveData[6]);
```

Librairie utiliser

"BluetoothSerial.h" version 1.0.1.

Auteur : Henri Abrahamsen

Site consulté le 31/01/2024 à 21 :43



"ESP32Servo.h" version 1.1.2

Auteur: Kevin Harrington, John K. Bennett

Site consulté le 14/02/2024 à 21 :26



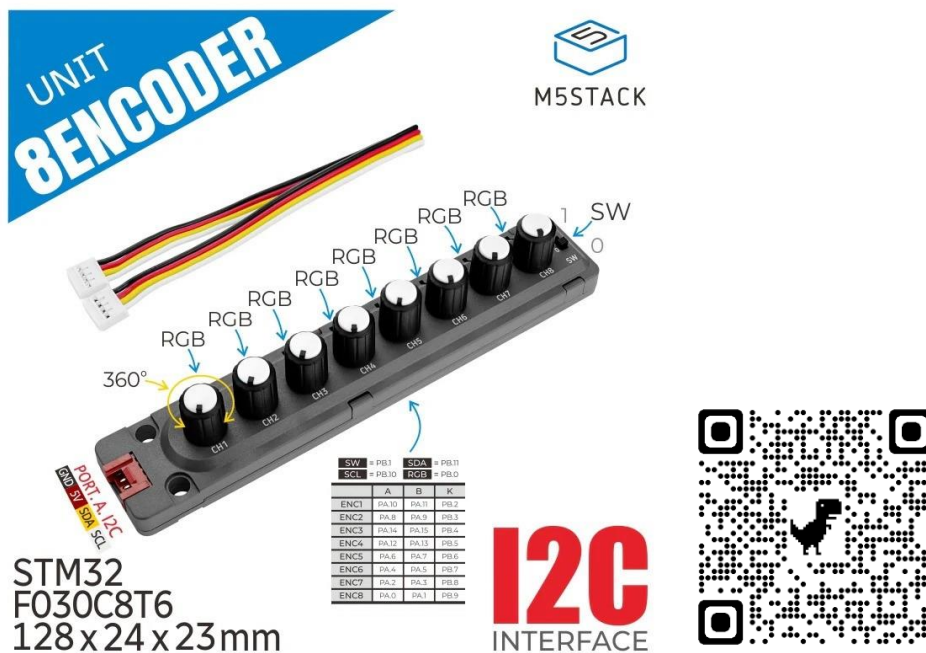
[Voici le lien pour en savoir plus sur le code](#)



Encodeur

Introduction

L'un des sous-ensembles de notre projet est le 8-encoder Unit qui nous permet de contrôler les moteurs qui contrôlent les doigts de la main. Les encodeurs utilisés pour notre projet sont les EC11 qui sont des encodeurs incrémentaux. La position des encodeurs est communiquée grâce à microcontrôleur en I2C vers notre ESP32.



Fonctionnement d'un encodeur incrémental

Définition :

Un encodeur incrémental est un capteur qui suit le mouvement d'un objet en incrémentant ou décrémentant sa position par rapport à une position de référence.

L'encodeur incrémental est constitué de 2 parties : le stator et le rotor.

Le rotor est la partie tournante qui est relié l'entité dont on souhaite mesurer la position, la distance ou la vitesse.

Le stator est la partie fixe sur laquelle se trouve le capteur qui va nous permettre de mesurer les données que l'on veut (vitesse, position, etc.).

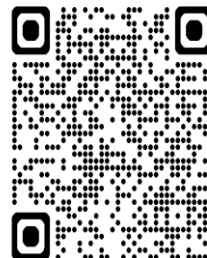
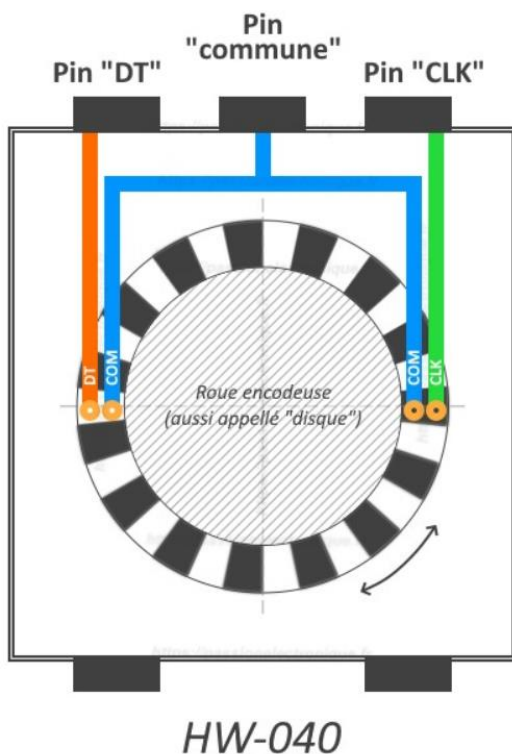
Fonctionnement :

Pin « DT » : Son rôle est de fournir un signal de données qui changent selon la position des contacts.

Pin « CLK » : Son rôle est de fournir un signal d'horloge qui change selon la position des contacts.

Pin « commune » : Elle est branchée au GND.

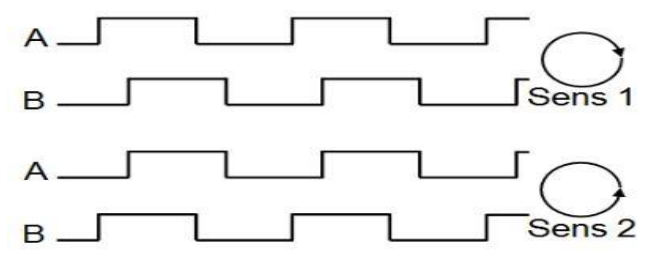
L'encodeur est composé d'un disque qui comporte des zones blanches conductrices et des zones noires non conductrices. Lorsque l'on va tourner le disque, l'un des 2 groupes de pin va passer sur une zone conductrice ce qui va créer un court-circuit.

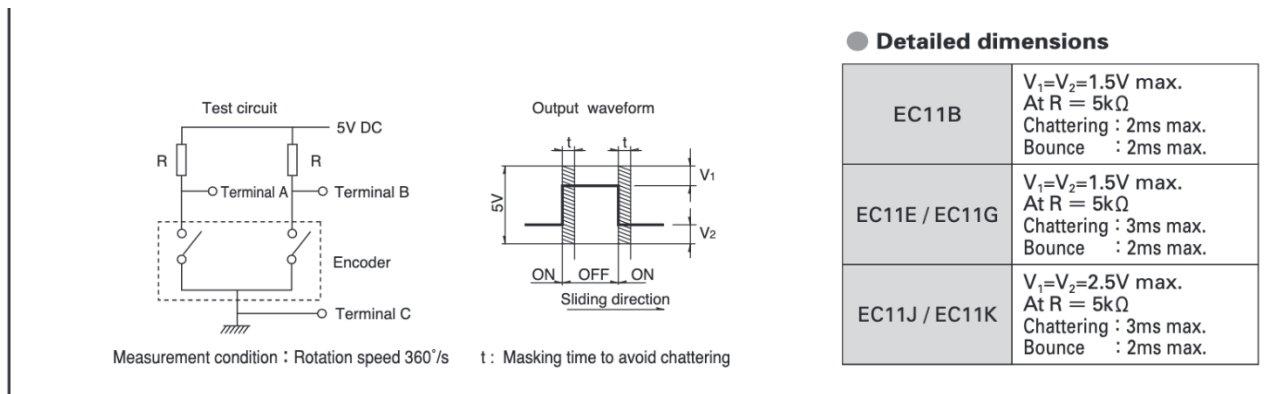


Traitements des impulsions électriques

Pour obtenir l'information souhaitée, il faut traiter les impulsions électriques générées par le capteur. Le système chargé d'effectuer le traitement des impulsions est un compteur qui va compter les impulsions et incrémenter ou décrémenter selon le déplacement de l'objet.

Le codeur va délivrer 2 signaux carrés (A et B) grâce à des contacts mécanique qui vont alterner entre l'état ON ou OFF ce qui va créer un déphasage qui va nous permettre de déterminer la position de l'objet et le sens de rotation. Chaque configuration des interrupteurs correspond à une position.





Les signaux A et B se comporte comme des interrupteur ouvert (ON) ou fermer (OFF) ce qui crée 2 choses :

Chattering : Variation indésirable dans les signaux.

Bounce : Rebondissement des « interrupteurs » mécanique après une ouverture ou une fermeture ce qui cause des variations indésirables dans le signal.

C'est 2 phénomènes nous montre qu'il y a une fréquence limite à ne pas dépasser pour avoir un signal lisible.

Programme de test :

```
void index() {
  encoderValue1 = MM.getAbsCounter(1); // Lecture de la position de l'encodeur
  mappedValue1 = map(encoderValue1, 0, 100, 50, 100);

  // Vérification des limites de l'encodeur
  if (encoderValue1 <= 0) {
    encoderValue1 = 0;
    MM.setAbsCounter(1, 0); // Réinitialisez l'encodeur à 0
  } else if (encoderValue1 >= 220) {
    encoderValue1 = 220; // Forcer encoderValue à 350 si c'est le cas
    MM.setAbsCounter(1, 220); // Réinitialisez l'encodeur à 0
  }
}
```

LIBRAIRIE : [M5ROTATE8](#)

Version : 0.3.0

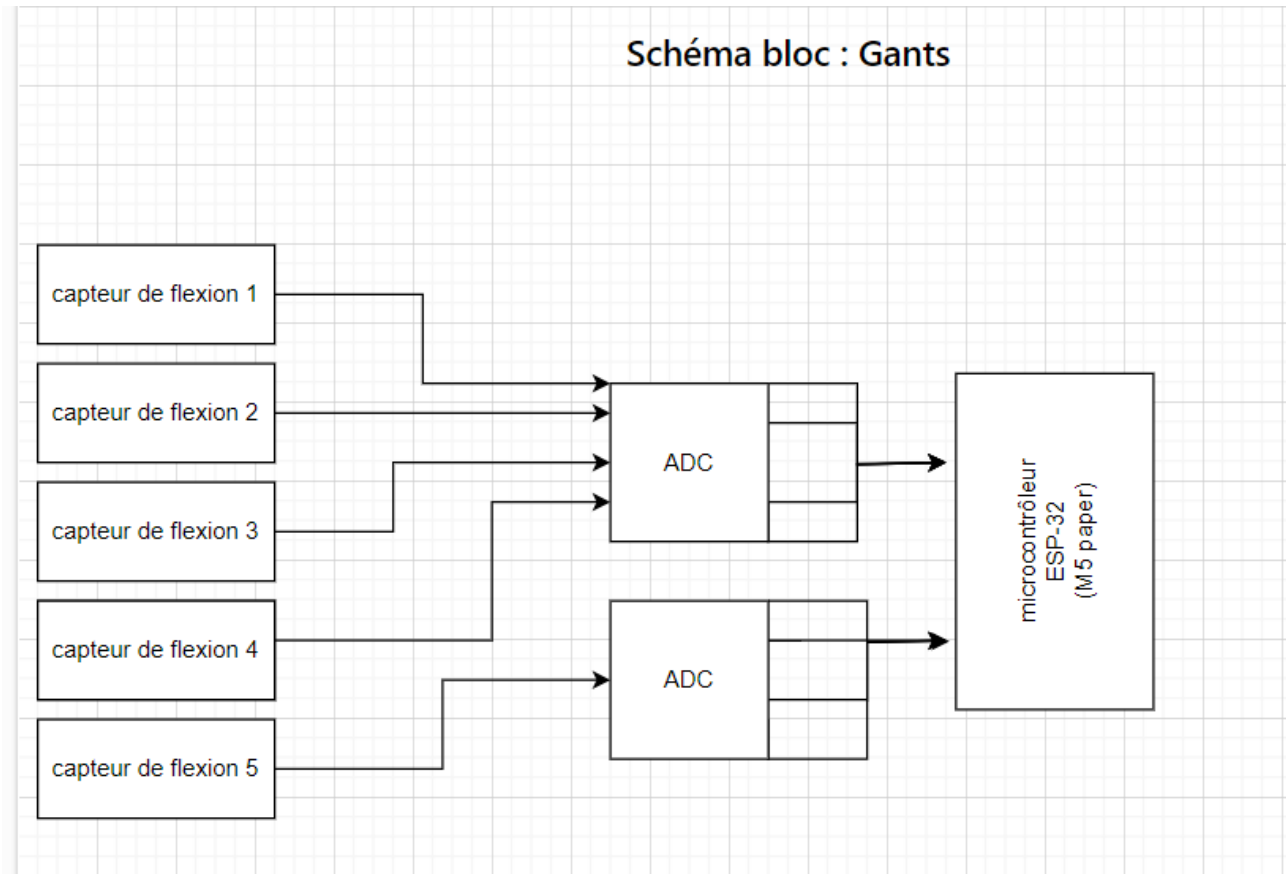


[Voici le lien pour en savoir plus sur le code](#)



4.Le schéma bloc

Gant



Gants :

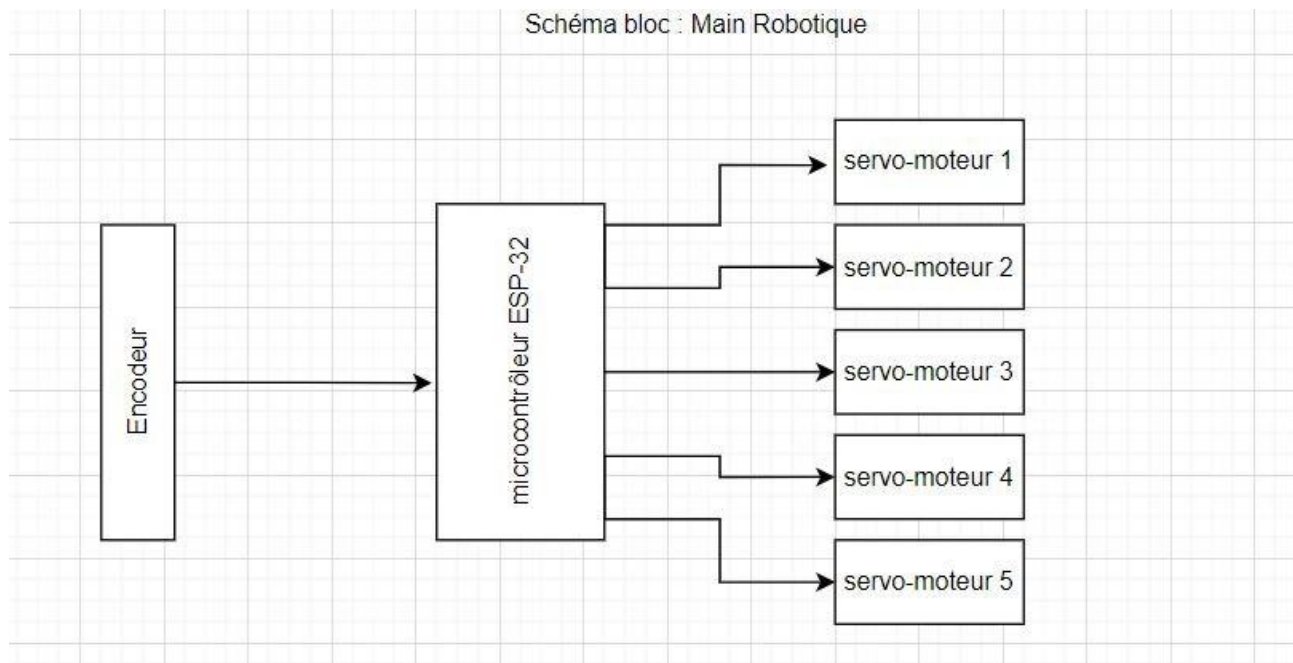
Capteur de flexion et module ADC :

Les capteurs de flexion vont transmettre le mouvement à la ESP32 (M5 Paper) grâce à deux modules I2c. 2 capteurs de flexion dans l'un des modules et 3 dans l'autre. Les modules vont permettre de relier les capteurs de flexions au microprocesseur en réduisant le nombre de connexions.

