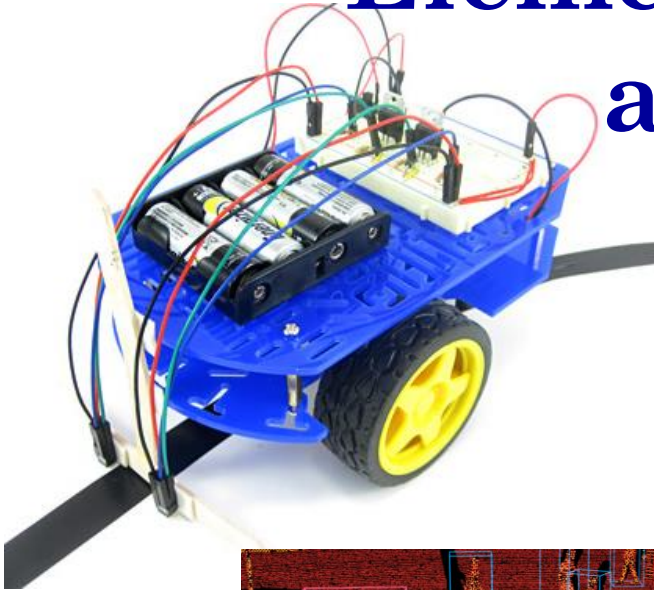


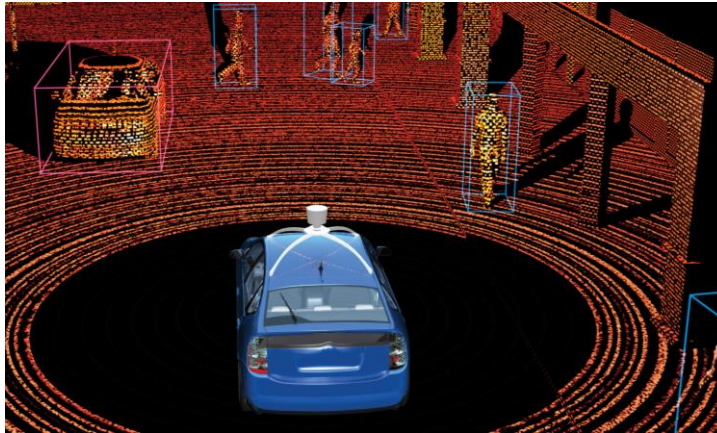
Éléments de robotique avec Arduino



Pascal MASSON

(pascal.masson@unice.fr)

*Version projection
Edition 2018-2019-V10*



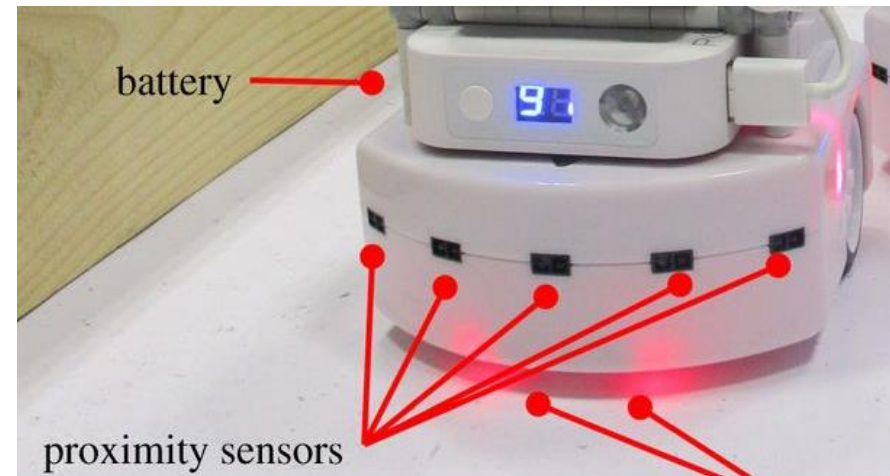
Distance et détection d'obstacles

Introduction

1. Distance par ultra-son
2. Distance par laser EN COURS D'ECRITURE
3. Détecteur de proximité / ligne
4. Mapping 2D et 3D EN COURS D'ECRITURE

❑ Pourquoi détecter des obstacles ?

- Eviter de stopper le robot dans ses déplacements
- Eviter de la détérioration du robot ou de son environnement

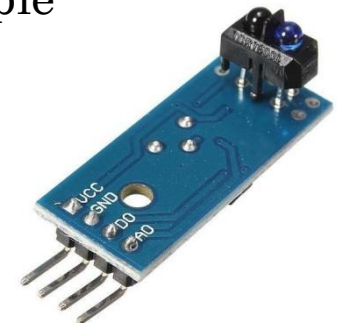


Thymio

- Les capteurs de proximité permettent aussi de suivre des trajectoires pour déplacer du matériel par exemple

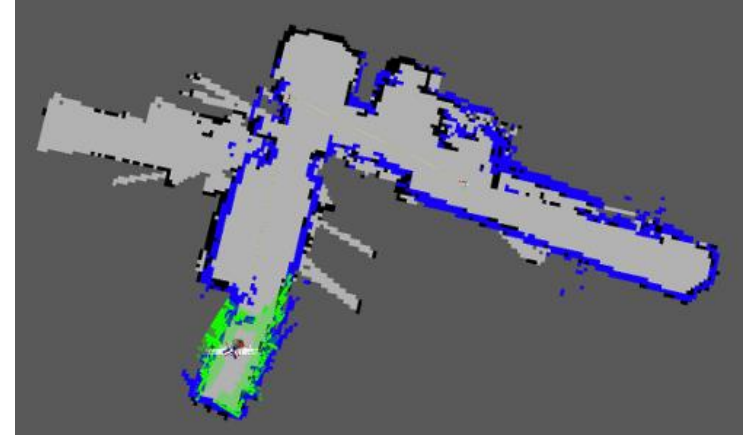


Robot industriel

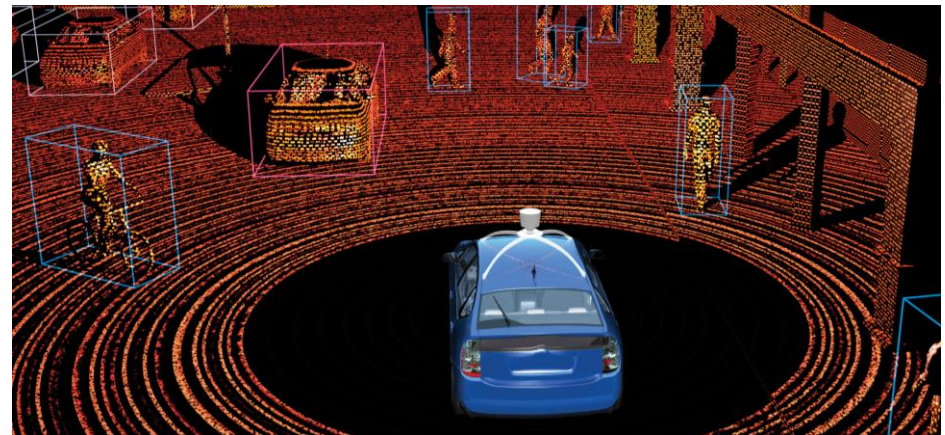


□ Pourquoi mesurer des distances ?

- Pour connaître l'environnement du robot afin de décider d'une trajectoire ou pour faire une cartographie



2D



3D

1.1. Le son

□ Définition

- Le son est une vibration mécanique d'un fluide, qui se propage sous forme d'ondes longitudinales grâce à la déformation élastique de ce fluide.

□ Gamme de fréquence

- Il existe 4 gammes de fréquences pour les sons :
 - ✓ Infrasons : fréquences inférieures à 16 Hz
 - ✓ Son : entre 16 Hz et 20 kHz. Cela correspond à l'ouïe des êtres humains
 - ✓ Ultrasons : fréquences comprises entre 20 kHz et 1 GHz
 - ✓ Hypersons : fréquences supérieures à 1 GHz
- Les fréquences perceptibles dépendent des êtres vivants. Par exemple, les chats entendent jusqu'à 65 kHz et les chauves souris peuvent émettre un son jusqu'à 120 kHz (en fonction des espèces)

1.1. Le son

□ Visualisation de la propagation du son

- On s'intéresse ici à une propagation dans l'air. Les vibrations d'un objet (haut-parleur, diapason, cordes vocales ...) sont transmises aux molécules de l'air en contact avec lui avec un mécanisme de dilation/contraction. Ce mouvement se propage ensuite de proche en proche dans l'air
- Pour visualiser ce mouvement, on pourra regarder ces deux vidéos :
 - ✓ <https://www.youtube.com/watch?v=Nh6Figt5jkl>
 - ✓ <https://www.youtube.com/watch?v=Nh6Figt5jkl>
- Remarque 1 : on conçoit aisément que le son ne peut pas se propager dans le vide de l'espace (ou à travers du double vitrage)
- Remarque 2 : la vitesse du son dépend du milieu de propagation. Elle est de 1500 m/s dans l'eau et de 5600 m/s dans l'acier.

