

Faites le vous-même - préparation et utilisation du capteur IR

Le but de ce projet est de commander une DEL par action musculaire. Ainsi, la contraction du bras va entraîner l'allumage de la DEL. Cela permet de montrer que des capteurs moins chers peuvent être une voie de recherche pour la création de prothèses moins coûteuses.

Pour plus d'information, la fiche technique du capteur IR que nous utilisons. Elle contient des informations supplémentaires que nous n'avons pas évoqué dans les paragraphes ci-après.

** Fiche technique en Anglais**

https://www.sparkfun.com/datasheets/Robotics/QR_QRE1113.GR.pdf

Vous trouverez dans les pages qui suivent, différentes étapes :

- 1- **Le matériel**
- 2- **L'assemblage**
- 3- **Le code**
- 4- **Le placement sur le bras (work in progress)**
- 5- **Test de fonctionnement (work in progress)**
- 6- **Conclusion (work in progress)**

Etape 1 - Matériel

Pour ce projet, vous aurez besoin de matériel électronique, de quelques outils (pour l'assemblage) et d'un ordinateur (pour transférer le programme et lire ce DIY) pour le faire fonctionner.

Nous avons commandé l'ensemble du matériel sur GoTronic (site de matériel électronique) afin d'essayer de trouver un rapport qualité/prix optimal.

Vous êtes bien sûr libre de choisir d'autres marketplaces si vous trouvez des prix plus intéressants.

1- Capteur de mesure de distance et d'intensité lumineuse IR

Quantité : 1 Prix : 3.90 Fabricant : Sparkfun

Lien d'achat : <https://www.gotronic.fr/art-suiveur-de-ligne-qre1113-20220.htm>

Fiche technique : https://www.sparkfun.com/datasheets/Robotics/QR_QRE1113.GR.pdf

2- Carte Arduino Nano 3.0

Quantité : 1 Prix : 8.90€ Fabricant : Arduino

Lien d'achat : <https://www.gotronic.fr/art-carte-nano-3-0-26126.htm>

Fiche technique : <https://store.arduino.cc/arduino-nano>

3- Platine d'essai -Kit plaque de montage SD80A

Quantité : 1 Prix : 9.50€ Fabricant : Non disponible:/

Lien d'achat : <https://www.gotronic.fr/art-kit-plaque-de-montage-sd80a-25864.htm>

• *Matériel optionnel – Si vous le possédez déjà, inutile de le racheter –*

4- Fer à souder – Station VTSS4N

Quantité : 1 Prix : 17.9€ Fabricant : Velleman

Lien d'achat : <https://www.gotronic.fr/art-station-vtss4n-16972.htm>

5- Cable d'étain – Soudure ESP002/50

Quantité : 1

Prix : 5.40

Fabricant : BMJ

Lien d'achat : <https://www.gotronic.fr/art-soudure-esp002-50-28648.htm>

Cable USB type B

– si vous possédez une calculatrice scientifique de type Texas Instrument, vous devriez en posséder un. Il va permettre le transfert des données Ordinateur/Carte Arduino.



• Sur l'image de la page précédente :

- La **platine d'essai** en blanc sur l'image
- Le **capteur IR** est dans le sachet au centre
- La **carte Aduino nano**, déjà sur la platine d'essai
- Différents **câbles** sur la droite de l'image

Fer à souder.

Fer allumé, potentiomètre réglé à 350° C.

Eponge humidifiée en dessous du fer.