M5 Paper :

L'ESP32(M5 Paper) va envoyer les instructions à l'ESP32 de la main robotisée grâce au module BT intégré. L'écran tactile intègre va afficher la position des doigts et le menu pour pouvoir choisir les différents modes de fonctionnement.

Main



Main robotisée :

ESP32 et servomoteur :

Le microprocesseur de la main robotique va recevoir les instructions grâce au module BT intégré qui va ensuite les appliquer sur les ou le servomoteur concerner. L'ESP-32 va renvoyer la position des doigts vers la M5 Paper afin de l'afficher.

Encodeur :

Il est composé de 8 encodeurs mécaniques qui vont nous permettre de faire bouger "manuellement" nos doigts. L'encodeur est branché à l'ESP-32 qui va contrôler les moteurs selon la position des encodeurs attribués à chaque doigt.

5.L'étude détaillée et les mesures

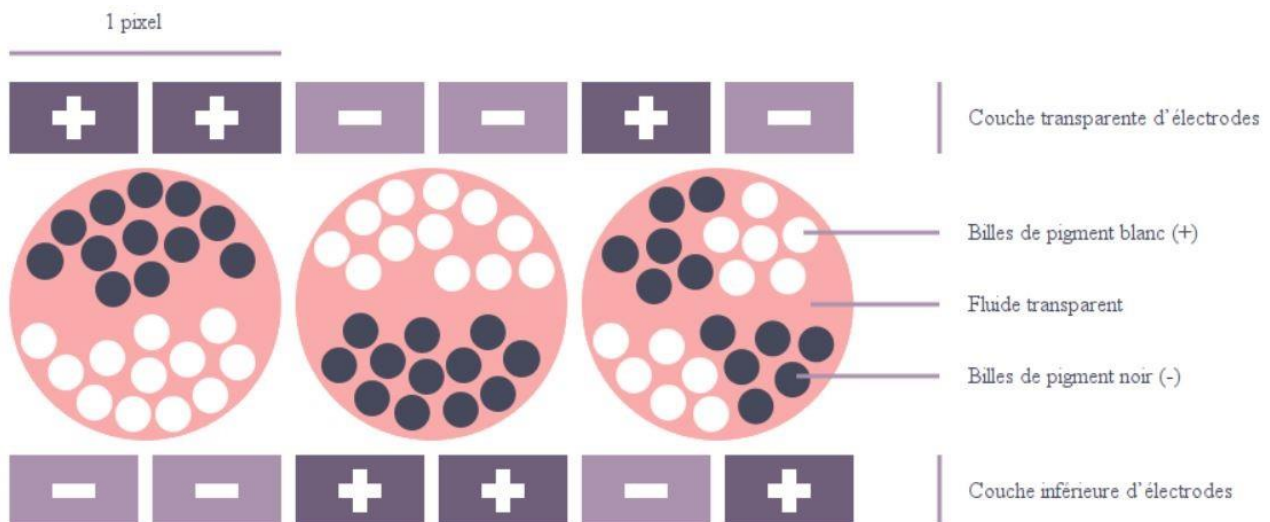
Analyse de l'écran (M5 PAPER)

Introduction

Notre projet se constitue de la M5 Paper qui va nous permettre de communiquer la position des capteurs à notre main robotique. La M5 Paper est aussi composée d'un écran le EPD_ED047TC1 qui va nous permettre d'afficher les différents modes (contrôle avec le gant, mouvement avec l'écran et le mode musique) et de les sélectionner grâce au tactile de l'écran. Il va aussi nous permettre de contrôler la main robotique en appuyant directement sur le choix de la position des doigts.

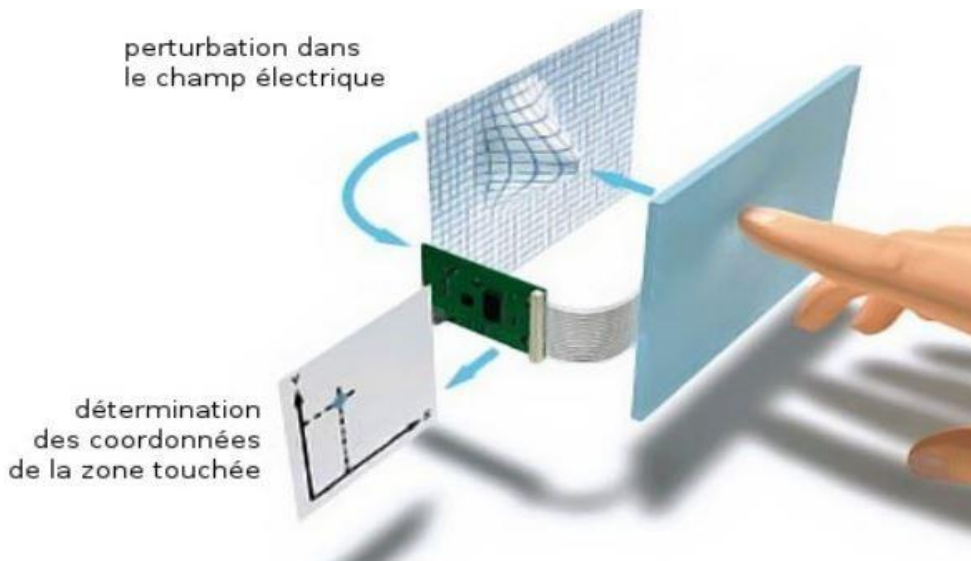
Comment marche l'affichage ?

L'écran utilise la technologie E-Ink qui permet un affichage à très faible consommation et n'a pas besoin d'énergie pour laisser une image afficher, car il ne consomme que lorsque l'image est modifiée. L'écran est composé de pixels dans lesquelles il y a de l'encre composée de billes de pigments chargées positivement pour créer un pigment blanc et négativement pour créer un pigment noir qui baignent dans un fluide transparent. Lorsque l'on va vouloir afficher la couleur noire, nous allons envoyer une tension électrique négative pour attirer les billes de pigments blancs et repousser de l'autre côté du fluide, une tension positive pour attirer les billes de pigments noirs et repousser les charges négatives de l'autre côté du fluide. Sans présence de tension, les pigments restent à leur place.



Comment fonctionne le tactile ?

L'écran va être traversé par un champ électrique faible qui quadrille l'écran. Le doigt est un conducteur électrique et lorsque l'on va appuyer sur une partie de l'écran le champ électrique va être perturbé et le processeur va pouvoir sur quelle partie du quadrillage le champ a été perturber.



Code de "teste":

```
if (M5.TP.available()) {
    if (!M5.TP.isFingerUp()) {
        M5.TP.update();
        canvas.fillCanvas(0);
        bool is_update = false;
        for (int i = 0; i < 2; i++) {
            tp_finger_t FingerItem = M5.TP.readFinger(i);
            if ((point[i][0] != FingerItem.x) ||
                (point[i][1] != FingerItem.y)) {
                is_update = true;
                point[i][0] = FingerItem.x;
                point[i][1] = FingerItem.y;
                canvas.fillRect(FingerItem.x - 50, FingerItem.y - 50, 00, 100, 15);
                Serial.printf("Finger ID:%d-->X: %d*C Y: %d Size: %d\r\n",
                    FingerItem.id, FingerItem.x, FingerItem.y,
                    FingerItem.size);
            }
        }
        if (is_update) {
            canvas.pushCanvas(0, 0, UPDATE_MODE_DU4);
        }
    }
}
```

[Voici le lien pour en savoir plus sur le code](#)



Caractéristiques de l'écran :

Taille diagonale : 4,7"

Zone active : 58,32 (L) × 103,68 (H) mm

Type d'interface : Données parallèles (1 canal, 8 bits), connecteur 44 broches

Résolution : 540 × 960, qHD 235 PPI

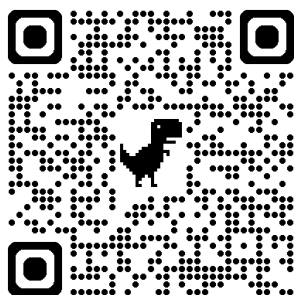
Fréquence : 85Hz

Tension d'entrée : 3,3 V (typique)

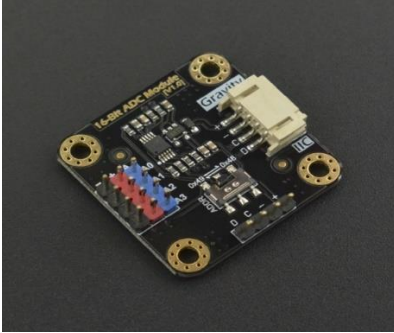
Environnement : Température de fonctionnement : 0 ~ 50 °C ; Température de stockage. : -25 ~ 70 °C

Niveau de gris : 16

Datasheet de l'écran



Module ADS1115



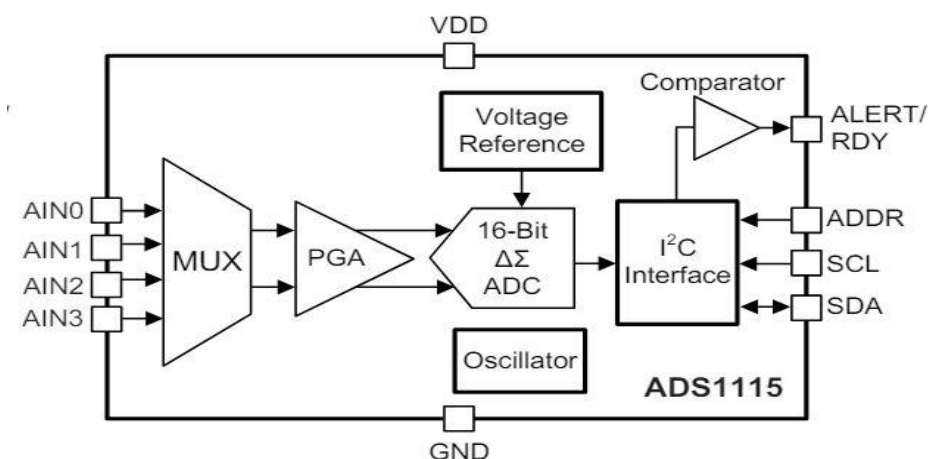
Introduction

Pour transmettre à notre MCU le signal analogique qui correspond à la position du doigt acquise grâce à nos capteurs de flexion nous avons besoin de d'abord convertir le signal analogique en signal numérique. Ce rôle sera rempli grâce à nos 2 modules ADC.

Qu'est-ce qu'un ADC

C'est un périphérique qui convertit un signal analogique en signal numérique.

Schéma interne du module ADS1115



AIN correspond à l'entrée analogique de notre convertisseur.

Le MUX correspond à un multiplexeur qui va sectionner l'entrée que nous voulons lire.

Le signal va ensuite passer par le PGA (Programmable Gain Amplifier) qui va amplifier le signal avec un Gain que l'on peut programmer.

Ensuite le signal est traité par le convertisseur delta-sigma de 16-bits pour récupérer un mot binaire. 16 bits représentent les différentes tensions disponibles (65536).

L'interface I²c permet de communiquer nos valeurs numériques au microcontrôleur qui les traite.

Le comparateur va comparer la tension d'entrée du bloc interface I²c à une tension de référence et va créer une alerte quand la tension de référence est dépassée.

L'oscillateur permet de synchroniser les différents blocs entre eux grâce à un signal d'horloge.

Capteur de flexion



Introduction

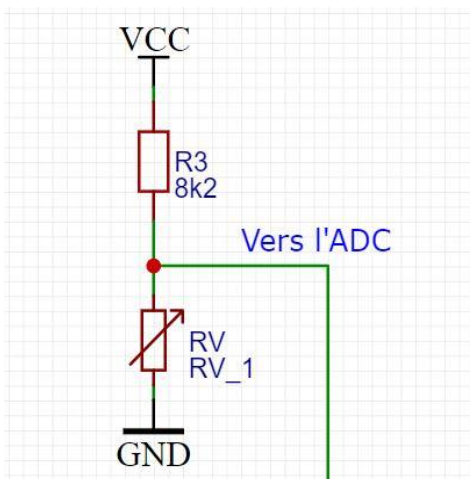
Pour notre projet nous avons besoin de connaître la position des doigts de l'utilisateur et pour cela nous allons utiliser des capteurs de flexions.

Qu'est-ce qu'un capteur de flexion

Le capteur de flexion est une bande dont la résistance va varier selon la flexion que l'on exerce avec.

Le capteur est constitué d'encre polymère composée de particules conductrices qui vont s'écarter lorsque l'on plie le capteur ce qui va augmenter la résistance. Selon le degré de flexion, la résistance va varier.

Le capteur doit être utilisé avec une résistance (8k2Ω dans notre cas) pour créer un pont diviseur de tension qui va lui permettre d'être lu par un convertisseur analogique /numérique.



Caractéristiques :

Modèle : ZD10-100

Tension : 3,3 V CC

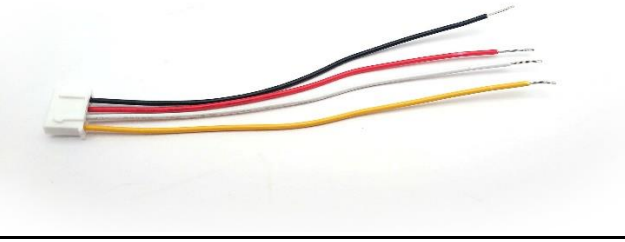
Durabilité : plus de 1 million de fois

Temps de réponse : moins de 10 ms

Le capteur de flexion a besoin d'un pont diviseur de tension pour pouvoir être lu par l'ADC.