1.1. Le son

❑ Facteurs qui modifient la vitesse du son

- La vitesse du son dépend de plusieurs facteurs comme la densité, la compressibilité (aptitude à changer de volume sous l'effet de la pression) ou encore la température.
- Au niveau de la mer, la célérité (ou vitesse) du son dans l'air (en m/s) est donnée par :

$$\begin{cases} c = 300 + 0,6T & (T \text{ en } ^\circ\text{C}) \\ c = 20\sqrt{T} & (T \text{ en Kelvin}) \end{cases}$$

1.1. Le son

□ Intensité sonore

- L'intensité sonore, I (en W.m^{-2}), est une grandeur qui permet de donner une indication sur la force (ou débit d'énergie) d'un son : plus l'intensité sonore est élevée et plus le son perçu par l'oreille humaine est fort
- Soit P (en W), la puissance sonore de la source, et S (en m^2) la surface du récepteur, l'intensité sonore a pour expression :

$$I = \frac{P}{S}$$

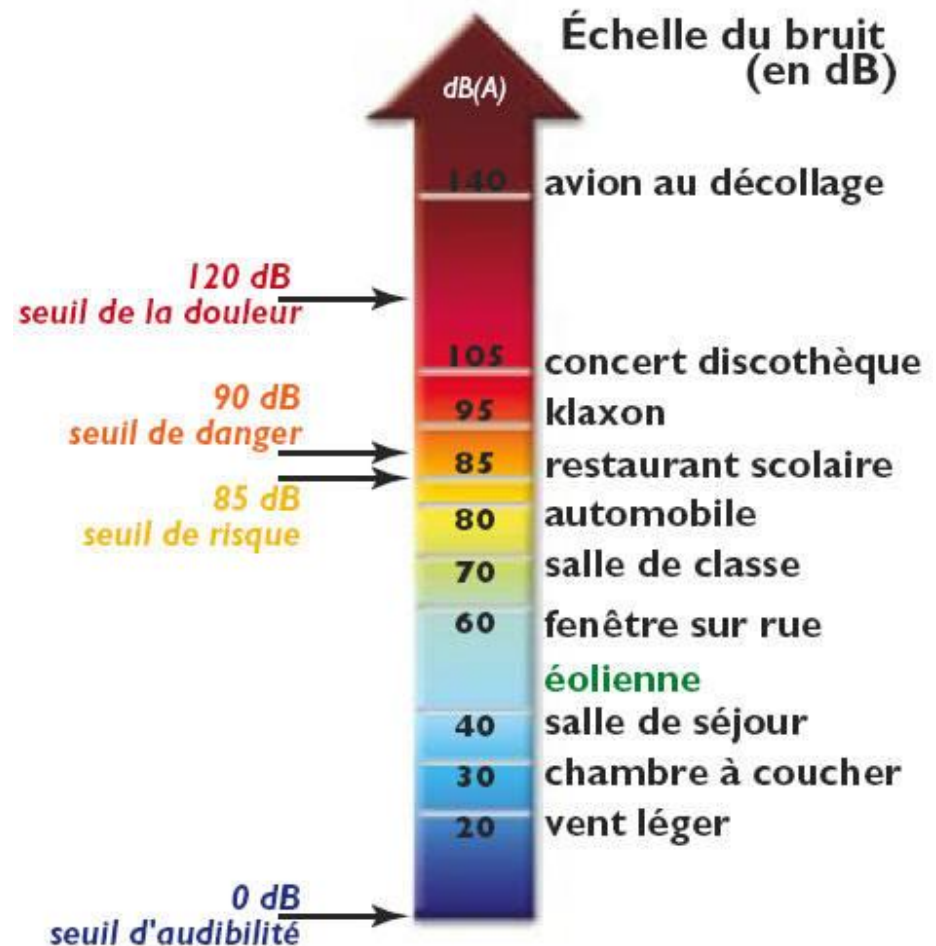
- L'oreille humaine ne peut pas percevoir des sons ayant une intensité sonore inférieure à $I_0 = 10^{-12} \text{ W.m}^{-2}$
- Le seuil de douleur correspond à une valeur d'environ 10 W.m^{-2} .

1.1. Le son

□ Niveau sonore

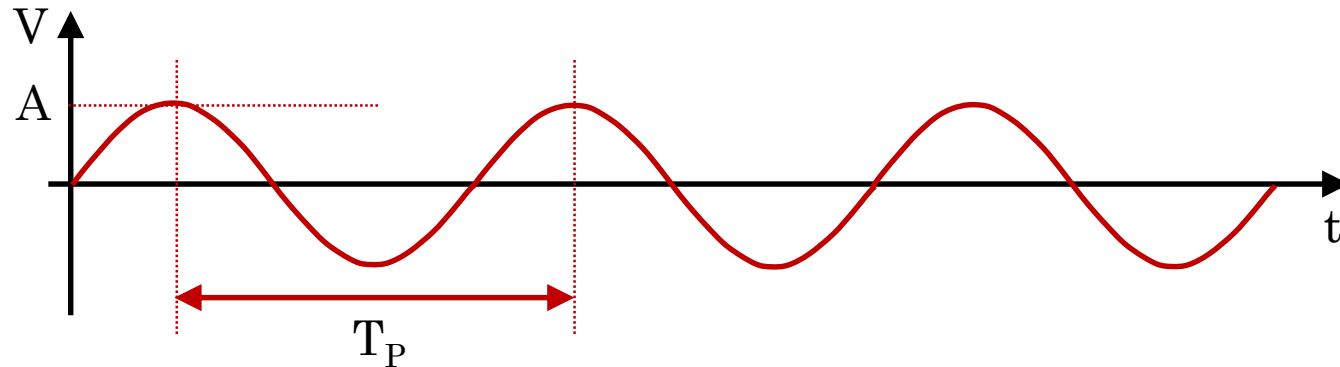
- Le niveau sonore, noté L , est une comparaison entre l'intensité sonore I et l'intensité sonore minimale que l'on peut percevoir, I_0 .
- Il s'exprime en décibel (dB) et correspond à l'équation :

$$L = 10 \cdot \log\left(\frac{I}{I_0}\right)$$

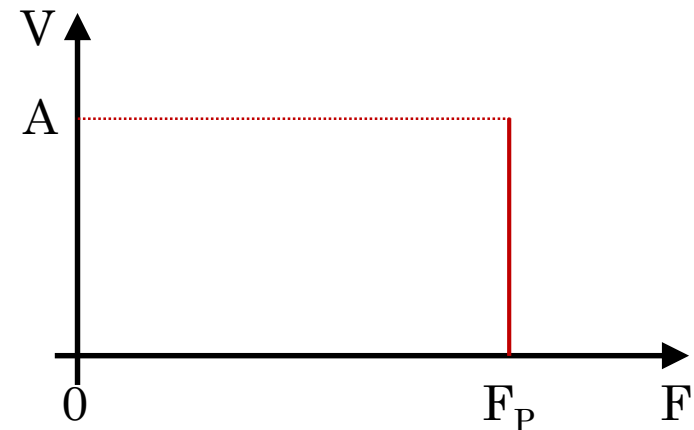


1.2. Notion de fréquence (et de longueur d'onde)

- Soit la note de musique « La3 pure » dont la fréquence est $F_P = 440$ Hz. Ce signal est de forme sinusoïdale de période $T_P = 1/F_P = 1/440 = 22.7$ ms



- La mise en équation de ce signal s'écrit : $V = A.\cos(2\pi F_P t)$
- On peut changer de représentation graphique en ne conservant que la fréquence et l'amplitude du signal



1.2. Notion de fréquence (et de longueur d'onde)

- On suppose à présent que cette note est émise par un haut parleur et se propage dans l'air. Essayons de représenter le positionnement spatial de cette note.
- A $t = 0$, On considère que le maximum de la sinusoïde est émis par le haut parleur.



1.2. Notion de fréquence (et de longueur d'onde)

- On suppose à présent que cette note est émise par un haut parleur et se propage dans l'air. Essayons de représenter le positionnement spatial de cette note.
- A $t = 0$, On considère que le maximum de la sinusoïde est émis par le haut parleur.
- On cherche alors la distance qu'aura parcouru ce maximum quand le prochain maximum sera émis par le haut parleur. Cette distance sera appelée longueur d'onde et notée λ (en m).

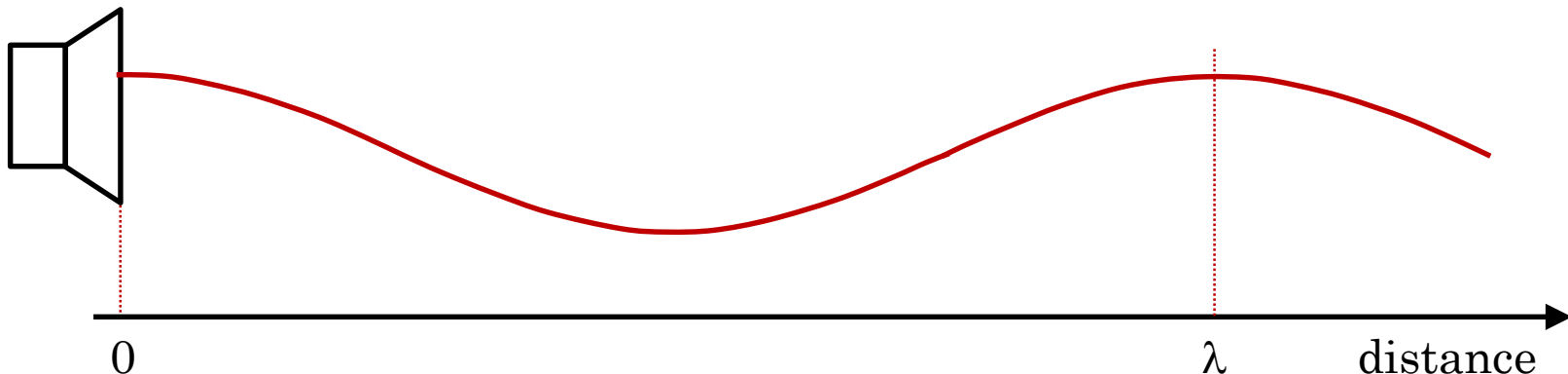


1.2. Notion de fréquence (et de longueur d'onde)

- La période d'apparition du maximum étant T_P et la vitesse du son étant c , la distance parcourue par le maximum est égale à :

$$\lambda = T_P \cdot c = \frac{c}{F_P}$$

- Pour la note considérée, nous obtenons une longueur d'onde de 77.2 cm

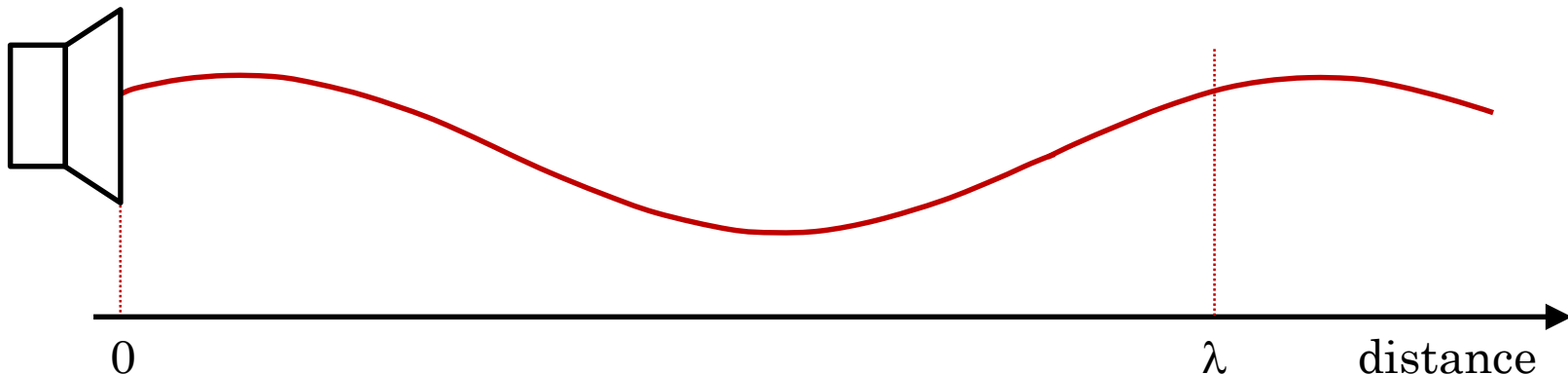


1.2. Notion de fréquence (et de longueur d'onde)

- La période d'apparition du maximum étant T_P et la vitesse du son étant c , la distance parcourue par le maximum est égale à :

$$\lambda = T_P \cdot c = \frac{c}{F_P}$$

- Pour la note considérée, nous obtenons une longueur d'onde de 77.2 cm
- La position des maximums changent évidemment avec le temps

