

# Un détecteur d'obstacles pour malvoyants

## Santé et prévention

Arthur Jacquin

Candidat 27397

- Introduction
- Capteurs de distance
- Interface utilisateur
- Produit fini
- Annexes

# Introduction - Contexte

- Solutions existantes partiellement satisfaisantes
  - Chiens-guides
  - Canes
  - Canes électroniques



Figure 1 : L'UltraCane (CECIAA, 1055 €)

Peut-on proposer une meilleure alternative ?

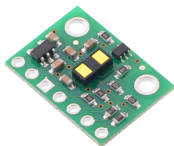
# Introduction - Cahier des charges fonctionnel

Fonction	Critère	Valeur
Détecter la distance de l'obstacle le plus proche	Portée	>3 m
	Faisceau de détection	
	Réactivité	>5 Hz
Traiter l'information	Cohérence	
Communiquer l'information à l'utilisateur	Clarté	
	Progressivité	
Être reproductible	Accessibilité économique	<50 €
	Accessibilité technique	
Être ergonomique	Compacité	
	Maniabilité	
	Robustesse	

# Capteurs de distance - Modèles

## Capteur

VL53L1X

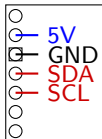


Lumineuse

Infrarouge (940 nm)

19,40 €

I<sup>2</sup>C



MB1013



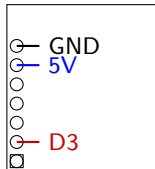
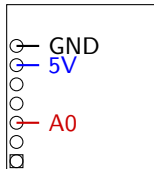
Sonore

Ultrasonore (42 kHz)

39,90 €

Analogique

MLA



Type d'onde

Domaine

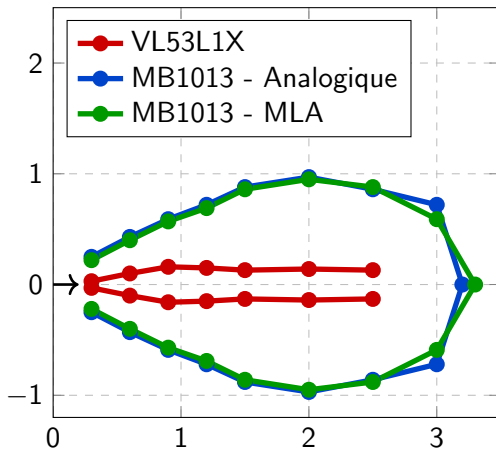
Prix

Interface

# Capteurs de distance - Faisceau de détection



(a) Dispositif



(b) Zone détectée par le capteur placé à l'origine  
(vue de haut, en m)