Formule : $\frac{R_v}{R_v + R_3} \cdot V_{CC} = U_{adc}$

I²C

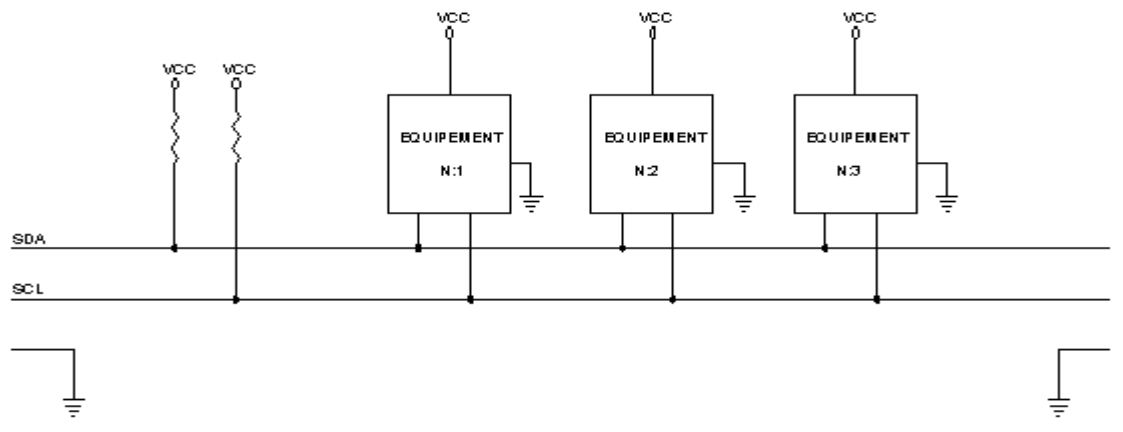


Introduction

Dans notre projet nous avons besoin de faire communiquer nos modules entre eux et pour cela nous avons besoin d'un bus de communication qui dans notre cas est le bus I²C.

qu'est-ce qu'un bus I²C

L'I²C est un bus de communication développé par PHILIPS dans les années 80 et qui avait pour but de relier plusieurs périphériques à un seul microprocesseur.



L'I²C est composée de 3 fils : SDA, SCL, Masse.

Il est half duplex donc on peut lire et écrire, mais pas en même temps.

Le SDA représente le signal de la data qui correspond à nos données.

Le SCL représente le signal d'horloge qui synchronise les appareils entre eux.

Pour communiquer, les périphériques doivent avoir une adresse. Dans ce cas nous avons la transmission d'une adresse.

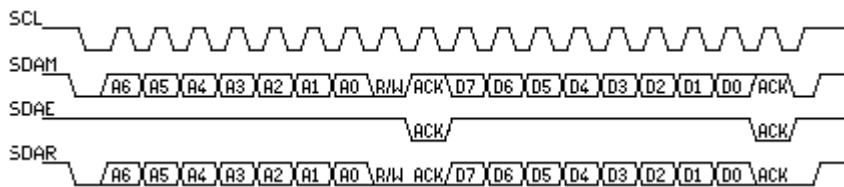


Les bits d'A6 à A0 contiennent l'adresse. L'adresse est unique pour chaque appareil.

Le bit R / W indique si le maitre s'apprête à lire ou écrire.

Un bit ACK (Acknowledge) doit être envoyé par le périphérique qui a reçu l'adresse pour confirmer au maitre qu'il l'a bien reçu.

Écriture



SCL correspond au signal d'horloge.

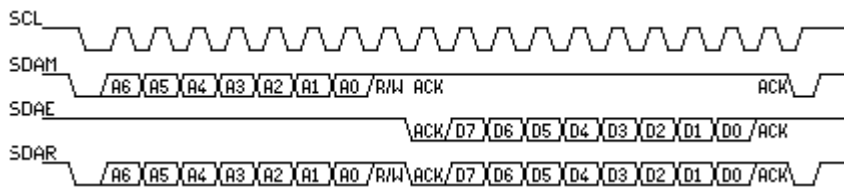
SDAM correspond au signal envoyer par le maitre.

SDAE correspond au signal envoyer par l'esclave.

SDAR correspond aux 2 signaux.

Ici nous pouvons voir l'adresse de A6 à A0 avec le R/W et ACK ensuite nous avons les donner de D7 à D0 et un ACK à la fin que l'esclave doit envoyer pour confirmer la réception.

Lecture d'une donnée par le maitre



Après avoir lu la donnée, le maitre va mettre ACK à 0 si il veut lire les autres données.