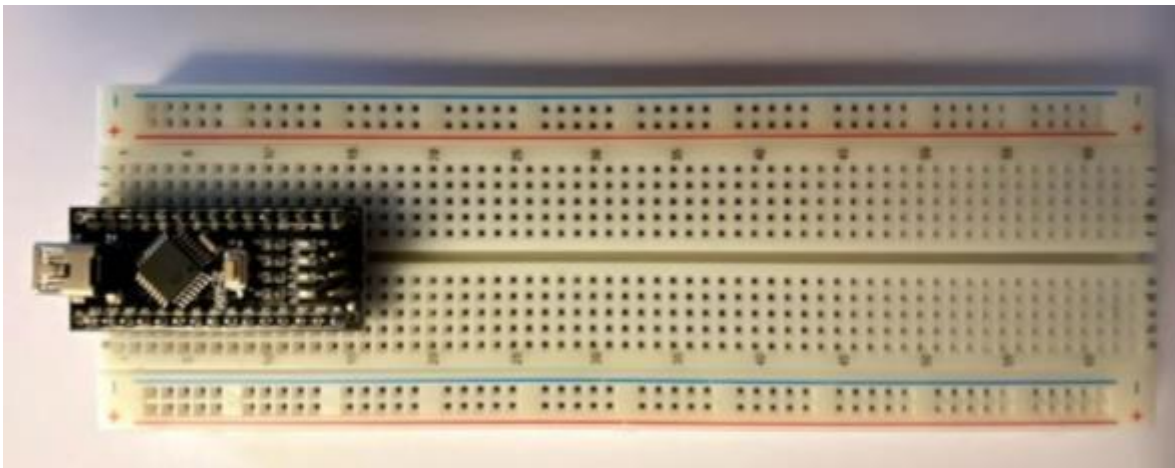
Etape 2 - Assemblage

On va pouvoir passer à l'assemblage de l'ensemble du matériel.

Vous pouvez tout d'abord assembler la carte Arduino nano sur la platine d'essai. (Pas de soudure nécessaire, juste l'enfoncer).

Prenez la **platine d'essai** face à vous, de manière que les chiffres soient lisibles. A 90° droite par rapport à l'image.

Munissez-vous de la **carte Arduino Nano**, enfoncez là de part et d'autre des colonnes. (En haut de la carte, vous avez de petits trous circulaires, ils doivent être en face du C et du G de la platine d'essai. Une fois trouvé, enfoncez la carte).



Ensuite, il va falloir souder trois câbles sur le capteur IR. (Voir tuto soudure)

<https://www.youtube.com/watch?v=Bv4iFrju1wM> (exemple de vidéo pour réaliser une soudure)

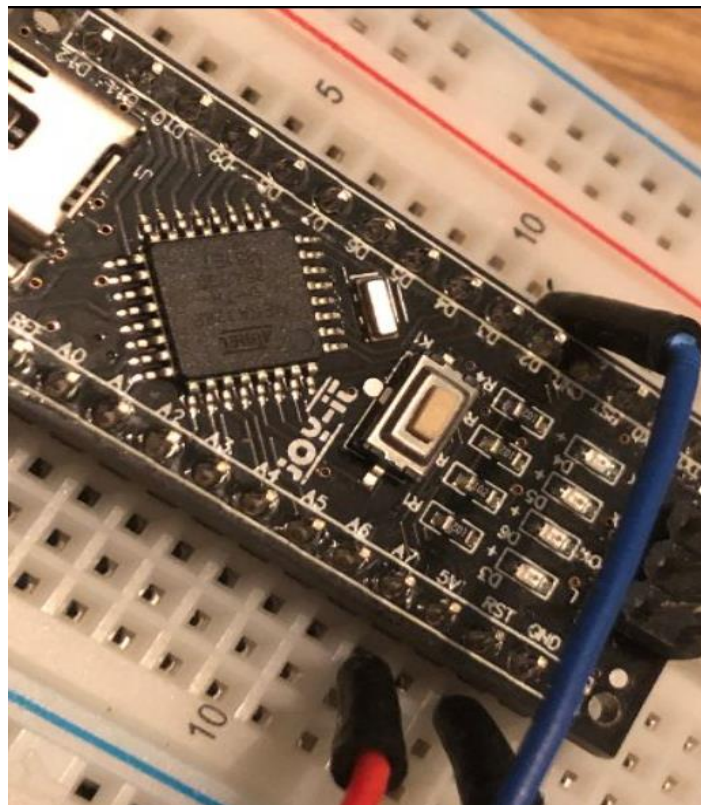
- Le câble Bleu est raccordé sur le port OUT du capteur et sur la digitale 2 (D2)
- Le câble Rouge est raccordé sur le VCC du capteur et sur le 5V de la platine.
- Le câble Noir est raccordé sur le GND du capteur et sur le GND de la platine.