Figure 2 : Faisceau de détection

## Capteurs de distance - Précision

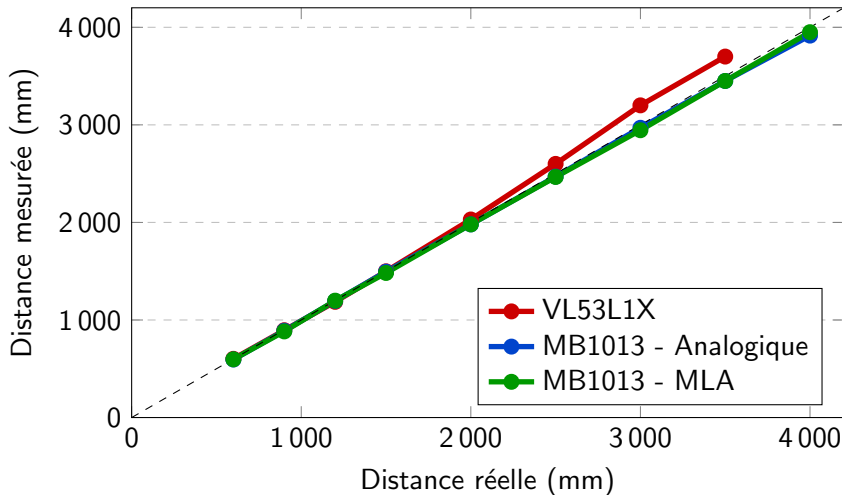


Figure 3 : Mesures de distance



## Capteurs de distance - Réactivité

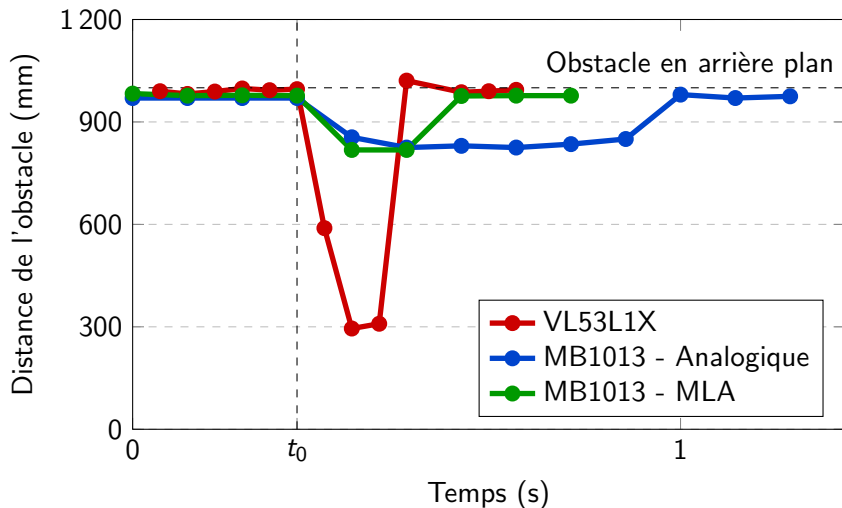


Figure 4 : Détection d'une feuille A4 lâchée à 30 cm du capteur à  $t_0$

# Capteurs de distance - Conclusion

- Récapitulatif des tests

<b>Capteur</b>	VL53L1X	MB1013
<b>Portée</b>	++	+++
<b>Faisceau</b>	Directif	Large
<b>Précision</b>	Suffisante	Suffisante
<b>Réactivité</b>	+++	+
<b>Prix</b>	19,40 €	39,90 €

- Capteur retenu : VL53L1X
- Léger bruit : moyenne glissante sur 3 points (0,15 s)

# Interface utilisateur - Modèles

- Signal sonore : désagréable
- Vibrations : comparaison de différents vibreurs

Vibreux Gotronic



1,30 €

VPM2



4,20 €

Gravity



2,60 €

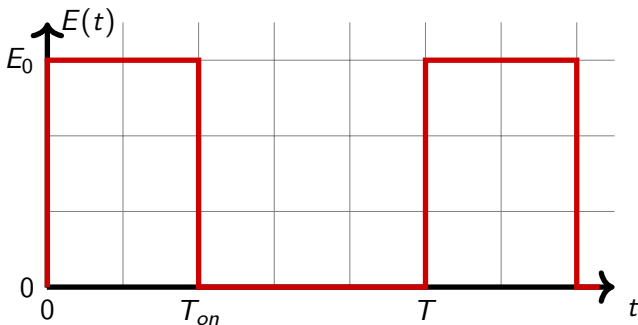
SMD



1,30 €

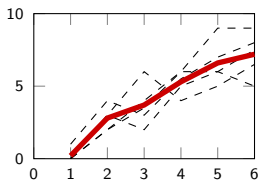
# Interface utilisateur - Modulation de largeur d'amplitude

- *Pulse Width Modulation (PWM)*
- Caractérisé par le rapport cyclique (*duty cycle*) noté  $D = \frac{T_{on}}{T}$

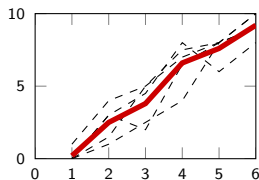


$$\langle E(t) \rangle = \frac{1}{T} \int_0^T E(t) dt = \frac{1}{T} \left( \int_0^{T_{on}} E_0 dt + \int_{T_{on}}^T 0 dt \right) = D \cdot E_0$$

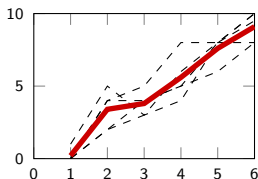
# Interface utilisateur - Courbes intensité-ressenti



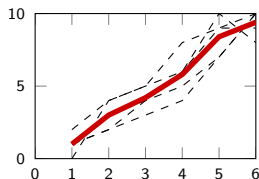
(a) Vibreur Gotronic



(b) VPM2



(c) Gravity



(d) SMD

**Figure 5 :** Intensité de vibration ressentie en fonction de l'intensité des vibrations (5 utilisateurs en pointillés, moyenne en rouge)

## Interface utilisateur - Fonction dist\_to\_intensity

$$f(x) = \begin{cases} ax^3 + bx^2 + cx + d & x \leq 2000 \\ 0 & x > 2000 \end{cases}$$

$$(a, b, c, d) = (-1.8 \cdot 10^{-8}, 1.08 \cdot 10^{-4}, -0.234, 255)$$

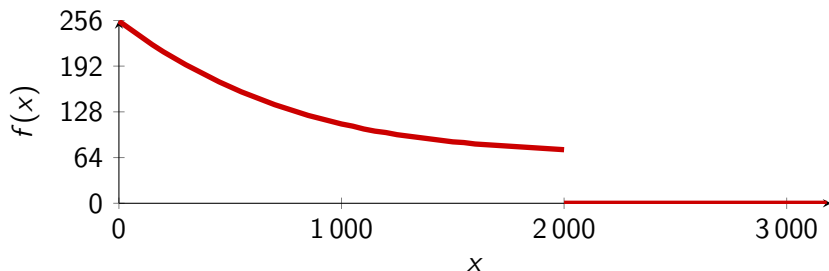


Figure 6 : Fonction dist\_to\_intensity

## Produit fini - Conception générale

- Choix du microcontrôleur
- Alimentation : externe
- Nombre de couples capteur-vibreux : un seul
- Boîtier : aucun
- Disposition générale des composants