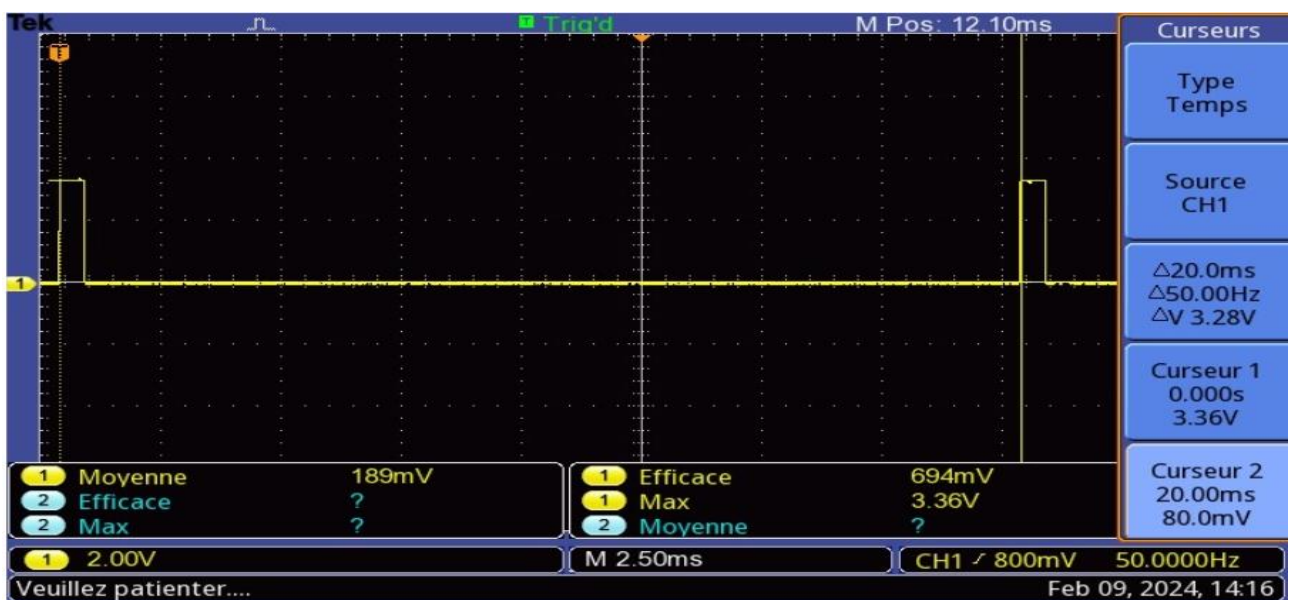
Mesures sur les servomoteurs

	T (ms)	F (Hz)	Ton (ms)	Rapport cyclique r ((Ton /T). 100))
0°	20ms	50Hz	528us	2,64
90°	20ms	50Hz	1,46ms	7,3
180°	20ms	50Hz	2,36ms	11,8

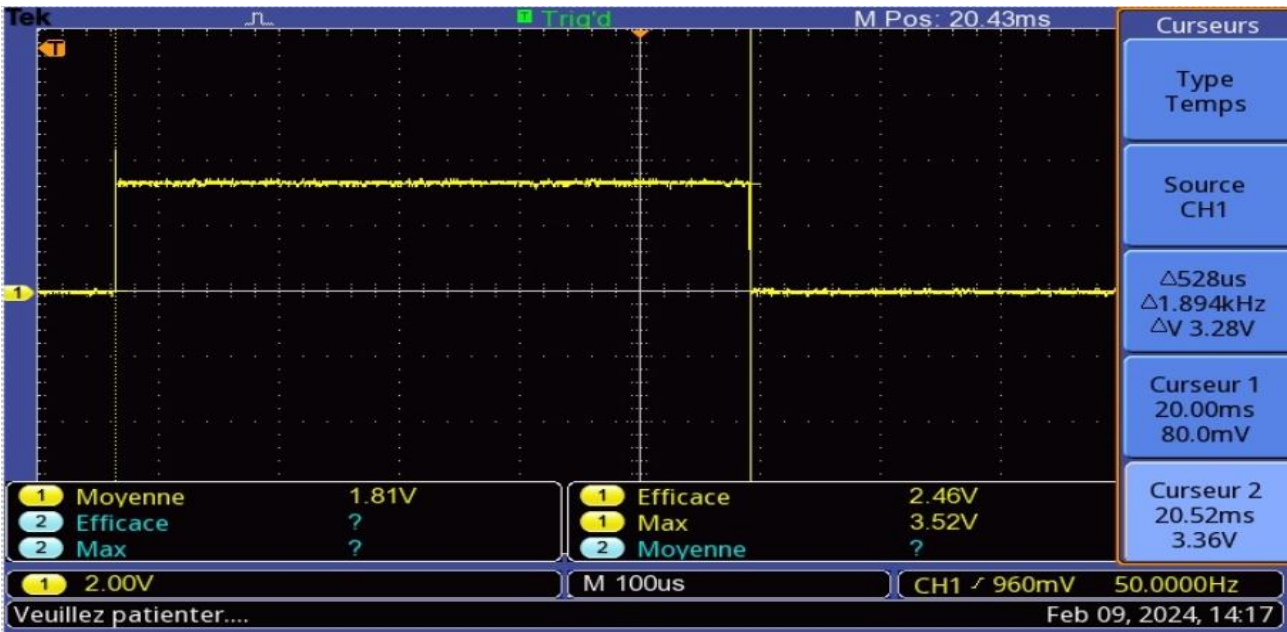
On peut remarquer que plus l'angle est grand plus le rapport cyclique augmente.

Moteur à 0°

Période

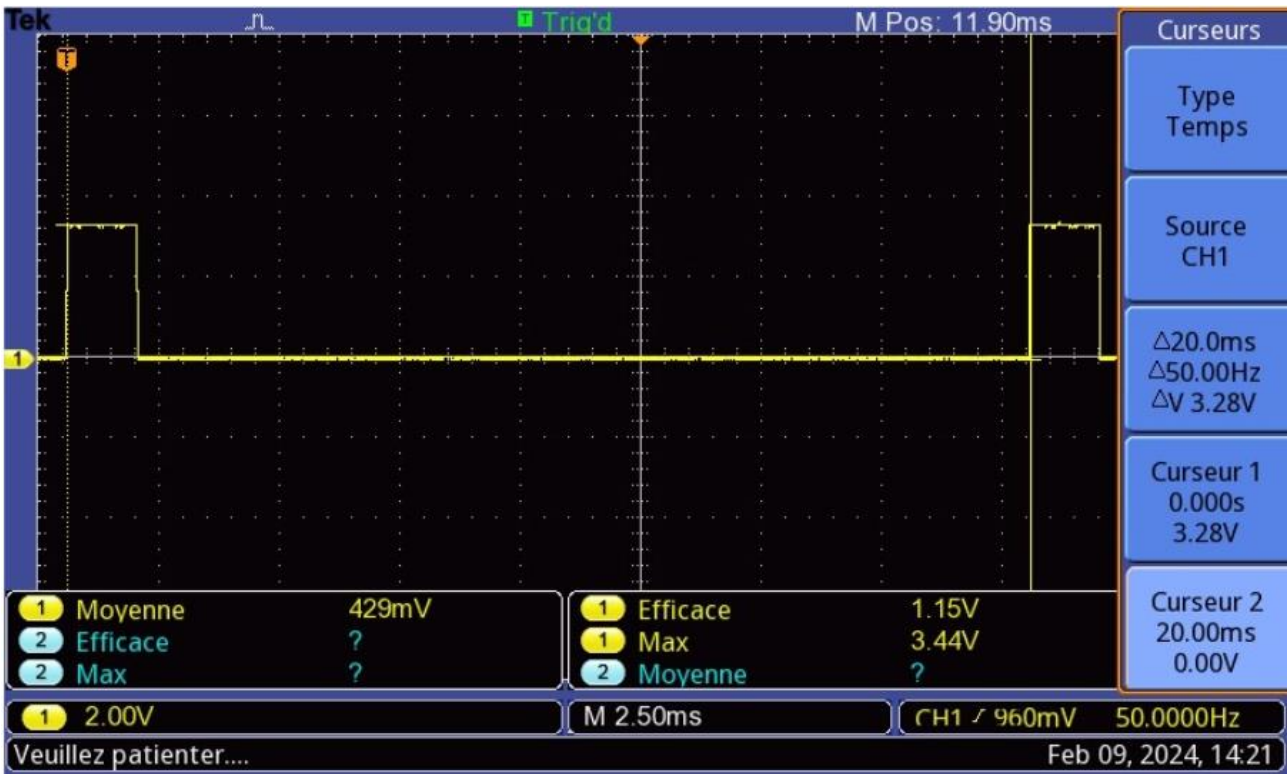


Ton

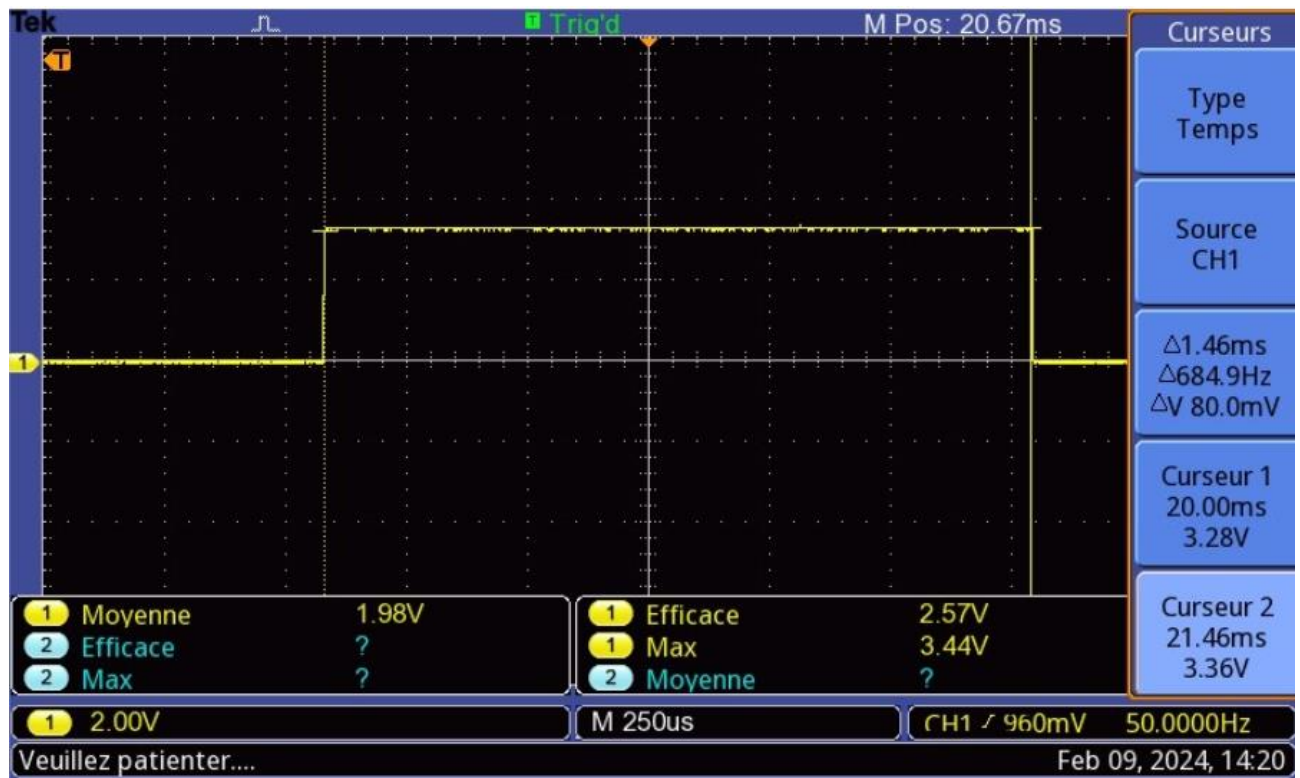


Moteur à 90°

Période

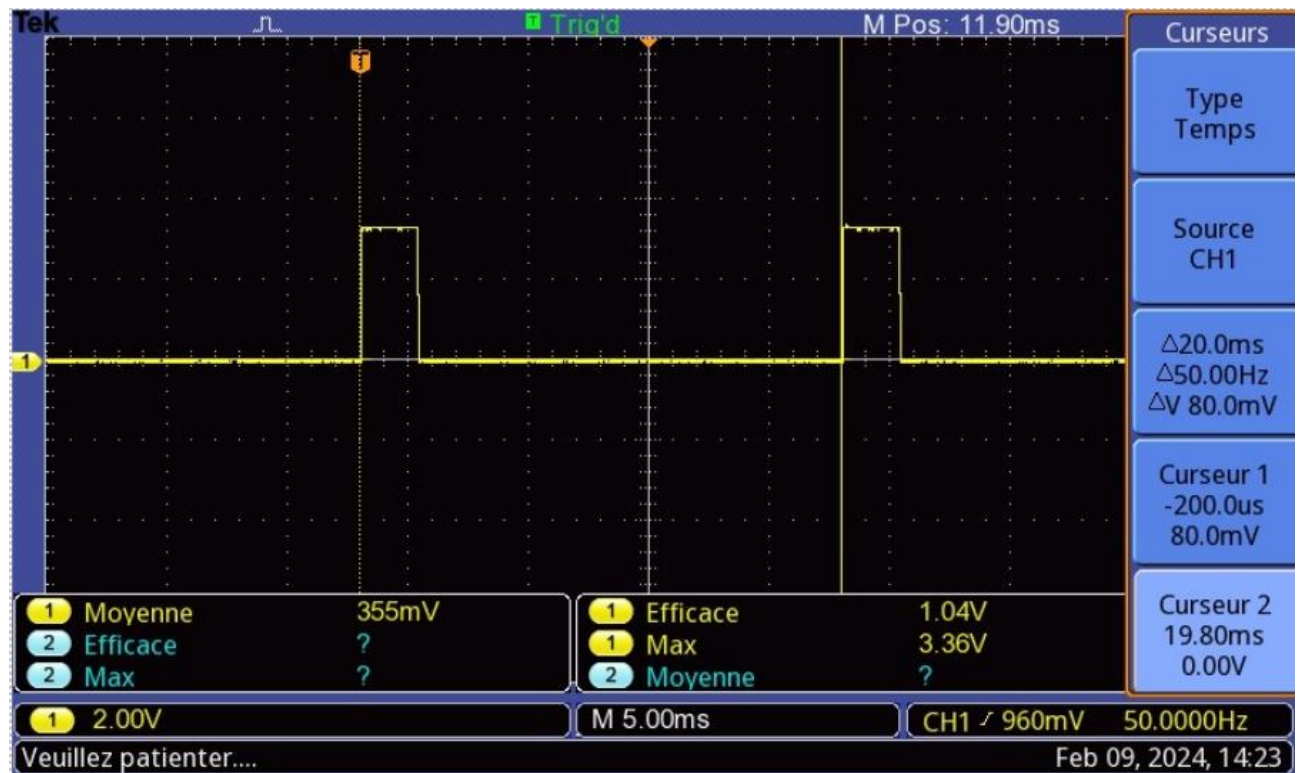


Ton

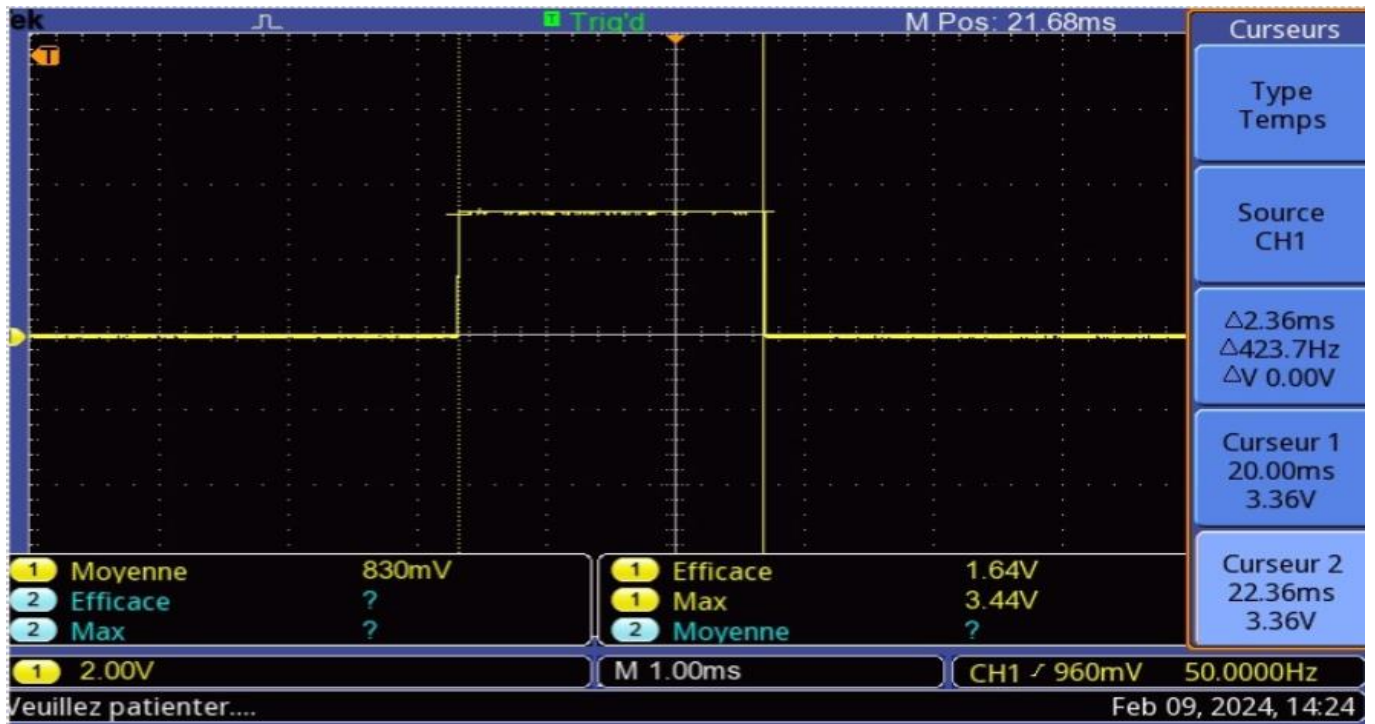


Moteur à 180°

Période



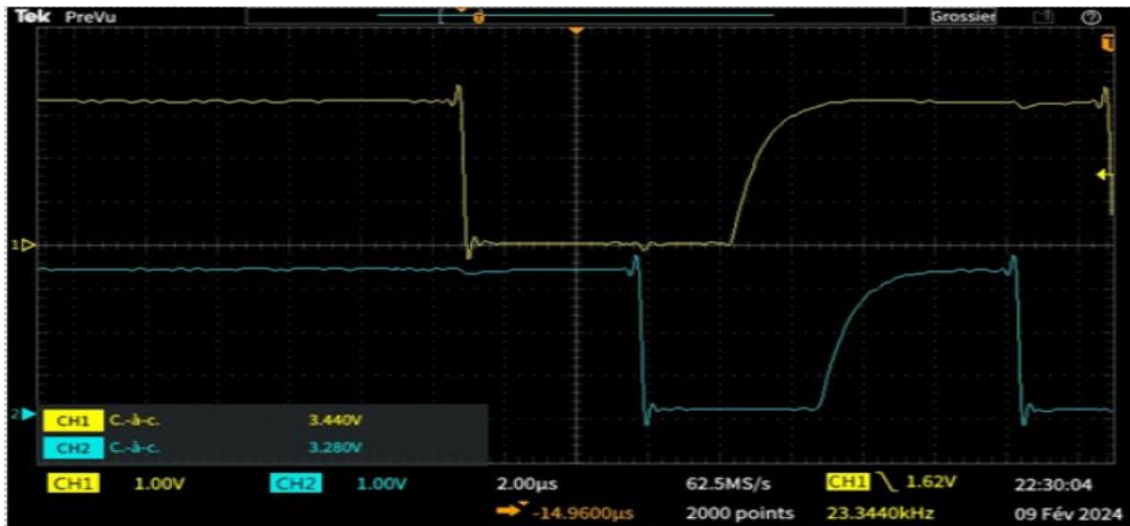
Ton



Mesure sur les capteurs de flexion

	Mesure (volt mètre)	Mesure (Ohm mètre)	Calcul du programme (ADC25ADC25)	Valeur en binaire
Sans flexion	2,267V	18k Ω	245	6143 0b 0001 0111 1111 1111
Flexion a 90°	2, 3V	7k3 Ω	131	4262 0b 0001 0000 1010 0110
Flexion totale	722mV	2k3 Ω	83	489 0b 0001 1110 1001

Bit de start



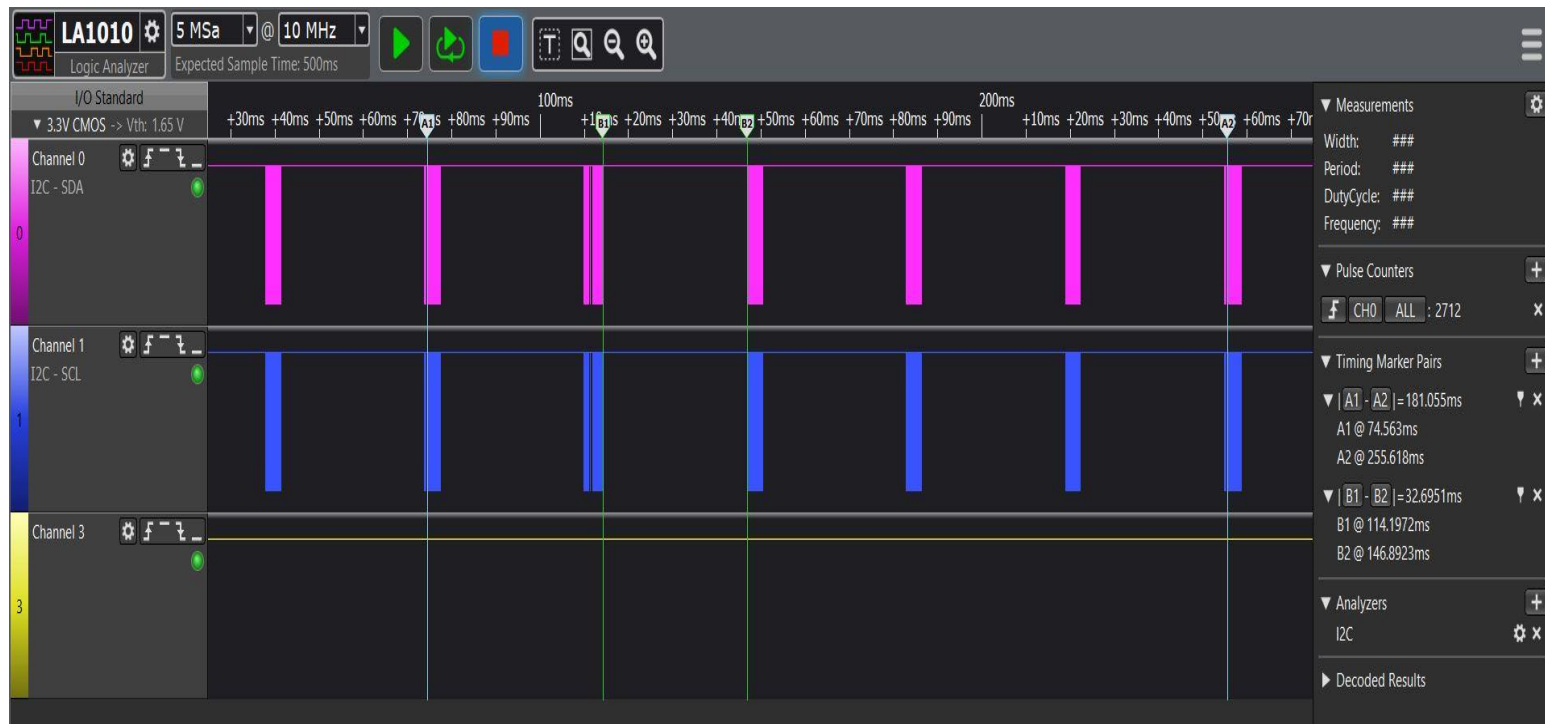
Flexion totale



Sans flexion



Mesure de l'interruption

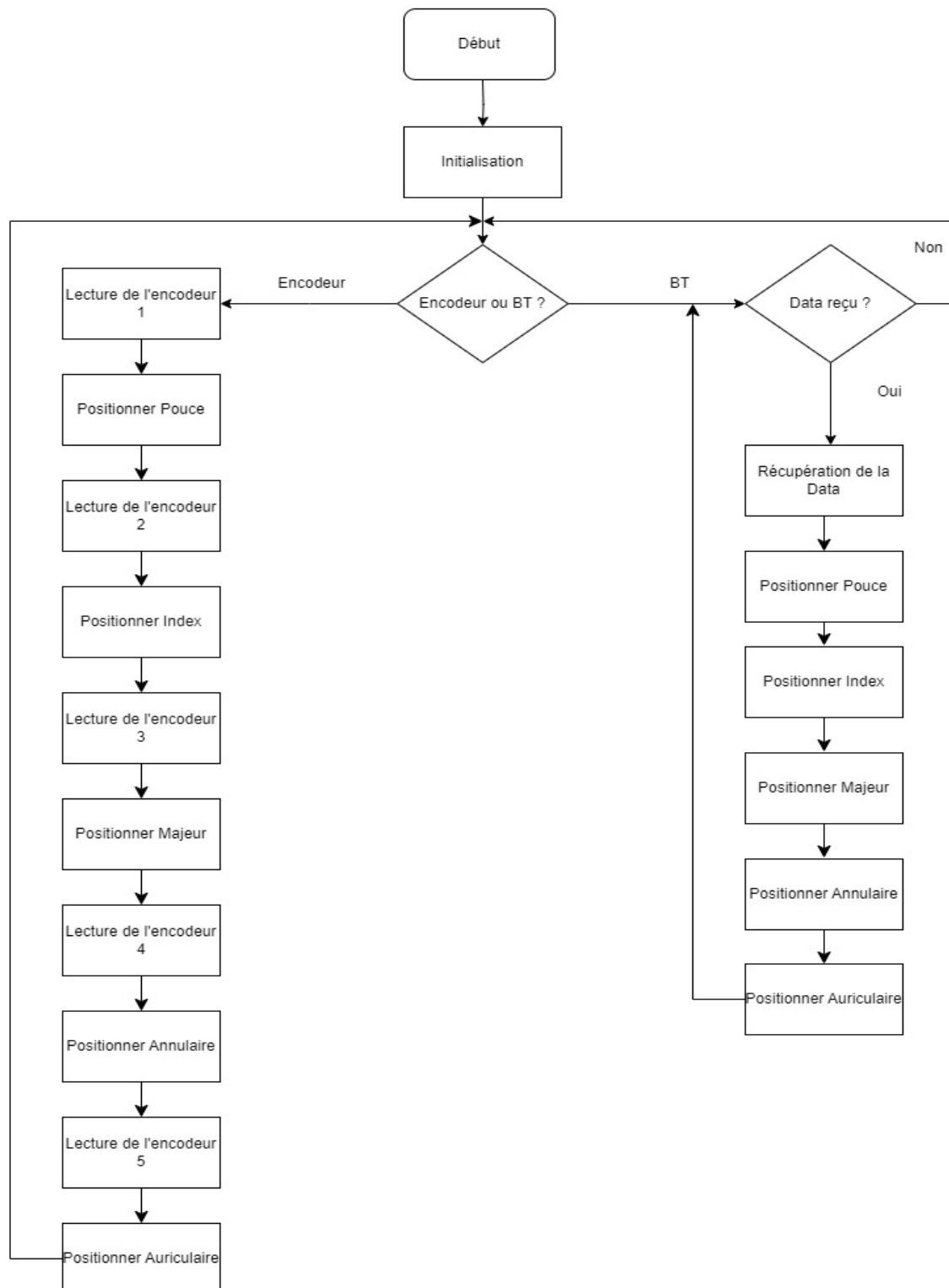


L'espace entre les curseurs A1 et A2 représentent la durée d'une interruption (180ms)

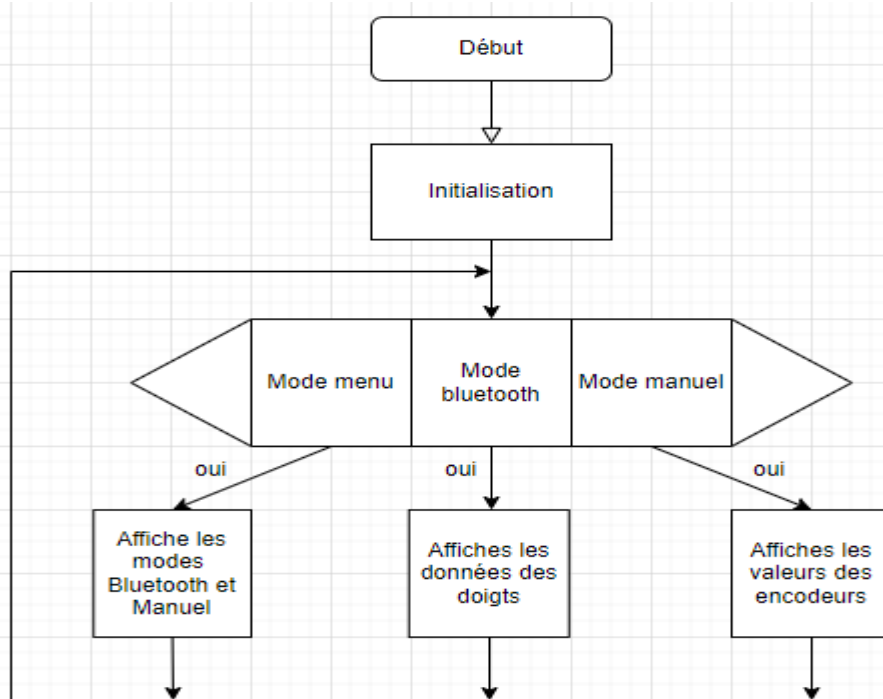
L'espace entre les curseurs B1 et B2 représentent la durée entre le traitement de chaque doigt (on peut voir les traitements qui correspondent à chaque doigt)

6.La programmation

L'ordinogramme de la réception

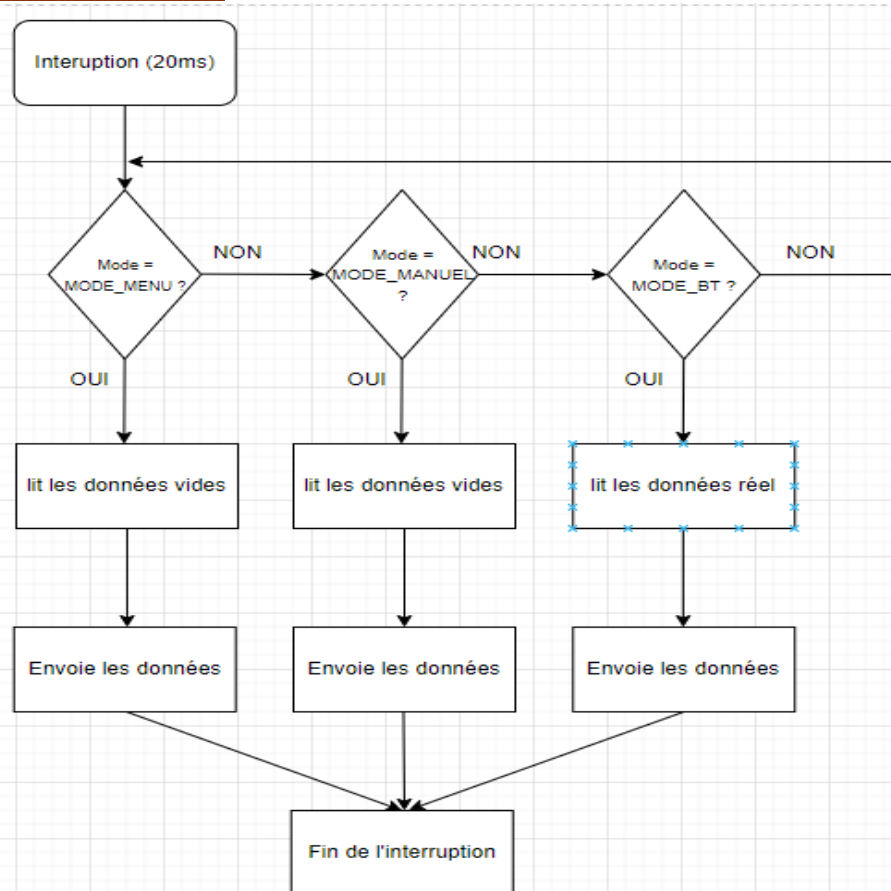


L'ordinogramme de la transmission



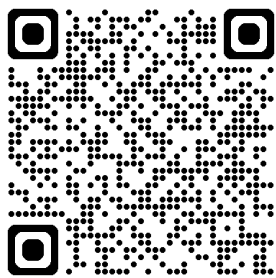
FAIT PAR THOMAS GIARRIZZO

L'ordinogramme de l'interruption

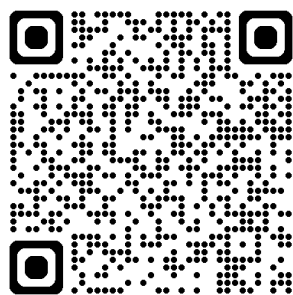


FAIT PAR THOMAS GIARRIZZO

Programme de réception



Programme de transmission