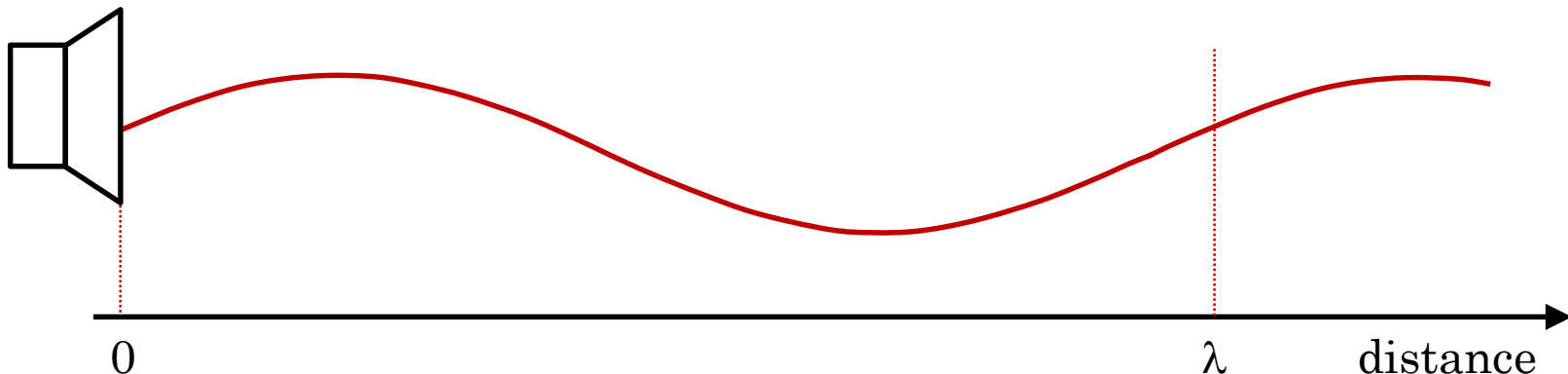


1.2. Notion de fréquence (et de longueur d'onde)

- La période d'apparition du maximum étant T_P et la vitesse du son étant c , la distance parcourue par le maximum est égale à :

$$\lambda = T_P \cdot c = \frac{c}{F_P}$$

- Pour la note considérée, nous obtenons une longueur d'onde de 77.2 cm
- La position des maximums changent évidemment avec le temps

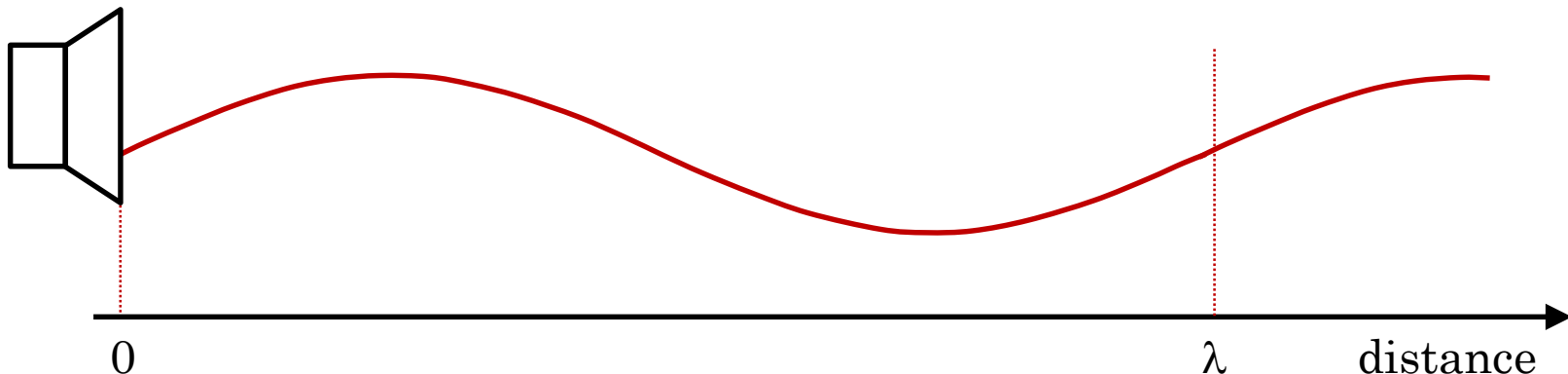


1.2. Notion de fréquence (et de longueur d'onde)

- La période d'apparition du maximum étant T_P et la vitesse du son étant c , la distance parcourue par le maximum est égale à :

$$\lambda = T_P \cdot c = \frac{c}{F_P}$$

- Pour la note considérée, nous obtenons une longueur d'onde de 77.2 cm
- La position des maximums changent évidemment avec le temps
- La longueur d'onde de cette note dans l'eau est de 3.4 m

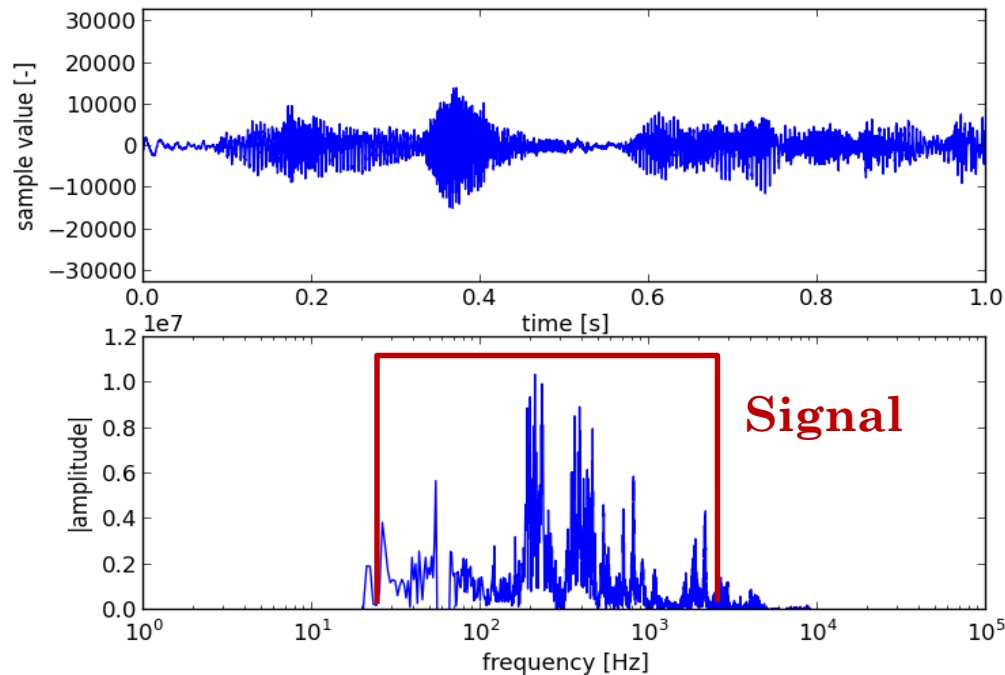


1.3. Composition d'un son

□ Décomposition en série de Fourier

- La voix humaine (et les autres sons) est constituée d'une somme de sinusôides d'amplitudes et fréquences différentes :

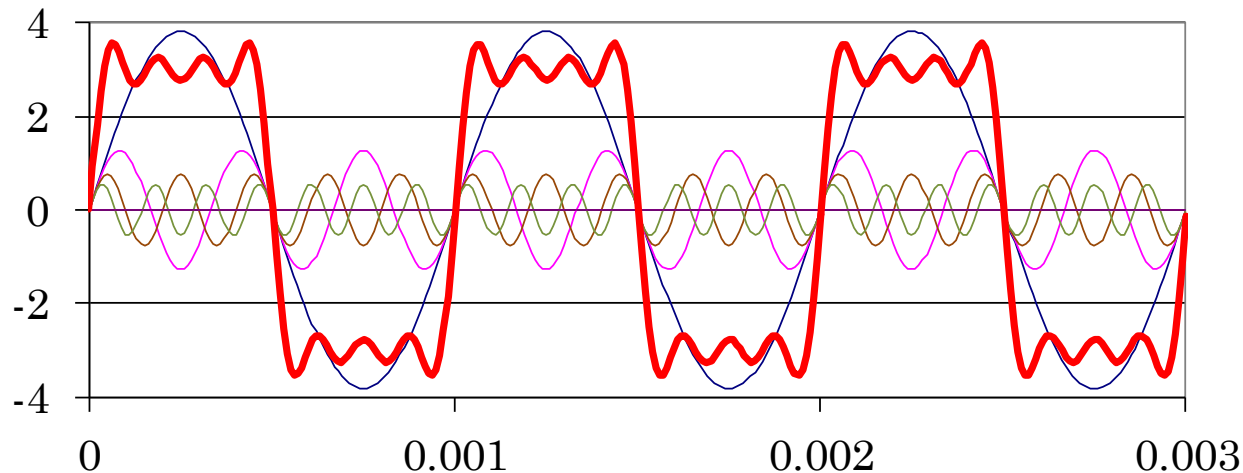
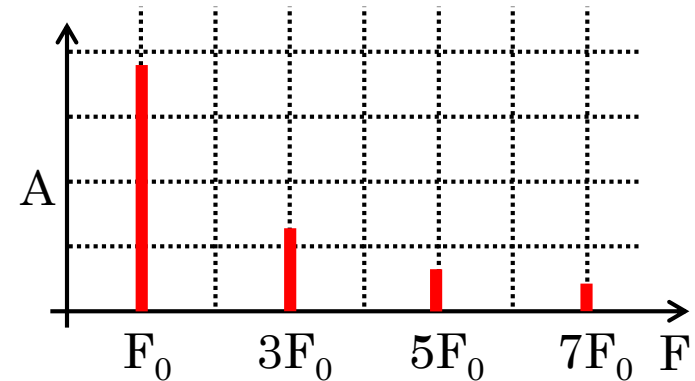
$$x(t) = a_0 + \sum_{n=1}^{\infty} \left(a_n \cdot \cos\left(2\pi \frac{n}{T} t\right) + b_n \cdot \sin\left(2\pi \frac{n}{T} t\right) \right)$$