# Produit fini - Composants

- Composants retenus

Composant	Nom	Prix (€)
Microcontrôleur	Arduino Nano Every	8,80
Capteur de distance	VL53L1X	19,40
Vibreur	VPM2	4,20
<b>TOTAL</b>		<b>32,40</b>

- Matériel nécessaire
  - Station de soudure
  - Scotch double face
  - Fils de prototypage



# Produit fini - Schéma de soudure

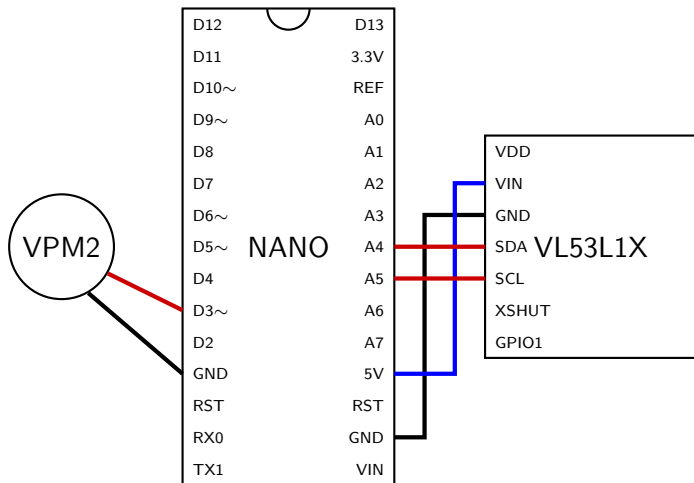
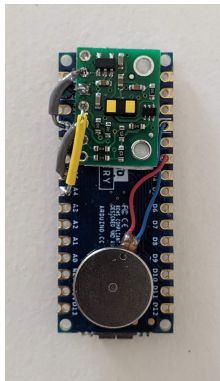
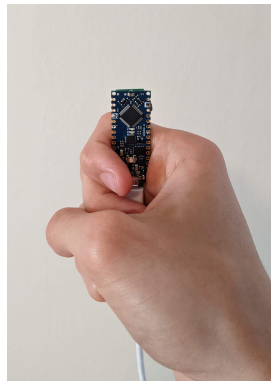


Figure 7 : Schéma de soudure

# Produit fini - Conclusion



(a) Vue de face



(b) Dans la main

Figure 8 : Produit fini

- Documentation disponible en ligne

- Liens
- Prototype final sur *breadboard*
- Liste complète des composants
- Code
  - Bibliothèques et variables
  - Initialisation
  - Boucle principale
- Schéma électrique du VL53L1X

- Présentation du projet : `jacquin.xyz/tipe`
- Toutes les ressources : `github.com/arthur-jacquin/tipe`