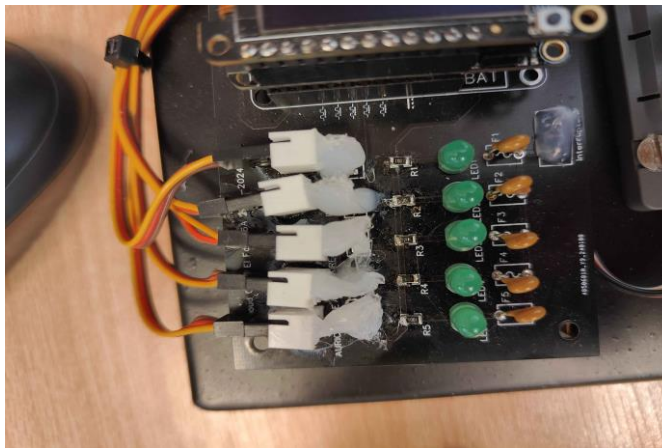


7. La fabrication

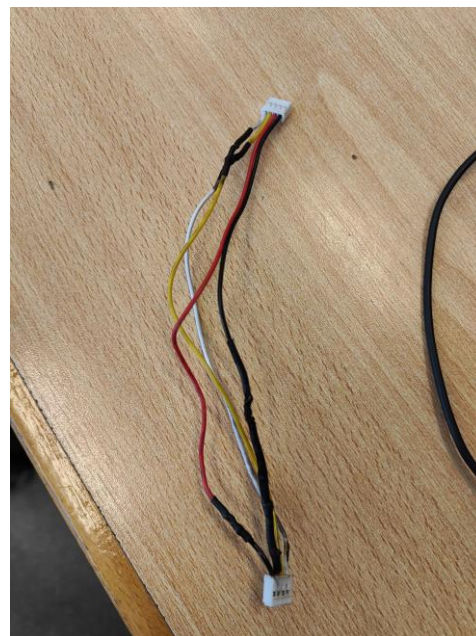
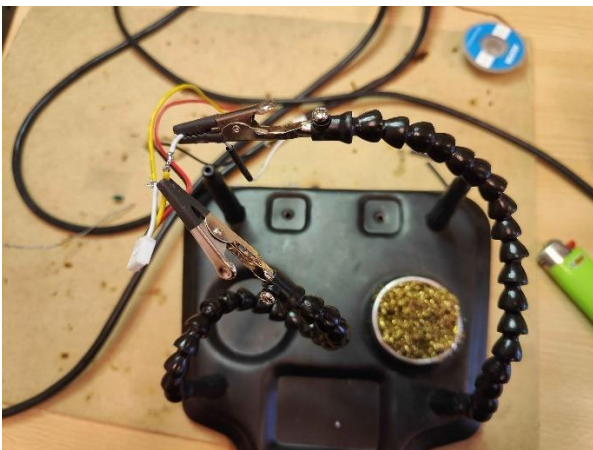
- Premièrement, j'ai commandé ma main et conçu mon PCB en attendant de recevoir la main.
- Après avoir tout reçu, j'ai conçu le plan du support, plier, découper et fais des trous sur de la taule. Ensuite mon père a soudé les coins.
- Pour la partie du gant, j'ai d'abord soudé les ADC pour faire les tests et les mesures et ensuite créer les plans des supports pour la M5, des modules et le capot de mon PCB. Après avoir effectué mes mesures sur les ADC j'ai branché tous mes capteurs de flexion et les ai collés avec du scotch double face aux gants.
- Enfin j'ai mis des entretoises pour fixer le PCB et placer l'interrupteur.
- Pendant tout le procédé de la conception mécanique, je codais en même temps le programme de réception.

8. La mise au point

Durant la conception de mon PCB, j'ai mal mesuré l'espace entre les trous des connecteurs de mon PCB. J'ai donc découpé des pattes de résistances et je les ai soudés avec le PCB et les connecteurs. Pour isoler les pattes, j'ai rajouté de la colle chaude.



J'ai eu un 2-ème problème pour le câble qui connecte ma M5 PAPER à mes ADC. Les câbles du connecteur n'étaient pas dans le bon ordre j'ai donc dû couper et ressouder les fils au bon endroit et rajouter de la gaine thermorétractable.



9. La conclusion

En conclusion, je suis content du résultat de mon projet même si j'aurais aimé rajouter quelque fonctionnalité en plus (ajouter une caméra qui détecte la position des doigts, contrôler les doigts grâce au tactile).

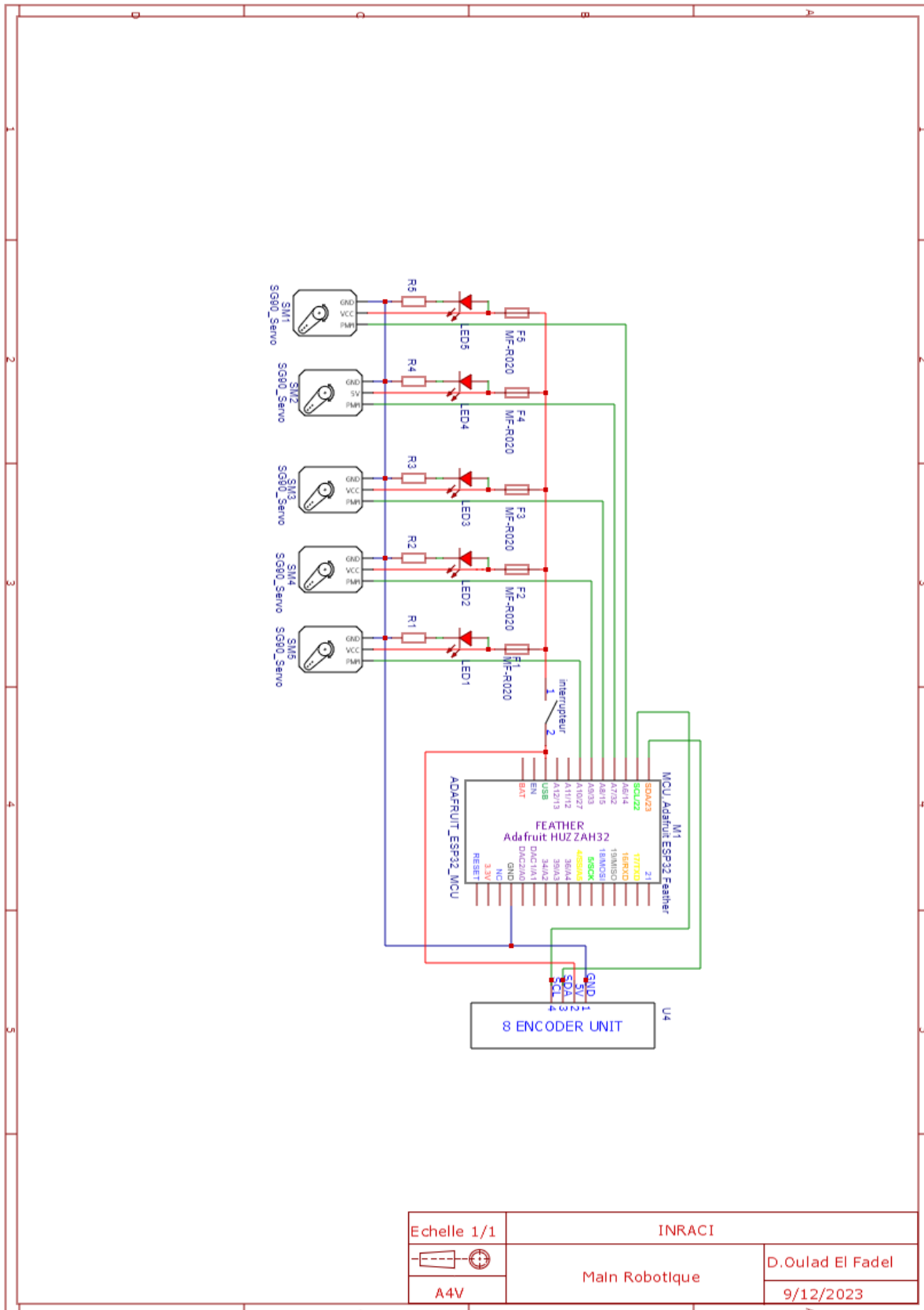
J'ai aimé faire ce projet et le partager avec les gens de mon entourage.