1.3. Composition d'un son

□ Décomposition en série de Fourier

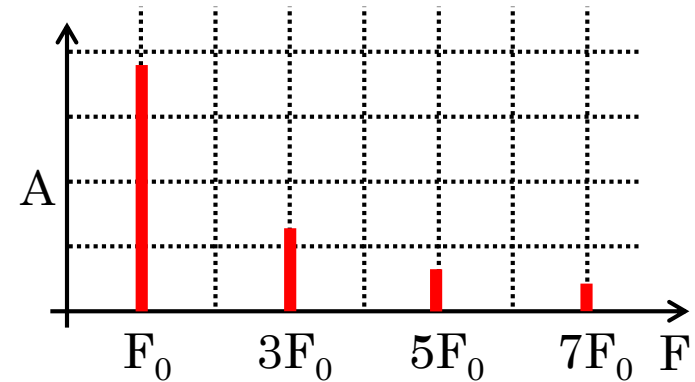
- Un signal carré se décompose en somme de sinusoïdes d'amplitudes et de fréquences différentes



1.3. Composition d'un son

□ Décomposition en série de Fourier

- Un signal carré se décompose en somme de sinusôides d'amplitudes et de fréquences différentes
- La fréquence la plus basse notée F_0 s'appelle le fondamental. Sa valeur est de 440 Hz pour la note « LA3 »
- A noter que ce fondamental s'appelle aussi « la hauteur du son » (ou hauteur de la note)
- Les autres fréquences $3F_0$, $5F_0$... s'appellent les harmoniques



1.3. Composition d'un son

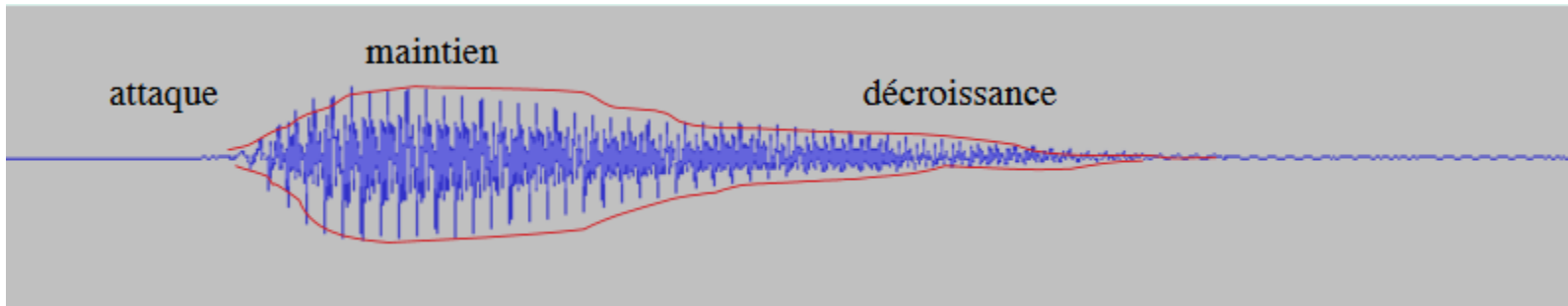
□ Musicalité

- A une note « LA3 » à 440 Hz, on peut associer une multitude d'harmoniques
- C'est cette diversité qui nous permet de reconnaître le « LA3 » d'un violon de celui d'un piano ou encore d'une cornemuse
- Cette diversité s'exprime aussi au sein d'une même famille d'instruments et permet de différencier un violon en contreplaqué d'un Stradivarius.

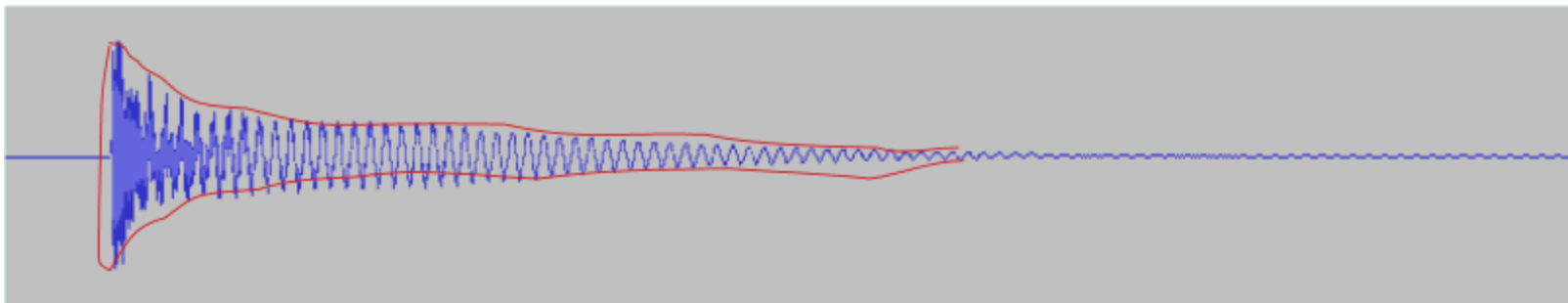
1.3. Composition d'un son

□ Enveloppe d'un son

- L'enveloppe d'un son est constituée de 3 phases, enfin quand elles sont identifiables



Clarinette

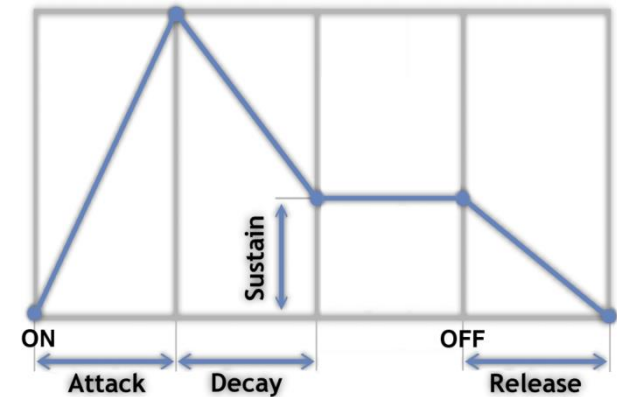


Xylophone

1.3. Composition d'un son

□ Enveloppe d'un son

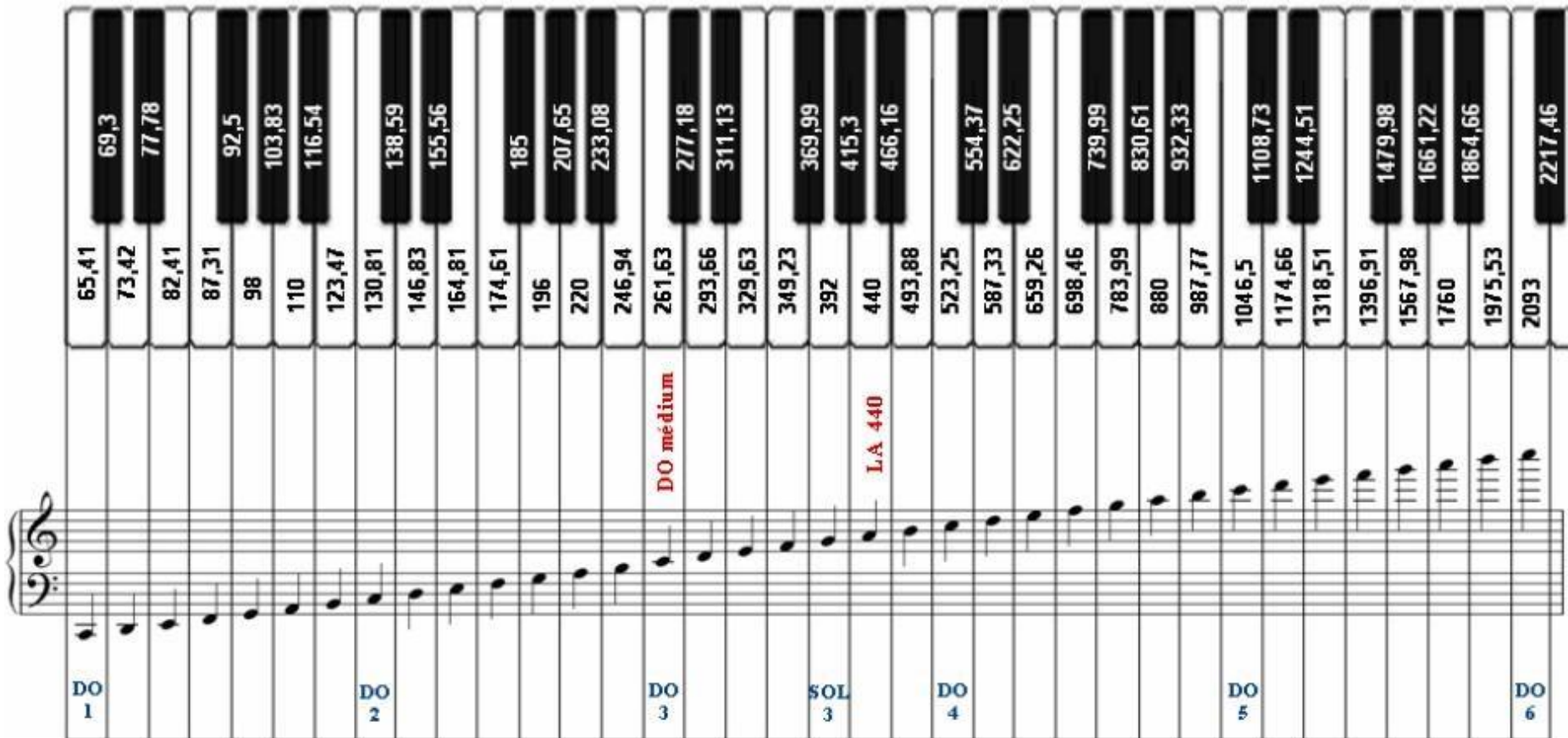
- L'enveloppe la plus courante dans les synthétiseurs est l'enveloppe Attack Decay Sustain Release (Attaque Chute Entretien Extinction en français). Synchronisée avec le clavier, elle permet de moduler le son de chaque note à l'aide de quatre paramètres.