## Prototype final sur *breadboard*

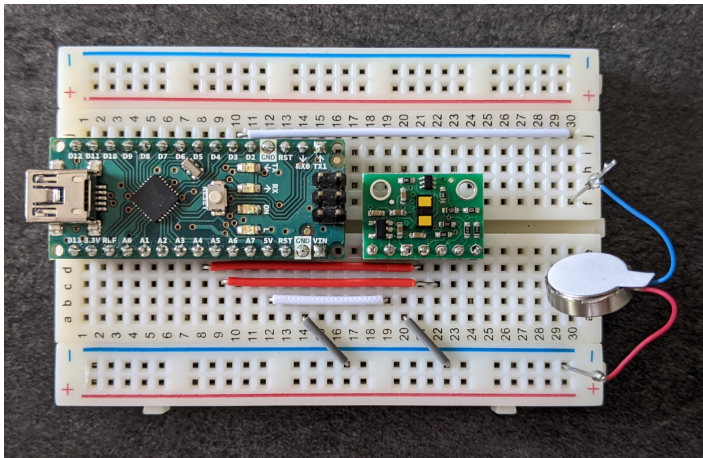


Figure 9 : Prototype final sur *breadboard*

# Liste complète des composants

Catégorie	Nom/objet	Qté	Fabriqueur	Réf. fabr.	Fournisseur	Prix (€)
Microcontrôleur	Arduino Nano	1	Arduino	A000005	Gotronic	22,90
Microcontrôleur	Arduino Nano Every	3	Arduino	ABX00028-3P	Arduino	25,10
Microcontrôleur	Seeeduino XIAO	1	Seedstudio	102010328	Gotronic	5,90
Microcontrôleur	Beetle	1	DFRobot	DFR0282	Gotronic	9,80
Capteur de dist.	VL53L1X	1	Polulu	3415	Gotronic	19,40
Capteur de dist.	HRLV-MaxSonar-EZ1	1	Maxbotic	MB1013	Gotronic	39,90
Vibreur	Vibreur miniature	2	Gotronic	25355	Gotronic	2,60
Vibreur	VMP2	1	Solarbotics	VPM2	Gotronic	4,20
Vibreur	Gravity	1	DFRobot	DFR0440	Gotronic	2,60
Vibreur	SMD	1	Seedstudio	316040005	Gotronic	1,30
Prototypage	Kit plaque de montage	1	Gotronic	SD80A	Gotronic	9,50
Prototypage	Alimentation	2	Velleman	PS910	Gotronic	15,90
Prototypage	Kit pour prototypage	1	Elegoo	E0	Amazon	20,99
Soudure	Station de soudage	1	Velleman	VTSS4N	Gotronic	17,90
Soudure	Pompe à dessouder	1	Gotronic	13580	Gotronic	3,50
Soudure	Fil de soudure	1	Gotronic	13673	Gotronic	8,30
<b>TOTAL</b>						<b>209,79</b>

# Code - Bibliothèques et variables

```
1 // Libraries
2 #include <Wire.h>
3 #include <VL53L1X.h>
4
5 // Declare used pins
6 const int pinVibr = 3;
7 VL53L1X sensor;      // I2C protocol
8
9 // Declare variables
10 int dist;             // Distance measured (in mm)
11 int intensity;        // Command for vibration (from 0 to 255)
12 const int n = 3;      // Number of points for sliding average
13 int logs[n];          // Array to store last measurements
14 int k = 0;            // Sliding index in array
15 int avg = 0;          // Average value
```

# Code - Initialisation

```
17 void setup () {
18     pinMode(pinVibr, OUTPUT); // Set vibration motor as output
19
20     for (int i = 0; i < n; i++) { // Initialize array
21         logs[i] = 0;
22     }
23
24     // Communication through I2C (for VL53L1X), set at 400 kHz
25     Wire.begin();
26     Wire.setClock(400000);
27
28     // Configure VL53L1X in long distance mode
29     // Read rate frequency set as 20Hz (<< 400 kHz)
30     sensor.setTimeout(500);
31     if (!sensor.init()) { while (1); } // No sensor detected
32     sensor.setDistanceMode(VL53L1X::Long);
33     sensor.setMeasurementTimingBudget(50000);
34     sensor.startContinuous(50);
35 }
```



# Code - Boucle principale

```
37 void loop () {
38     // Reading the sensor measurement
39     dist = sensor.read();
40
41     // Computing sliding average
42     k = (k + 1)%n;
43     avg = avg + (dist - logs[k])/n;
44     logs[k] = dist;
45
46     // Modifying vibration intensity
47     intensity = dist_to_intensity(avg);
48     analogWrite(pinVibr, intensity);
49 }
50
51 int dist_to_intensity (int x) {
52     // Takes dist in mm, return command between 0 and 255
53     if (x > 2000) {
54         return 0;
55     } else {
56         return -1.8*pow(10, -8)*x*x*x + 0.108*x*x/1000 - 0.234*x
           + 255;
57     };
58 }
```

# Schéma électrique du VL53L1X

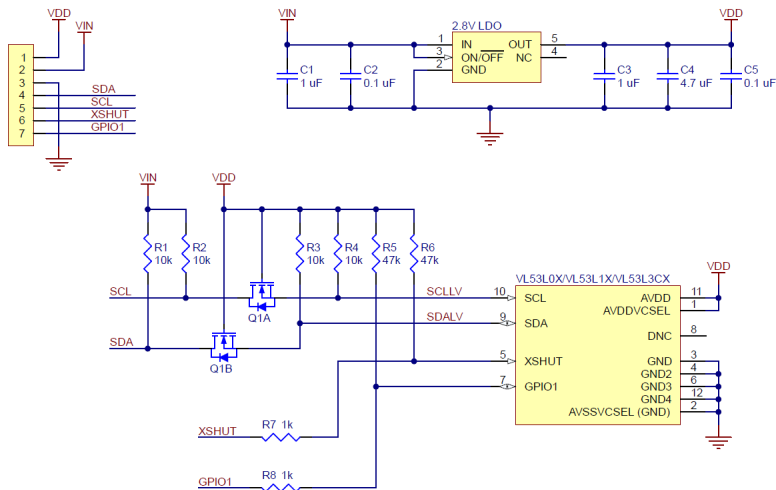


Figure 10 : Schéma électrique du VL53L1X