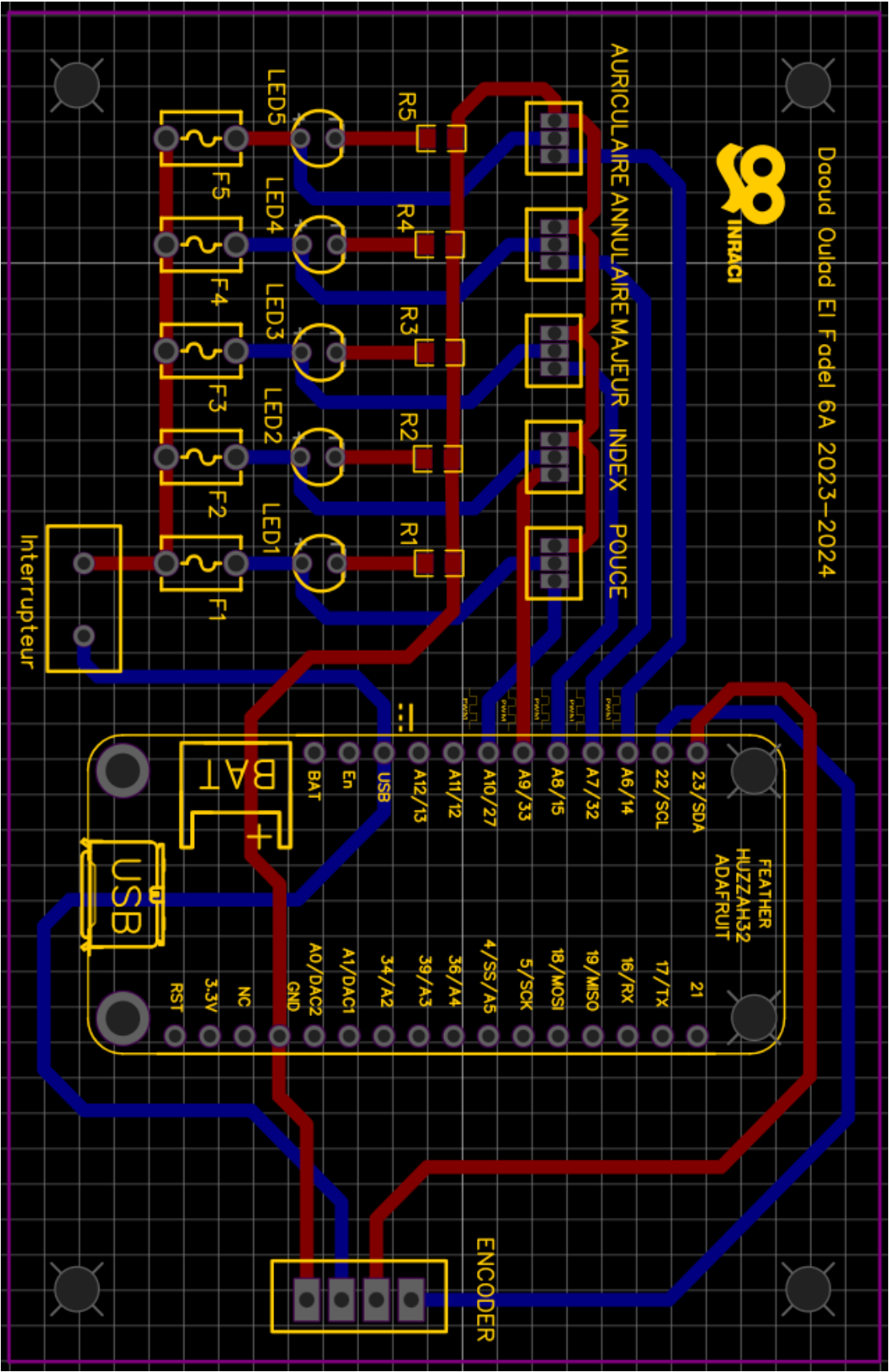
Cette expérience a été enrichissante pour moi, j'ai beaucoup appris et j'ai aimé partager mes travaux et aider les autres.

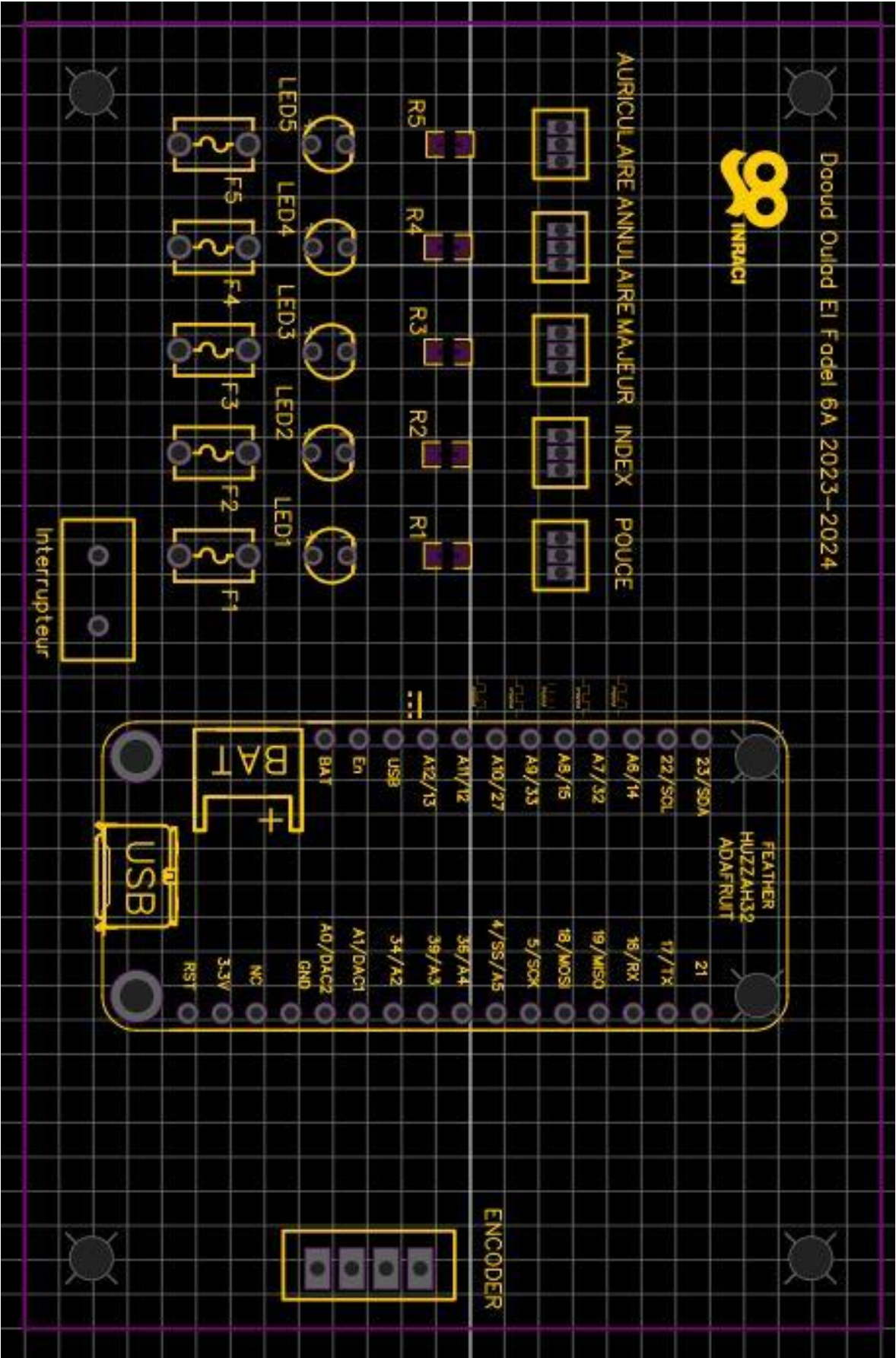
De plus, le fait de travailler en équipe pour mon projet m'a beaucoup aidé. J'ai pu appliquer mes compétences en électronique et les étendre pendant la conception de mon projet.

10. Les annexes

Schéma de principe sans connecteur

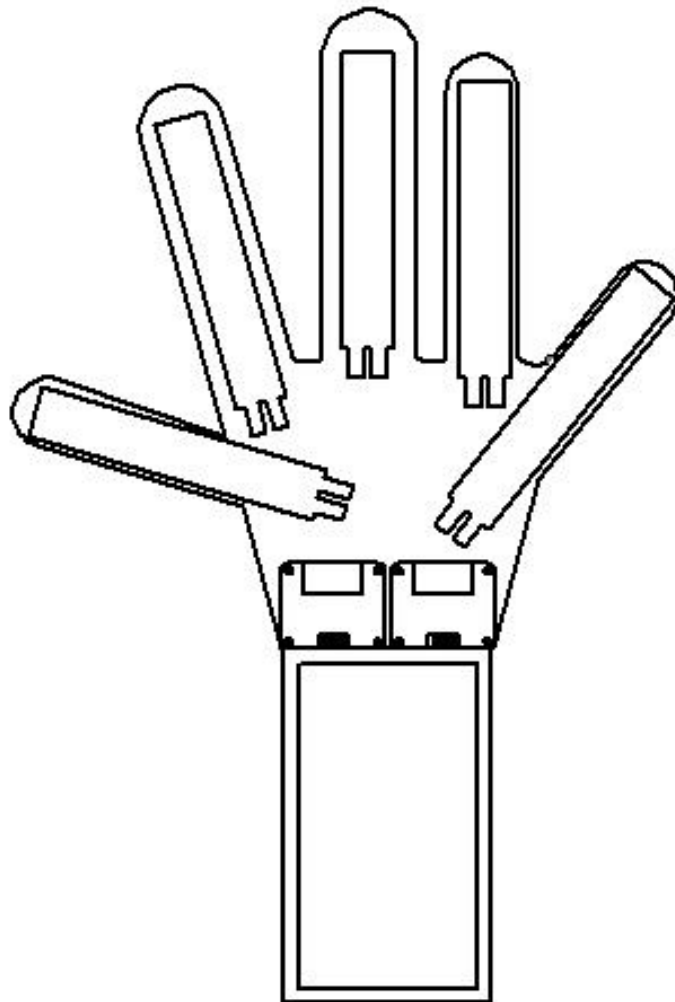






11. Vue 3D

Assemblage du gant



Echelle 1/1

INRACI



A4V

Assemblage du
gant

Oulad El Fadel
Daoud

16/10/2023



Echelle 1/1

INRACI



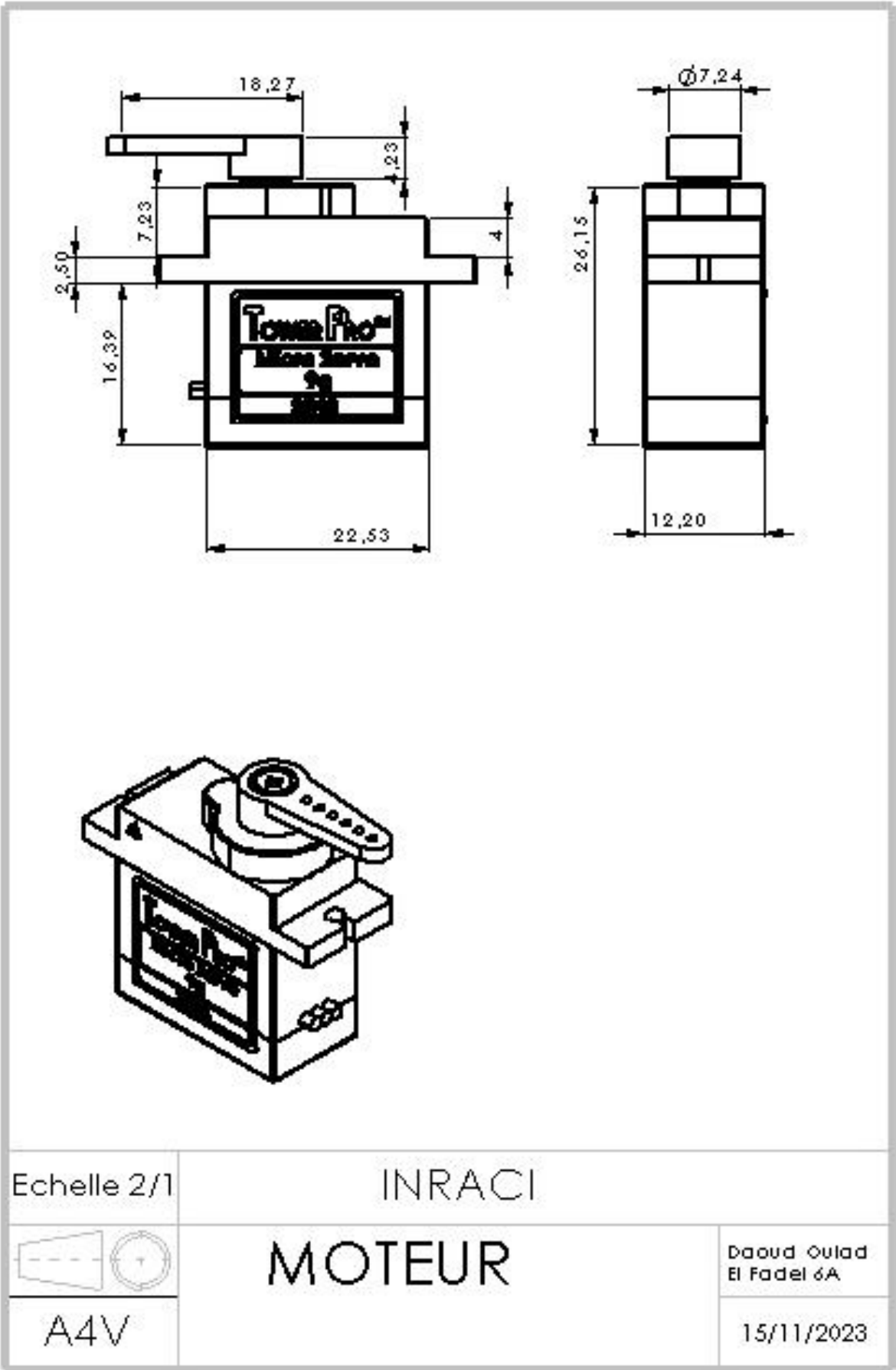
Main Robotisée

Oulad El Fadel
Daoud

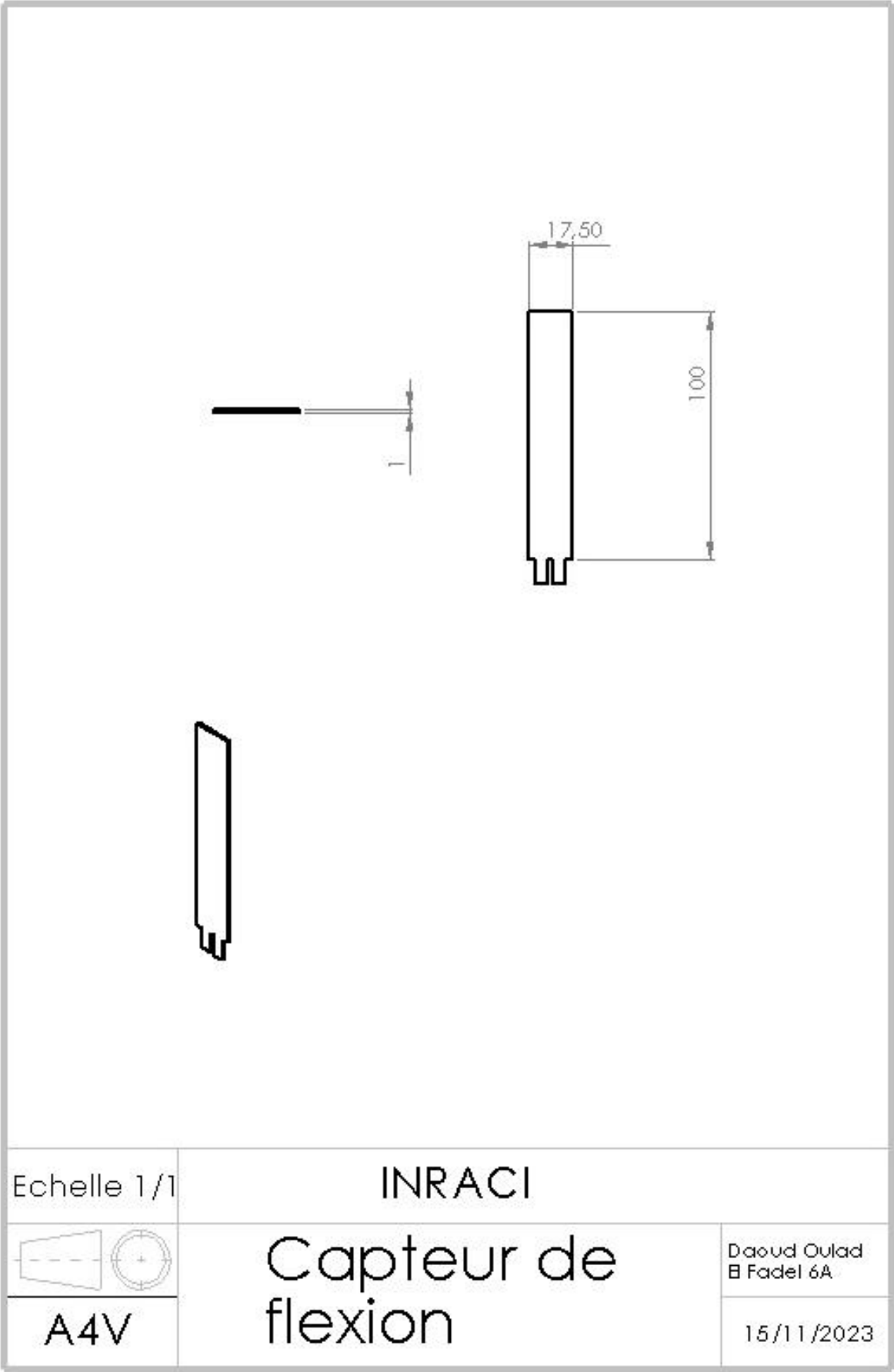
A4V

16/10/2023

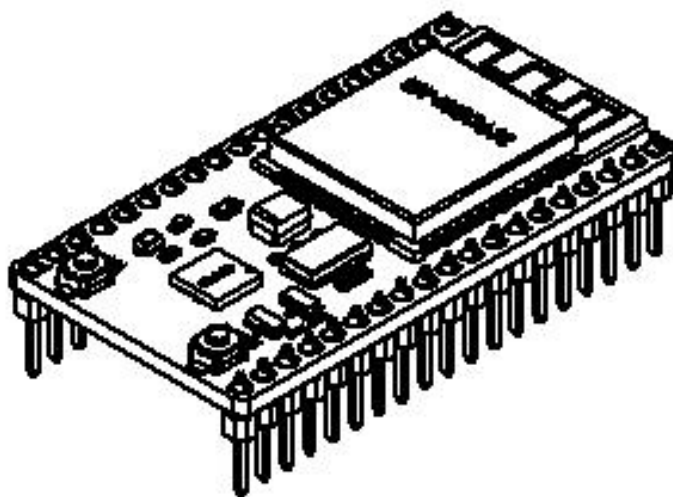
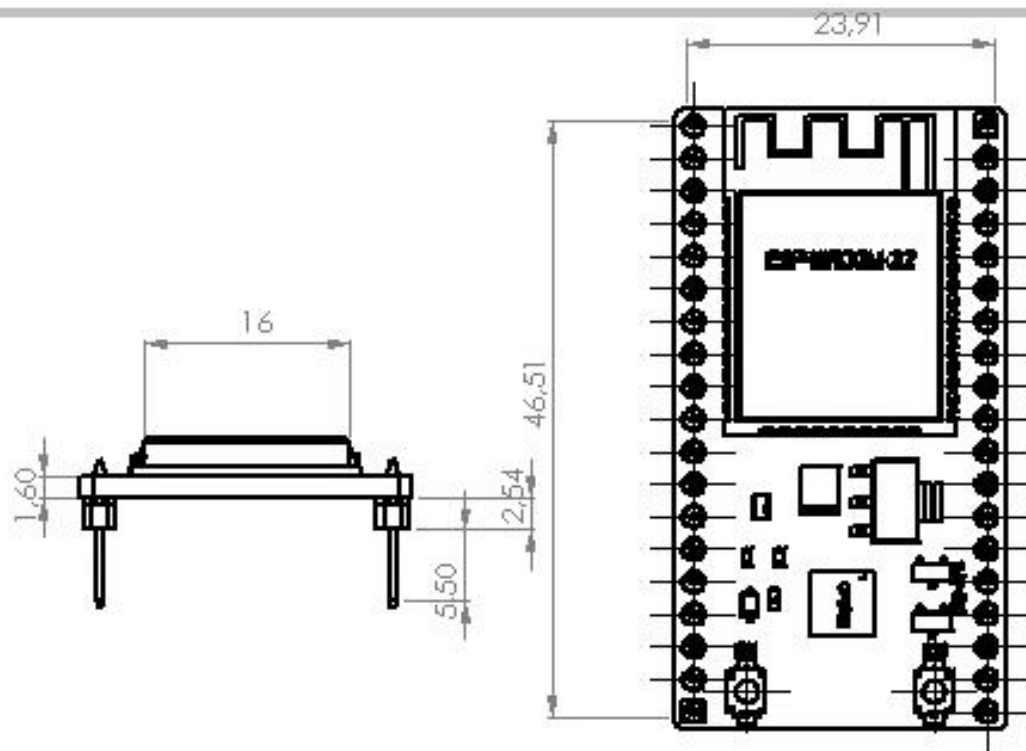
Moteur



Capteur de flexion



ESP32



Echelle 2/1

INRACI



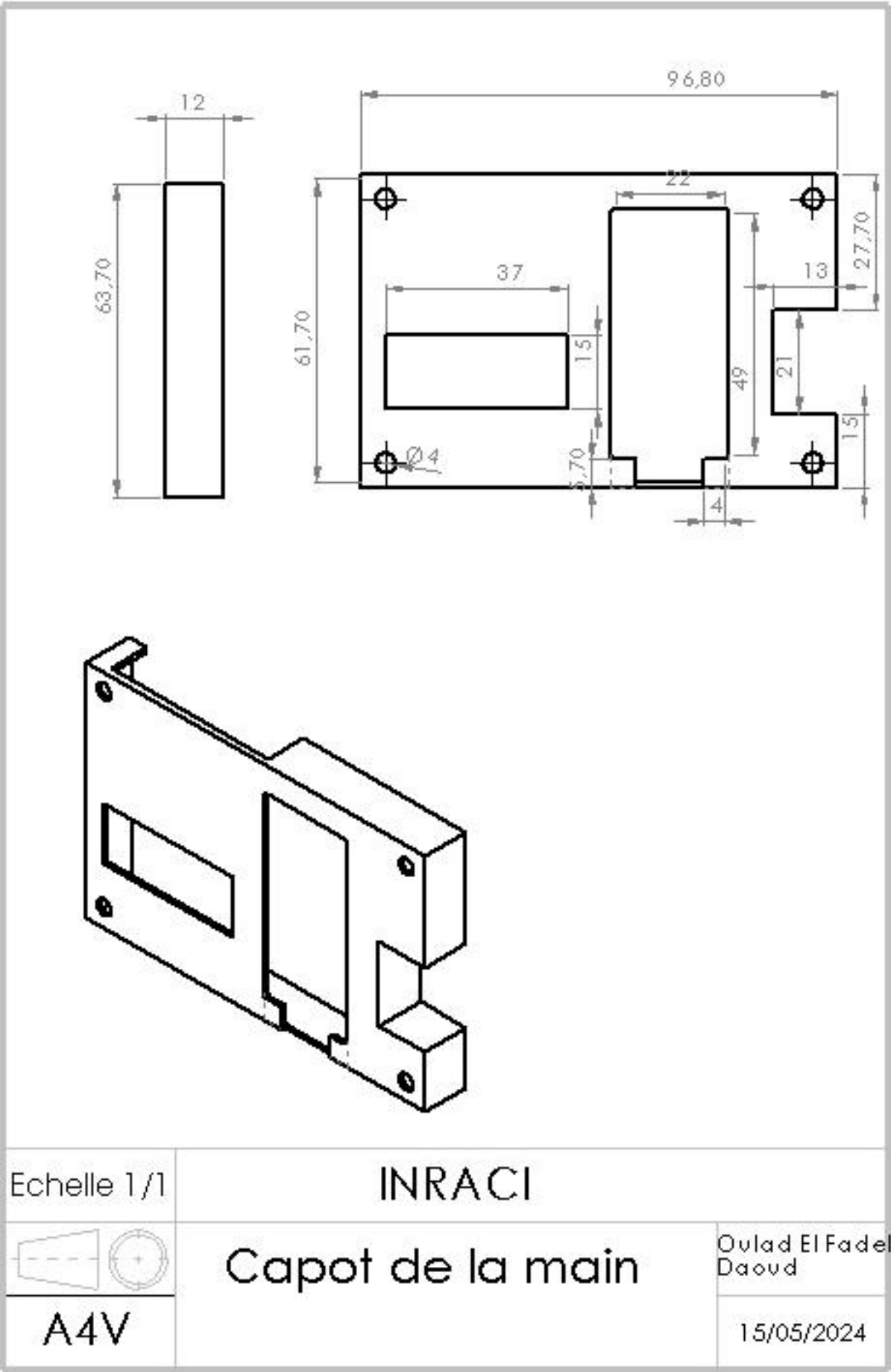
ESP32

Daoud Oulad
B Fadel 6A

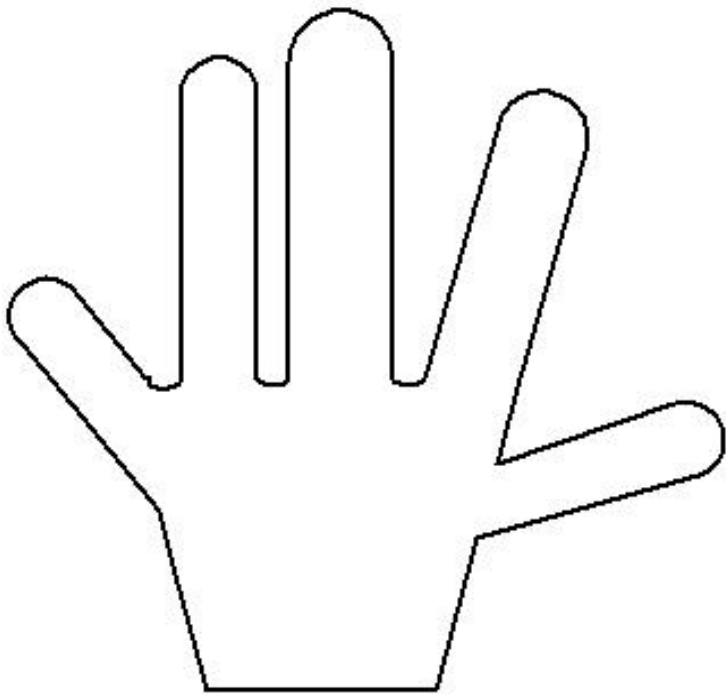
A4V

15/11/2023

Capot de la main

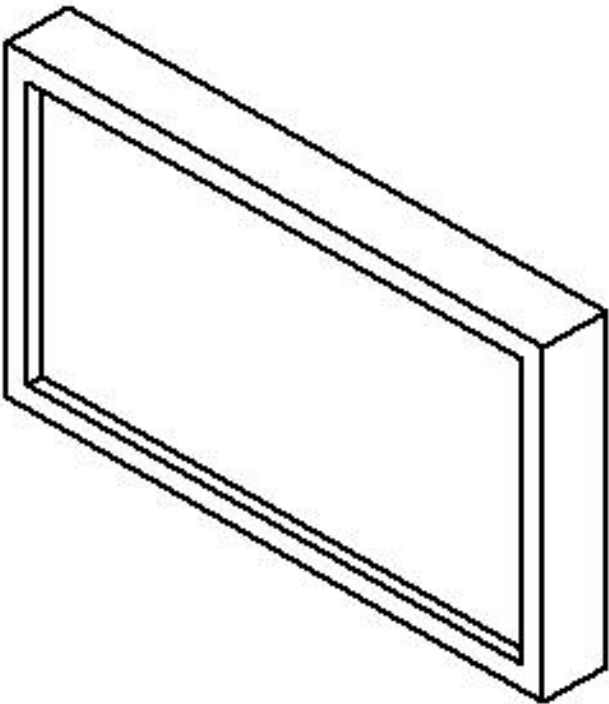



Gant



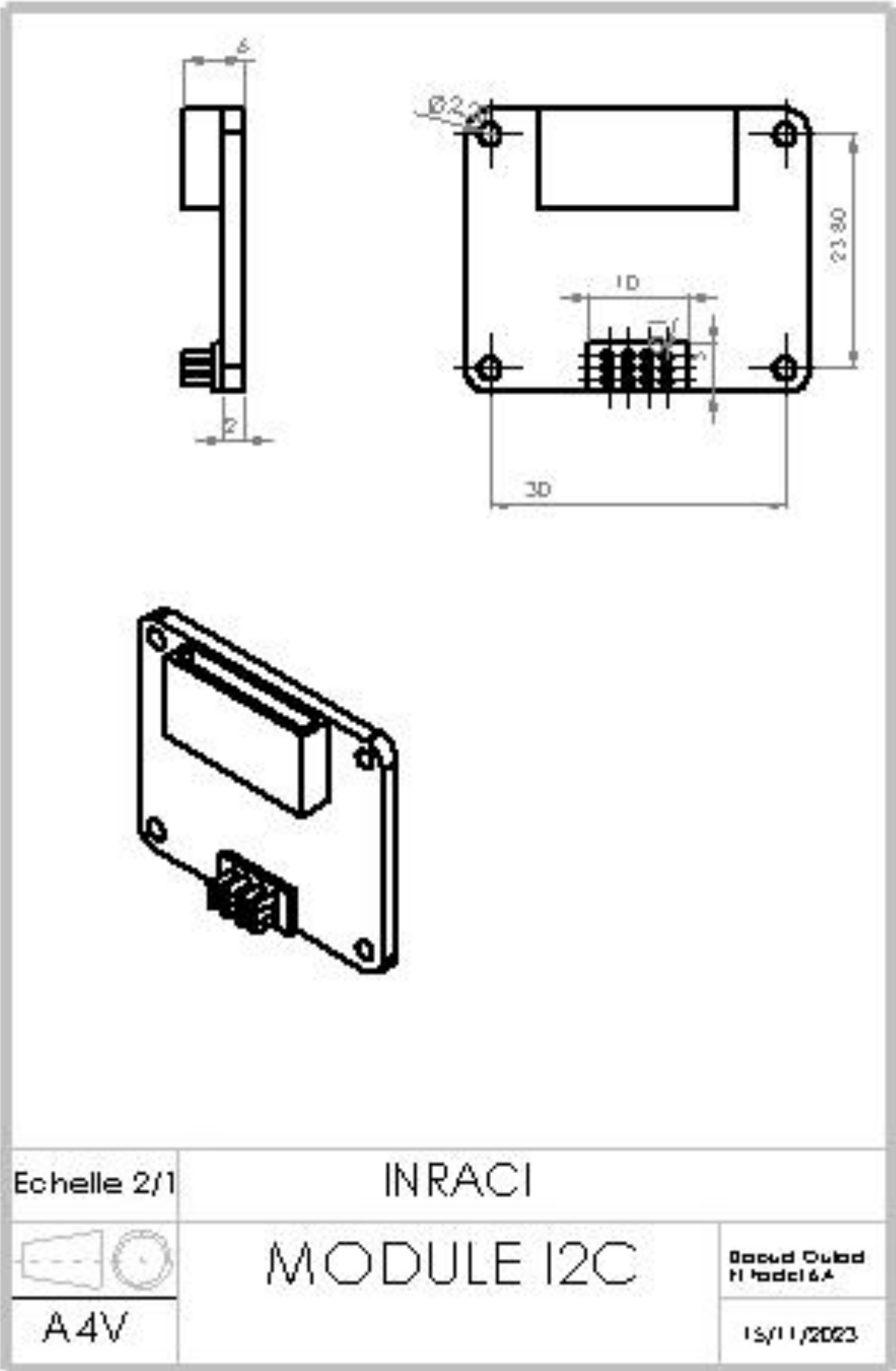
Echelle 1/2	INRACI	
	Gant	Daoud Oulad B Fadel 6A
A4V		15/11/2023

M5 PAPER




Echelle 1/1	INRACI	
	M5 PAPER	Daoud Oulad El Fadel 6A
A4V		15/11/2023