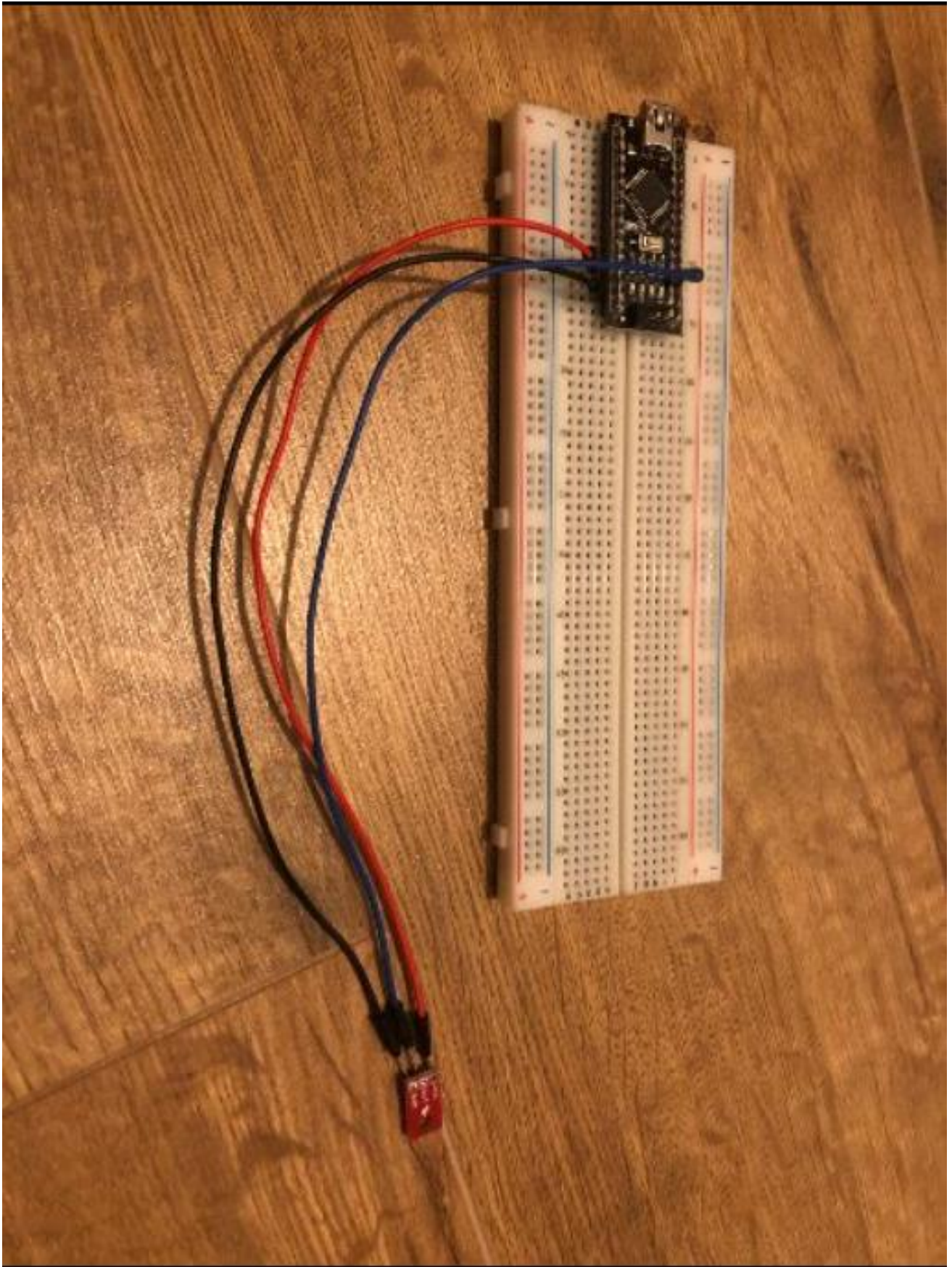
Les **GND**, **VCC** et **SIG** sont écrits sur le bord gauche du capteur sur son verso

Les couleurs sont libres à vous, mais par convention, nous nous sommes basés sur ces dernières



Une fois les câbles installés, on vient les raccorder sur les ports de la platine d'essai correspondant





Etape 3- le code

Une fois **le capteur** et **la carte** assemblés, un code en C++ (langage de programmation) va permettre d'utiliser le capteur et d'allumer la LED lorsqu'il y a une contraction du muscle.

Pour cela il va vous falloir installer l'application **Arduino**, rien de très bien compliqué là-dedans.

(Si vous l'avez déjà, vous pouvez directement récupérer le code.)

- Pour le télécharger : <https://www.arduino.cc/en/Main/Software>

Download the Arduino IDE



- **Choisissez** la version selon l'ordinateur sur lequel vous êtes.
- Une fois terminé, L'IDE va s'ouvrir et va vous demander d'accepter des propositions. Faites-le.