1.3. Composition d'un son

□ Quelques notes de musique

- Voici les notes et les fréquences sur 5 octaves d'un piano

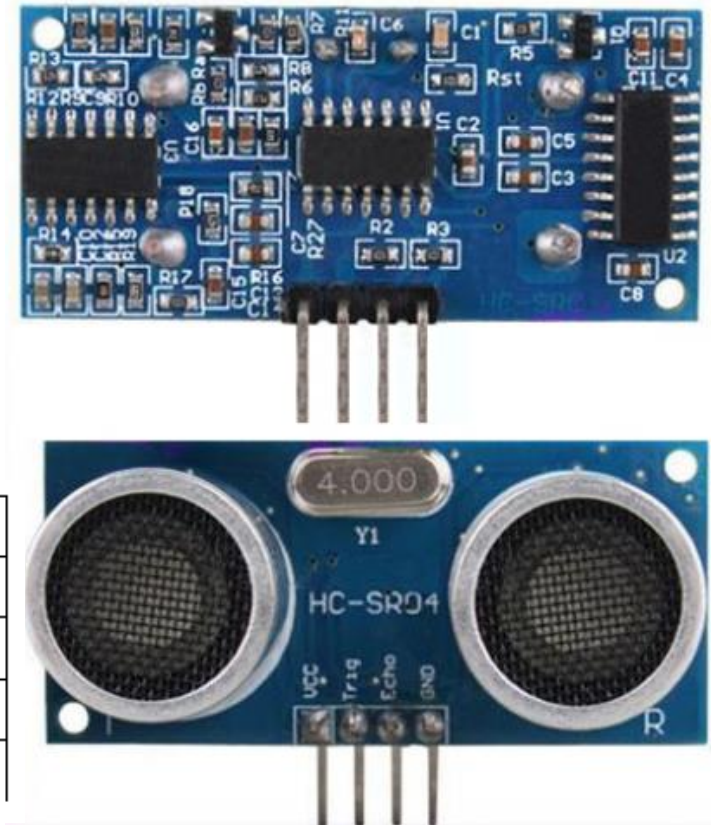


1.4. Présentation du module HC-SR04

□ Matériel

- Il existe plusieurs modules qui permettent de mesurer une distance et nous utiliserons le HC-SR04.
- Il comprend 2 haut-parleurs et l'onde utilisée a une fréquence de 40 kHz.
- L'électronique de gestion des haut-parleurs et de mesure de la distance est déjà intégrée au module

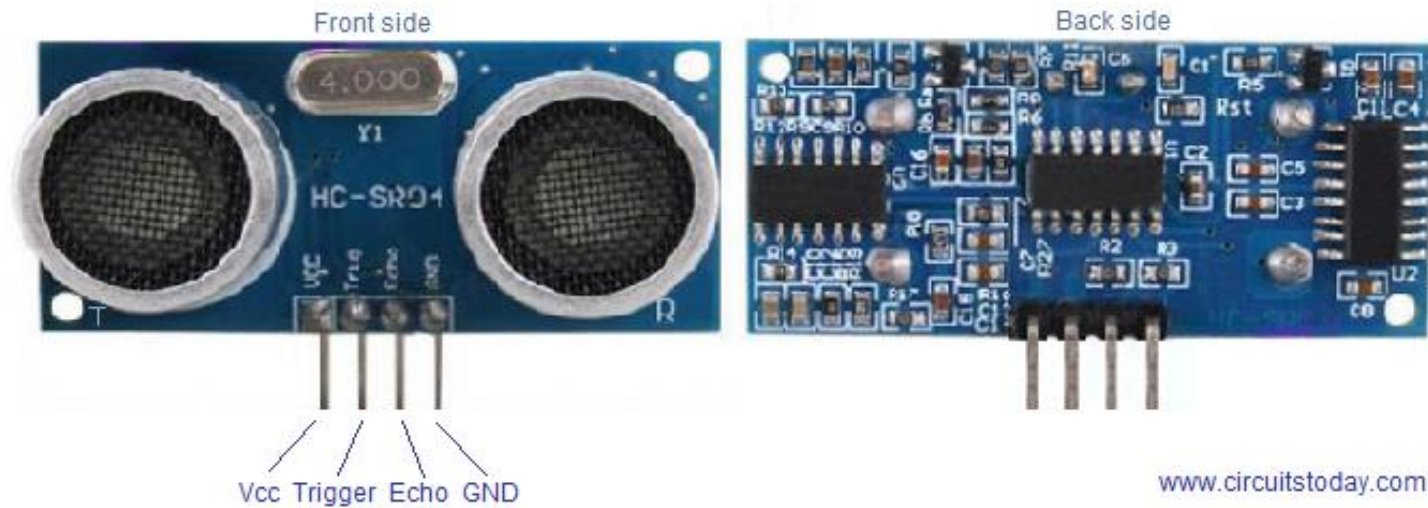
Parameter	Min	Typ.	Max	Unit
Operating Voltage	4.50	5.0	5.5	V
Quiescent Current	1.5	2	2.5	mA
Working Current	10	15	20	mA
Ultrasonic Frequency	-	40	-	kHz