Module I2C



12. Fiche technique des composants

ESP32

Image	
Microcontrôleur	Tensilica LX6 double cœur 32 bits, 240 MHz
Fabricant de microcontrôleurs	Expressif
Système sur puce	ESP32
Tension d'entrée (limite)	Utilisez soit USB (5 V) soit LiPoly (3,7/4,2 V)
Tension de fonctionnement	3,3 V
Mémoire flash	4 Mo
SRAM	520 Ko
Broches GPIO	21
Broches PWM	Tous
Entrées analogiques	14
Wifi	Émetteur-récepteur Wi-Fi 802.11b/g/n HT40
Bluetooth	Mode double (classique et BLE)



M5 PAPER

Resources	Parameter
ESP32-D0WDQ6-V3	240MHz dual core, 600 DMIPS, 520KB SRAM, Wi-Fi
Flash	16MB
PSRAM	8MB
Input Voltage	5V @ 500mA
Ports	TypeC*1, HY2.0-4P*3 , TF-card(microSD) slot
E-Ink Display	Model Number:EPD_ED047TC1 540*960@4.7" Gray scale : 16 Levels Display area : 58.32*103.68mm Display Driver : IT8951E
Physical Button	Multi-function button*1 , Reset Button*1
RTC	BM8563
Antenna	2.4G 3D Antenna
PINS	G25, G32, G26, G33, G18, G19
Battery	1150mAh@3.7V
Working Temp	0°C to 60°C
Net Weight	86g
Gross Weight	100g
Product Dimension	118*66x*10mm
Packaging Dimension	120*70x*14mm
Casing Material	Plastic (PC)

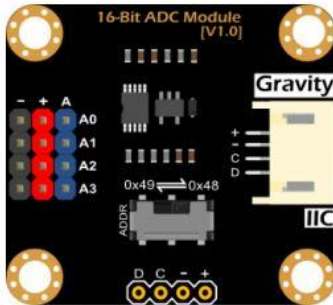


Module ADC

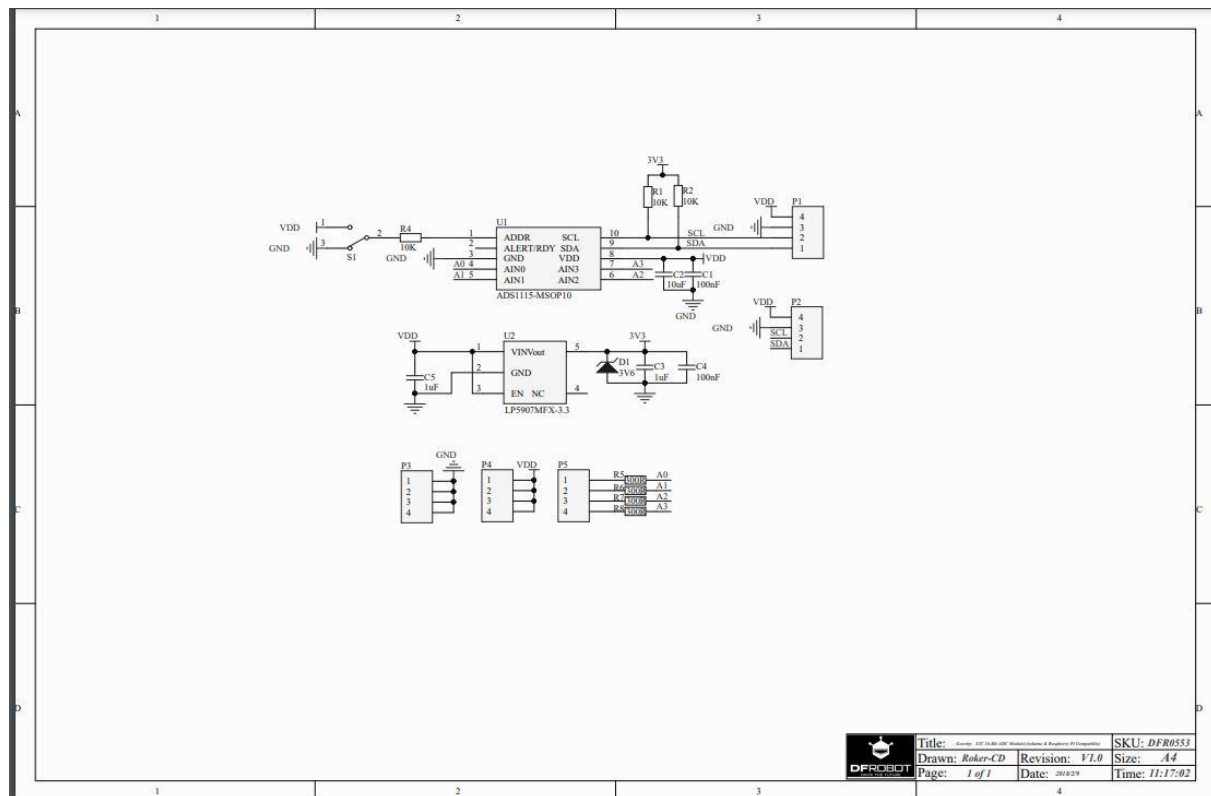
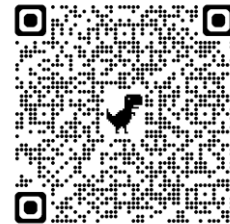
Specification

- Supply Voltage (VCC): 3.3~5.0V
- Analog Signal Detection Range: 0~VCC