Ensuite, vous allez arriver dans la **console d'Arduino**.

```
sketch_jun04a | Arduino 1.8.12
Fichier Édition Croquis Outils Aide
sketch_jun04a
void setup() {
  // put your setup code here, to run once:
}

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

Supprimer les quelques lignes pré-écrites et les remplacer par le code proposé ci-dessous.

Le code à utiliser : Copiez-le et collez le dans la console :


```

// #####
// DEBUT DE VOTRE PROGRAMME ARDUINO

float Voltage;

float MoyValue;

float Tolerance= 0.05;

//Code for the QRE1113 Digital board
//Outputs via the serial terminal – Lower numbers mean more reflected
//3000 or more means nothing was reflected.

int QRE1113_Pin = 2; //connected to digital 2

/*
Variable declaration
*/

int led = 4; // Pin N° of the pin where the LED is connected

void setup(){
  int QRE_Value = readQD();
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);

  Serial.println("Beginning of the program");
  pinMode(13, OUTPUT); // Configures pin 13 of the arduino board at output

  // don't move for 10s
  Serial.println("Hey.. Don't contract for 10s, i will take value each 0.1s");

```

```

// retrieve the values returned for 10 seconds
float TotalValue = 0;
int N = 100;
for(int i=0; i<N; i=i+1)
{
// Process values every 0.1 seconds
delay(100);

// add the value
TotalValue = TotalValue + QRE_Value; //make the sum of all value between this 10s
}

// average these values for the time T=10S
MoyValue = TotalValue / N; //make the average of all the value taken by the sensor
}

void loop(){

int QRE_Value = readQD();
Serial.println(QRE_Value);

if (Voltage > MoyValue + MoyValue*Tolerance)
{
//turn on la L.E.D.
digitalWrite(13, HIGH); // Switches the output to logic HIGH, Lights LED
}
else

```

```

{
//turn off la L.E.D.
digitalWrite(13, LOW); // Switches the output to LOW logic level, Switches the LED off
}

}

int readQD(){
//Returns value from the QRE1113
//Lower numbers mean more reflective
//More than 3000 means nothing was reflected.
pinMode( QRE1113_Pin, OUTPUT );
digitalWrite( QRE1113_Pin, HIGH );
delayMicroseconds(10);
pinMode( QRE1113_Pin, INPUT );

long time = micros();

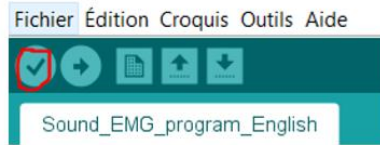
//time how long the input is HIGH, but quit after 3ms as nothing happens after that