1.4. Présentation du module HC-SR04

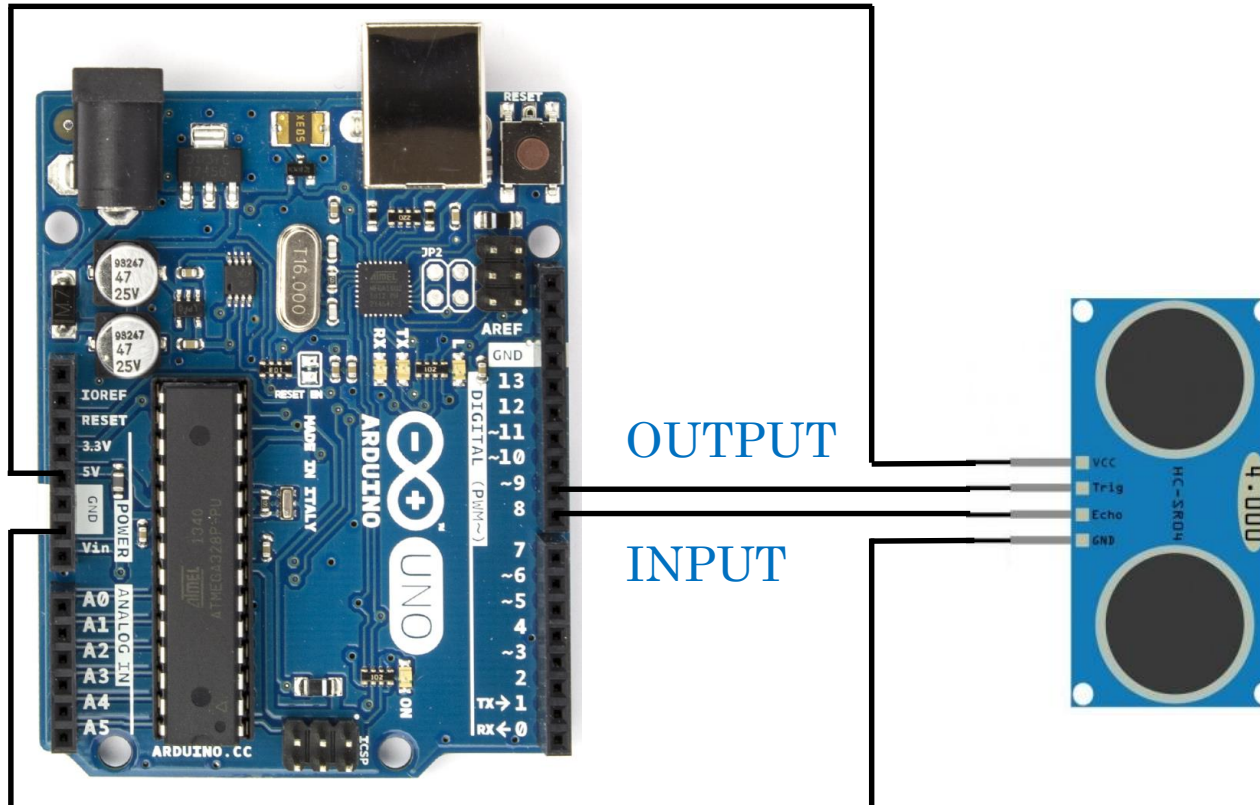
□ Matériel

- Il y a 3 entrées dans le module : VCC, GND et Trig
- Il y a 1 sortie : Echo



1.4. Présentation du module HC-SR04

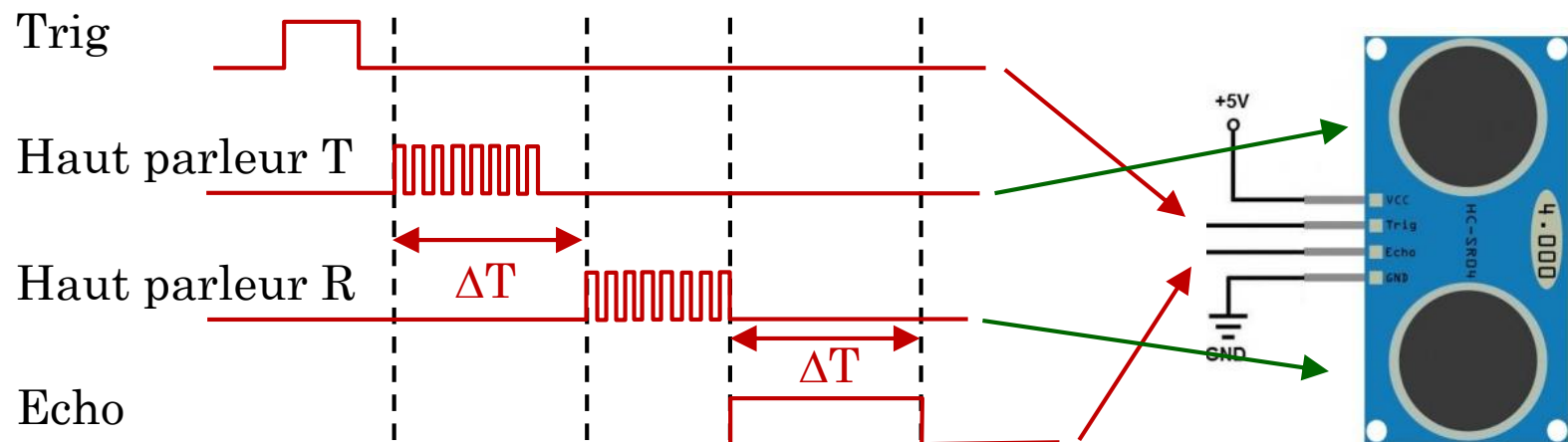
❑ Exemple de montage



1.4. Présentation du module HC-SR04

□ Fonctionnement

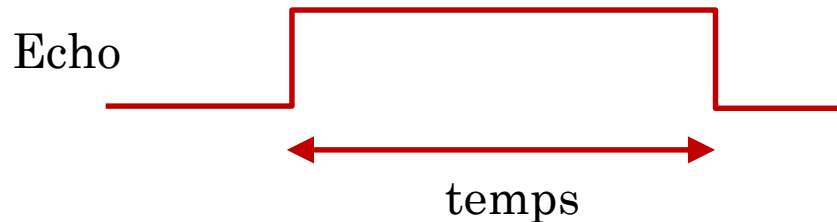
- Pour déclencher une mesure il faut appliquer une impulsion d'au moins 10 μ s sur l'entrée Trig
- Le module envoie alors 8 impulsions sonores
- On obtient en sortie (sur Echo) une impulsion dont la longueur correspond au temps que mettent les 8 impulsions pour faire l'aller/retour
- Il est préférable d'attendre 60 ms avant de demander une nouvelle mesure



1.4. Présentation du module HC-SR04

□ Détermination de la distance

- En une 1 s, le son parcourt 340 m donc en 1 μ s il parcourt 0.034 cm



- On en déduit la distance en cm :

$$\text{distance} = \frac{\text{temps}}{2} 0.034 = 0.017 \cdot \text{temps}$$

1.5. Les nouvelles fonctions

- **delayMicroseconds(X)** : permet de bloquer le déroulement du programme durant $X \mu s$
- **pulseIn(X,Y)** : retourne la durée d'une impulsion Y sur l'I/O X. Le temps retourné est en μs et va de $10 \mu s$ à 3 min. Si $Y = \text{HIGH}$ alors on attend une impulsion **LOW-HIGH-LOW**. Si $Y = \text{LOW}$, on attend une impulsion **HIGH-LOW-HIGH**. Le temps d'attente pour l'apparition de l'impulsion est de 1s. Au-delà de ce temps, la fonction retourne la valeur 0.
- **pulseIn(X,Y,Z)** : Z permet de modifier le temps (en μs) d'attente de l'apparition de l'impulsion. Si le temps d'attente est dépassé, la valeur retournée est 0

```
digitalWrite(trig, HIGH); // génération de l'impulsion Trig de 10  $\mu s$   
delayMicroseconds(10);  
digitalWrite(trig, LOW);  
lecture_echo = pulseIn(echo, HIGH); // lecture de la longueur temporelle de l'impulsion Echo
```

1.6. Librairie du HC-SR04

- Des personnes trouvent que les librairies de certains modules ne sont pas assez performantes et décident de faire mieux. C'est le cas pour le module HC-SR04 (dont nous avons utilisé la librairie arduino)
- Cette librairie s'appelle « NewPing » et il faut installer son répertoire dans le répertoire « librairie » du logiciel arduino
- Cette librairie s'accompagne de nouvelles fonctions

□ Nouvelles fonctions

- **NewPing sonar(X,Y,Z)** : initialise l'utilisation du sonar. X = trigger, Y = echo, Z est la distance maximale à mesurer (en cm). Il n'est pas obligatoire de donner une valeur pour Z qui par défaut est de 500 cm

```
NewPing sonar(trig, echo, 200);
```

- **sonar.ping_cm()** : donne la distance en cm ou 0 si il n'y a pas d'écho

```
cm=sonar.ping_cm();
```


1.6. Librairie du HC-SR04

- Il est possible d'envoyer un ordre ou plus globalement des données à la carte arduino via le port série.
- Ces données sont stockées dans le buffer série de la carte arduino avant d'être lues. Elles peuvent être envoyées par le moniteur série du logiciel

□ Nouvelles fonctions

- **Serial.read()** : lit le premier caractère du buffer et renvoi le caractère -1 s'il est vide

```
Serial.println("Souhaitez-vous la valeur en cm de la distance Y/N ?");
```

```
.....
```

```
test = Serial.read(); //on lit le 1er caractère du buffer
```

```
if (test == 'Y') {
```

```
    Serial.print(cm); //envoi de la distance à l'ordinateur
```

```
    Serial.println(" cm");}
```

- **Serial.available()** : permet d'obtenir le nombre de caractères dans le buffer

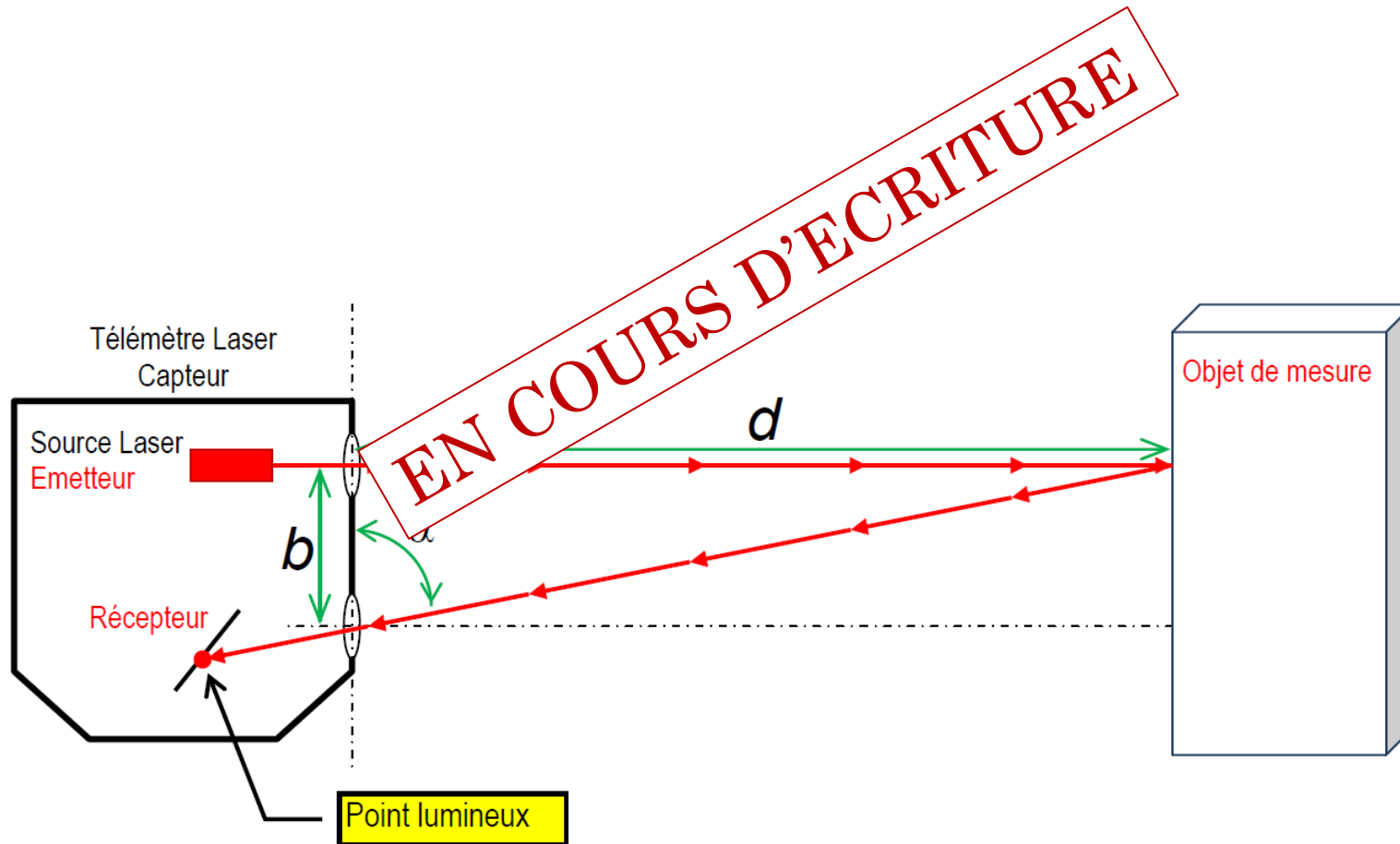
2.1. Méthodes de mesure

- Temps de vol

EN COURS D'ECRITURE

2.1. Méthodes de mesure

□ Mesure d'un angle



2.2. Le module VL53L0X

□ Présentation

- Il permet de mesurer des distances jusqu'à 40 m avec une précision de 1 cm



EN COURS D'ECRITURE

2.3. Le module TFmini

□ Présentation

- Il permet de mesurer des distances jusqu'à 40 m avec une précision de 1 cm



EN COURS D'ECRITURE

2.4. Le module lidar lite 3

□ Présentation

- Il permet de mesurer des distances jusqu'à 40 m avec une précision de 1 cm
- Il peut effectuer jusqu'à 500 mesures par seconde (i.e. 500 Hz) et 1 kHz pour le LLV3HP
- Laser de 940 nm de classe 1
- Il est commandable par bus I²C ou en PWM
- Il est utilisé pour les drones, la robotique générale, la détection industrielle ...