- Number of Analog Channels: 4
- ADC Bits: 16 Bit
- Operating Current: 2~3mA (Not include sensor module)
- Interface Type: Gravity I2C
- Interface Level: High 3.3V, Low 0V
- Product Size: 32mm * 32mm(1.26in*1.26in)

Board Overview

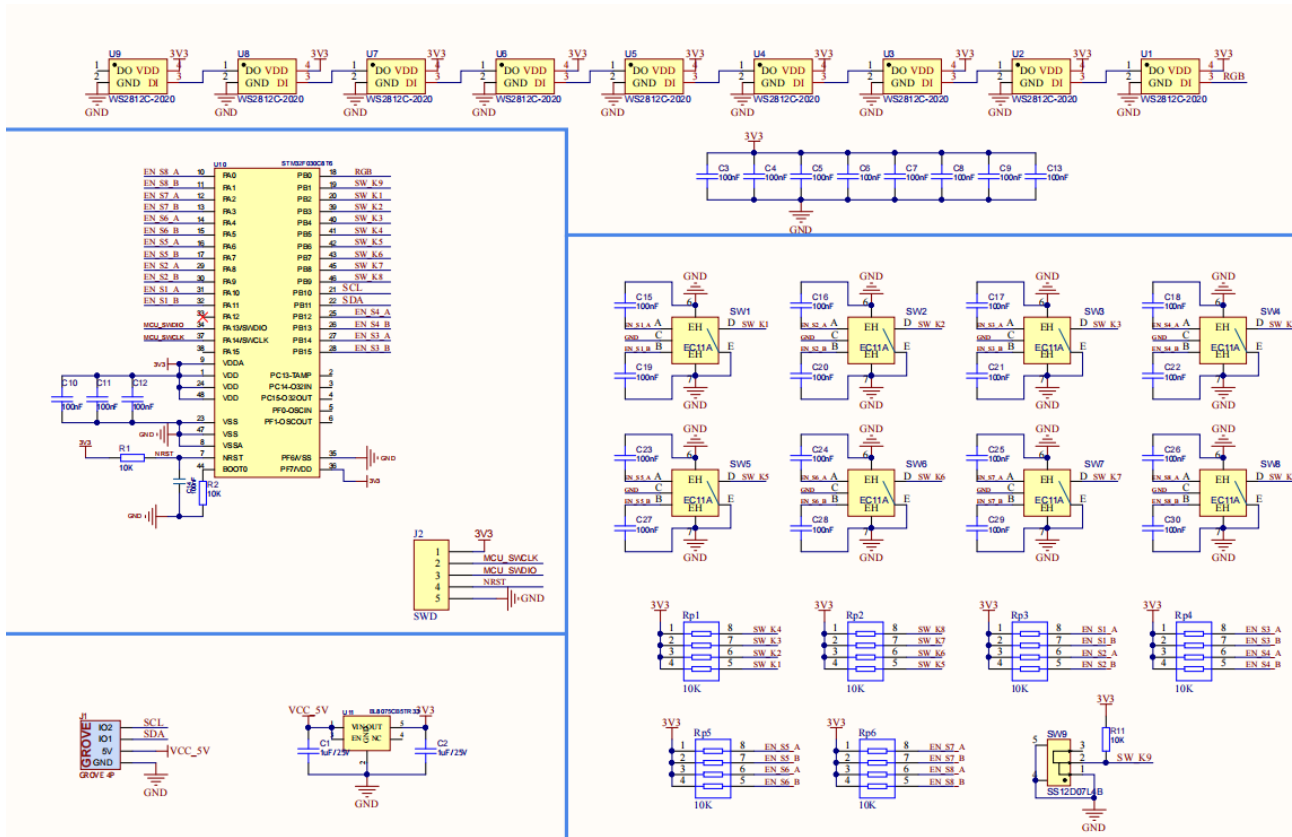


Num	Label	Description
\	VCC	Power VCC(3.3~5.0V)
-	GND	Power GND(0V)
C	SCL	I2C Bus Clock Line
D	SDA	I2C Bus Data Line
A	Analog In	Analog Input Channels: A0, A1, A2, A3
ADDR	I2C Address	I2C address selection switch,I2C address: 0x48, 0x49



Encodeur

Resources	Parameters
MCU	STM32F030C8T6
RGB	WS2812C-2020
Input voltage	5v
I2C communication address	0x41
Product Size	128mm × 24mm ×22.7mm
Package Size	130mm × 27.7mm ×27.7mm
Product Weight	42.8g
Package Weight	52.4g



13. Bibliographie et webographie

Consulté le 1/02/2024

Explication de l'encodeur incrémentale

18h34



Fonctionnement de l'encodeur

18h45



Datasheet de l'encodeur

18h53



Encodeur rotatif

18h12



I²C

19h



Adc



Adc delta sigma



Différence entre les Bluetooth

20h32



Démodulation

21h



Fonctionnement de la Démodulation

21h09



Consulté le 10/10/2024

Servomoteur

18h30



Main robotisée

18h 36

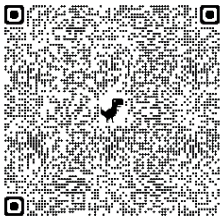


ADC datasheet



Capteur de flexion

19h



Projet

17h30



Esp32

18h



M5paper

18h20