while (digitalRead(QRE1113_Pin) == HIGH && micros() - time < 3000); int diff = micros() - time;
return diff; }

```

- **Connecter** ensuite votre câble USB type B avec l'ordinateur et la carte Arduino

Ensuite **vérifier** le programme, en cliquant sur le V en haut à gauche.



Si vous rencontrez des problèmes, voici quelques paramètres à prendre en compte,

- Le port sur lequel est connecté votre carte Arduino

Sous **Windows**, allez dans les **paramètres**, dans **Périphériques**, puis dans **Appareils Bluetooth et autres**. Dans **autres appareils**, vous verrez une indication comme quoi votre USB est connecté avec le port précis.



Appareils Bluetooth et autres

USB OPTICAL MOUSE

Audio

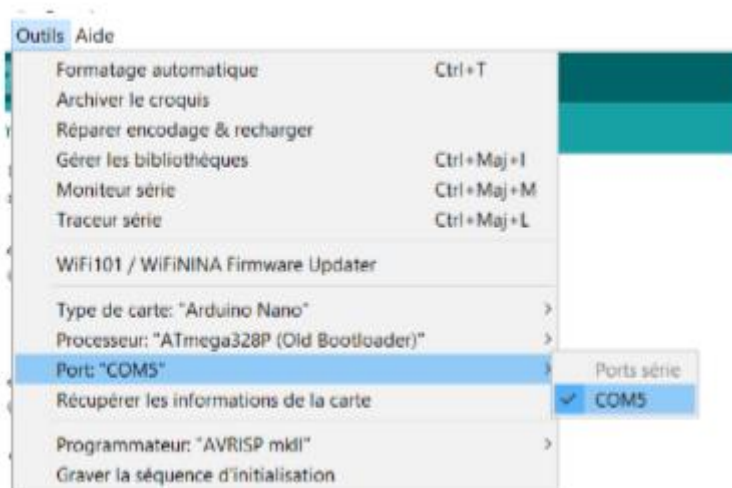


Autres appareils



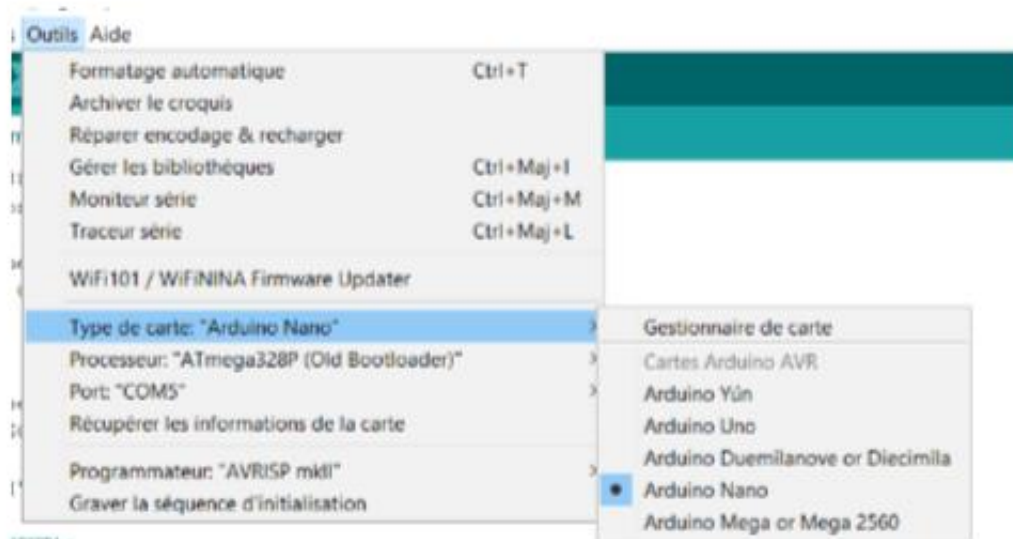
USB-SERIAL CH340 (COM5)

Vous pouvez ensuite en allant dans **outils** dans l'IDE Arduino, modifier le **Port** sur lequel vous êtes connectés.



Pour continuer dans les paramètres à régler sur l'IDE, on retrouve :

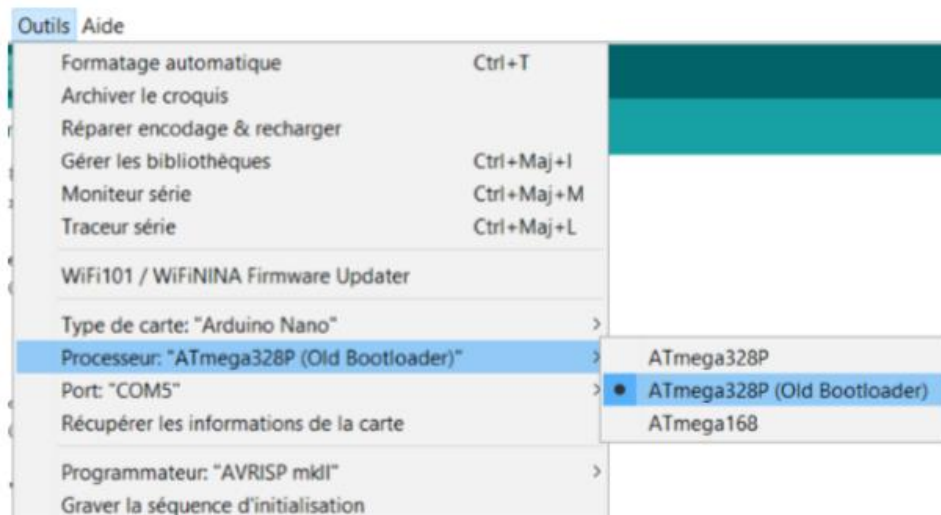
- Le choix du type de carte ; dans notre cas l'Arduino Nano



Le Processeur à choisir

Ici, on utilise ATmega328p (Old Bootloader).

Il faut donc retourner dans **outils** et **Processeur**



Développement en cours

- A cause de contraintes techniques, nous ne sommes pas en mesure de vous présenter la suite de ce DIY. Cependant, cela arrive bientôt.