Lidar lite 3

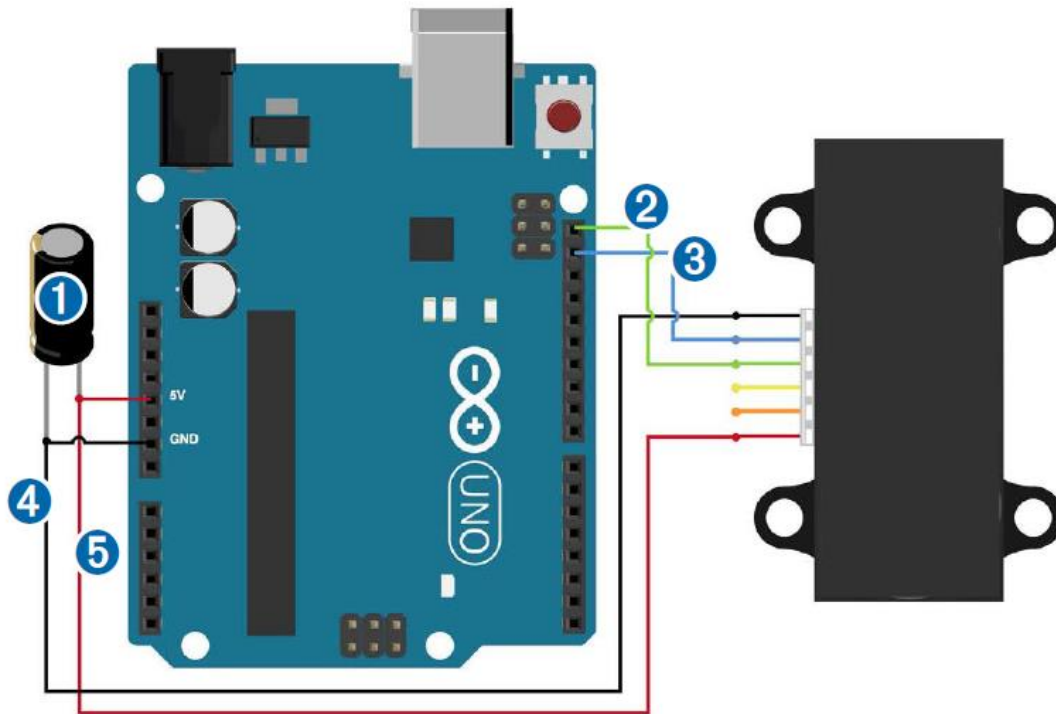


LIDAR-Lite 3 (LLV3HP)

2.4. Le module lidar lite 3

□ Montage avec I²C

- Un condensateur permet de filtrer les appels de courant du lidar (capacité de découplage)



2.4. Le module lidar lite 3

□ La librairie LIDARlite.h

- Cette librairie s'occupe de tout le protocole de communication avec le lidar

```
#include <Wire.h>
#include <LIDARLite.h>
// Globals
LIDARLite lidarLite;
int cal_cnt = 0;
void setup()
{
  Serial.begin(9600); // Initialize serial connection to display distance readings
  lidarLite.begin(0, true); // Set configuration to default and I2C to 400 kHz
  lidarLite.configure(0); // Change this number to try out alternate configurations
}
```


2.4. Le module lidar lite 3

❑ La librairie LIDARlite.h

- Toutes les 100 mesures, il faut effectuer une calibration du lidar

```
void loop()
{
  int dist;
  // At the beginning of every 100 readings,
  // take a measurement with receiver bias correction
  if ( cal_cnt == 0 ) {
    dist = lidarLite.distance(); // With bias correction
  } else {
    dist = lidarLite.distance(false); // Without bias correction
  }
  // Increment reading counter
  cal_cnt++;
  cal_cnt = cal_cnt % 100;
  // Display distance
  Serial.print(dist);
  Serial.println(" cm");
  delay(10);
}
```

3.1. Principe de fonctionnement

- Une LED émet de la lumière qui est réfléchié ou non sur un phototransistor

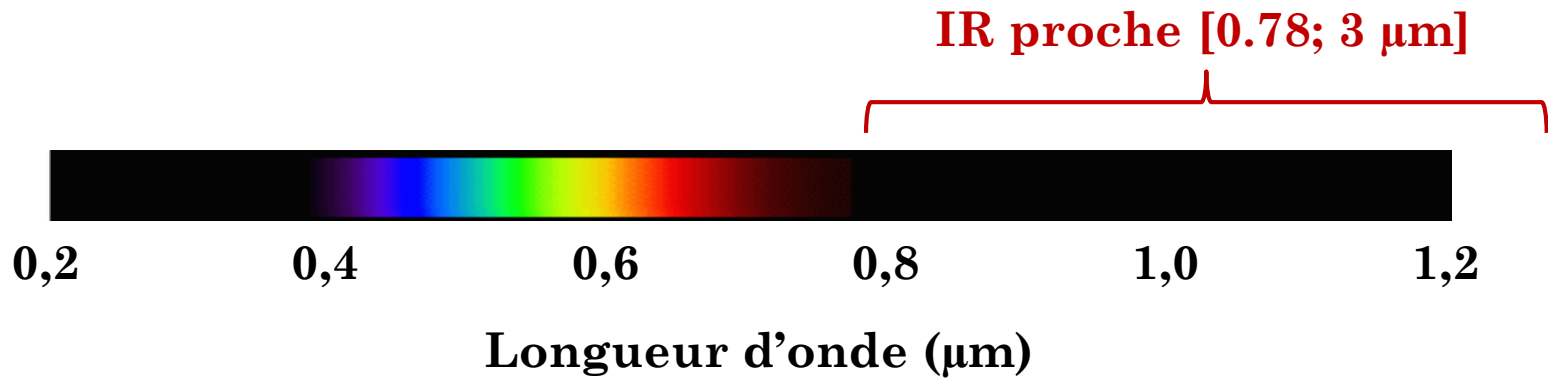


- L'obstacle doit être assez proche du détecteur parfois moins d'1 cm
- L'obstacle peut aussi être une ligne à suivre

3.2. Diodes et phototransistors IR

□ Diodes IR

- Les LED IR émettent dans l'Infra Rouge (infrarouge) qui est une longueur d'onde qui s'étale de 0.78 μm à 5 mm



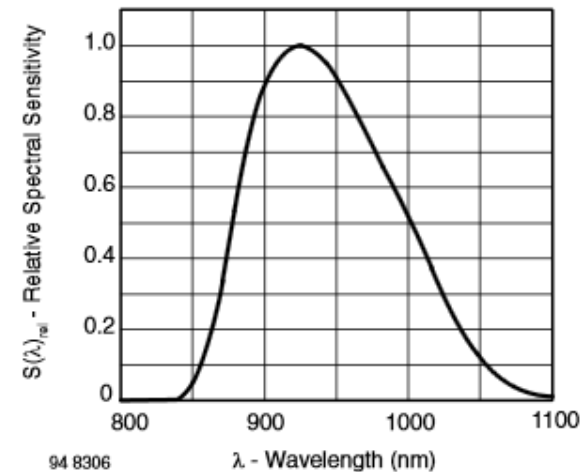
- Il est évident que l'œil humain ne peut pas faire la différence entre une LED allumée et éteinte



3.2. Diodes et phototransistors IR

■ Phototransistors IR

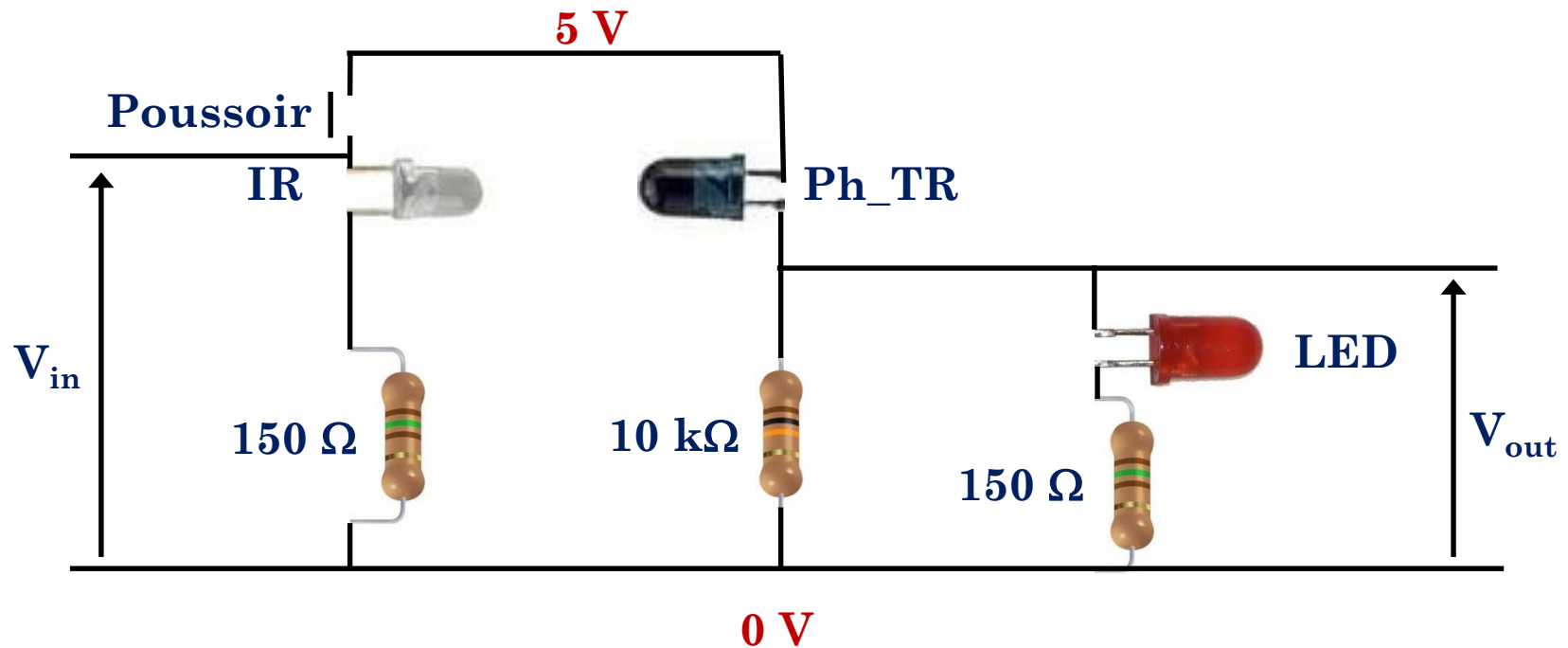
- Les phototransistors sont des transistors bipolaires dont la base est sensible à la lumière.
- Dans le cas des phototransistors IR, la base n'est sensible qu'à (ou presque) une longueur d'onde
- La couleur du boîtier évite de « polluer » le phototransistor avec les autres longueurs d'onde



3.3. Réalisation d'un détecteur IR

□ Montage de base

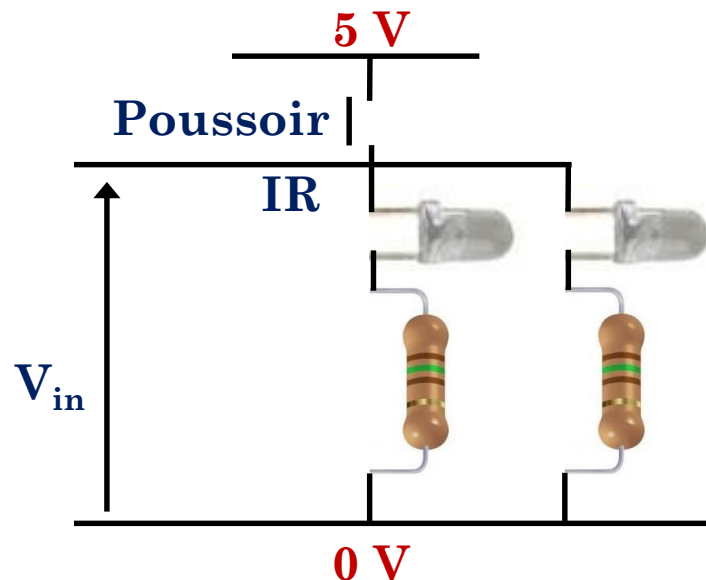
- Pour réaliser le détecteur, on peut commencer par un montage simple qui ne fait pas intervenir l'arduino (sauf pour l'alimentation)
- Si $V_{in} = "1"$ alors $V_{out} = "1"$ et si $V_{in} = "0"$ alors $V_{out} = "0"$



3.3. Réalisation d'un détecteur IR

□ Amélioration du montage

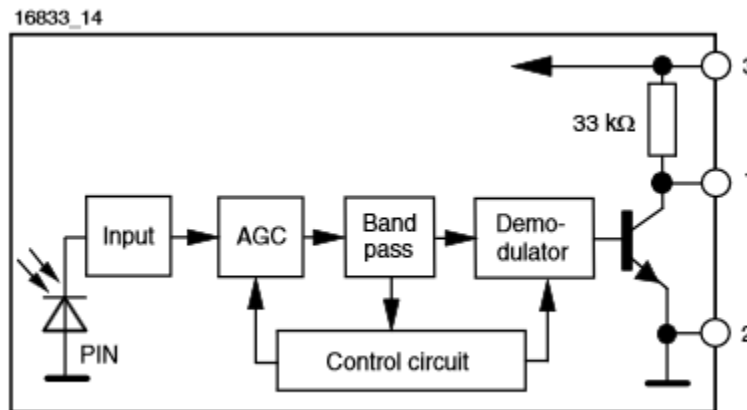
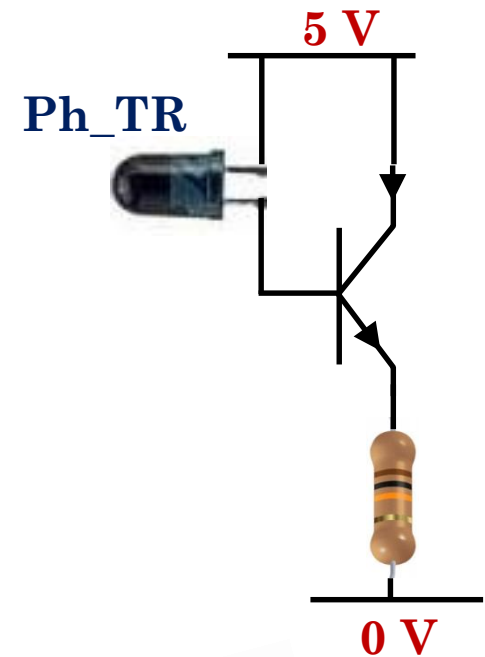
- Plus la diode IR s'éloigne, moins le phototransistor reçoit de photons. A partir d'une certaine distance la valeur de V_{out} n'est plus suffisamment grande pour être comprise comme un "1" logique
- On peut augmenter le nombre de photons en utilisant plusieurs LED IR



3.3. Réalisation d'un détecteur IR

□ Amélioration du montage

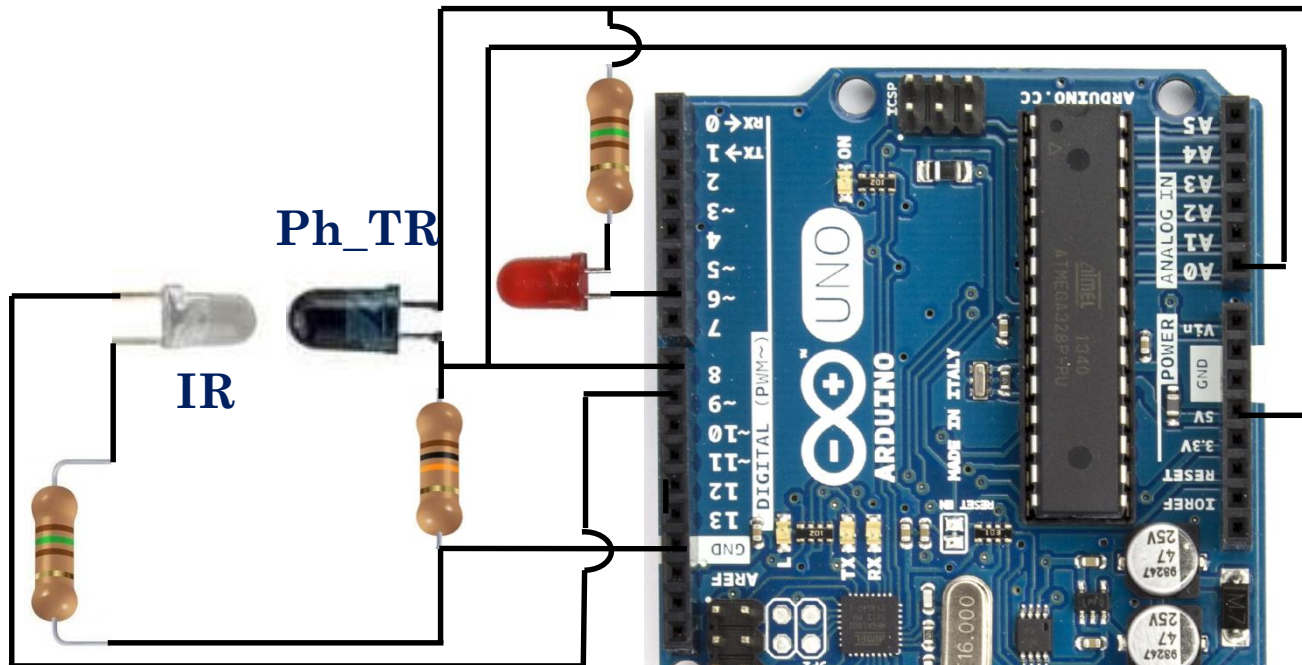
- On peut aussi augmenter le courant qui sort du phototransistor avec un montage Darlington
- Sur ce principe, il existe des composants qui intègrent déjà des amplificateurs.



3.3. Réalisation d'un détecteur IR

□ Montage final

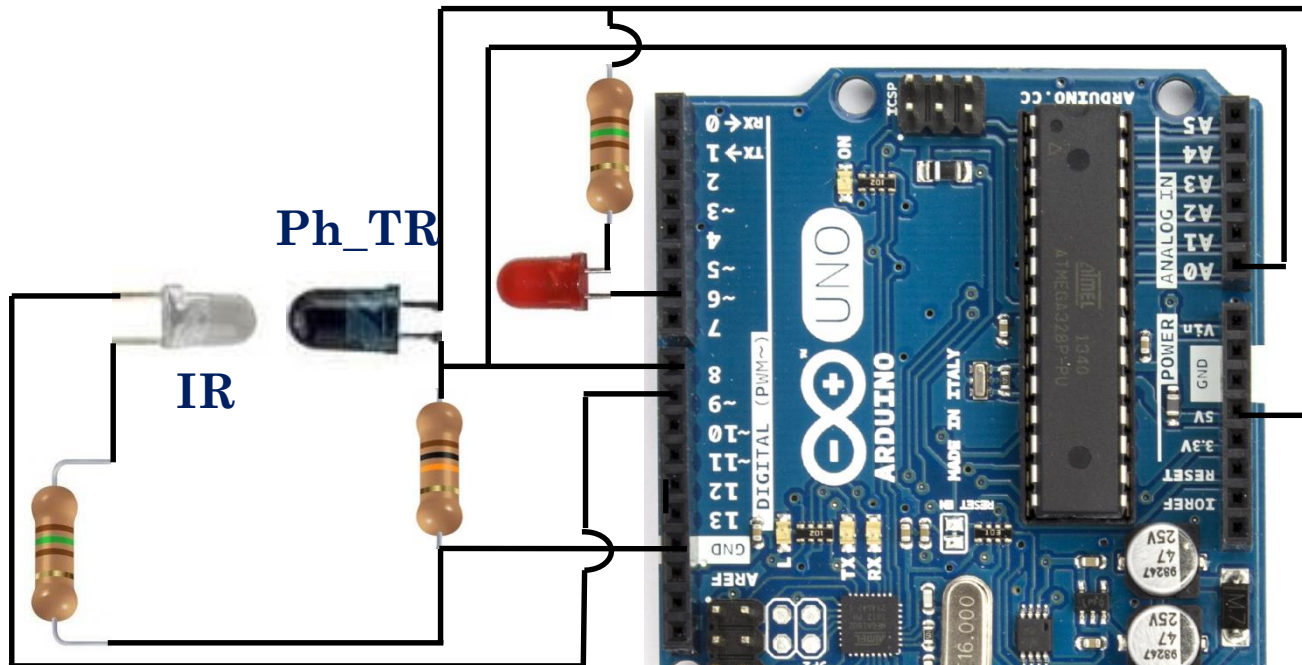
- Pour atteindre plus facilement les 5 V à la réception, on retire la LED rouge
- On peut alors définir l'I/O n°9 pour la transmission et l'I/O n°8 pour la réception.



3.3. Réalisation d'un détecteur IR

□ Montage final

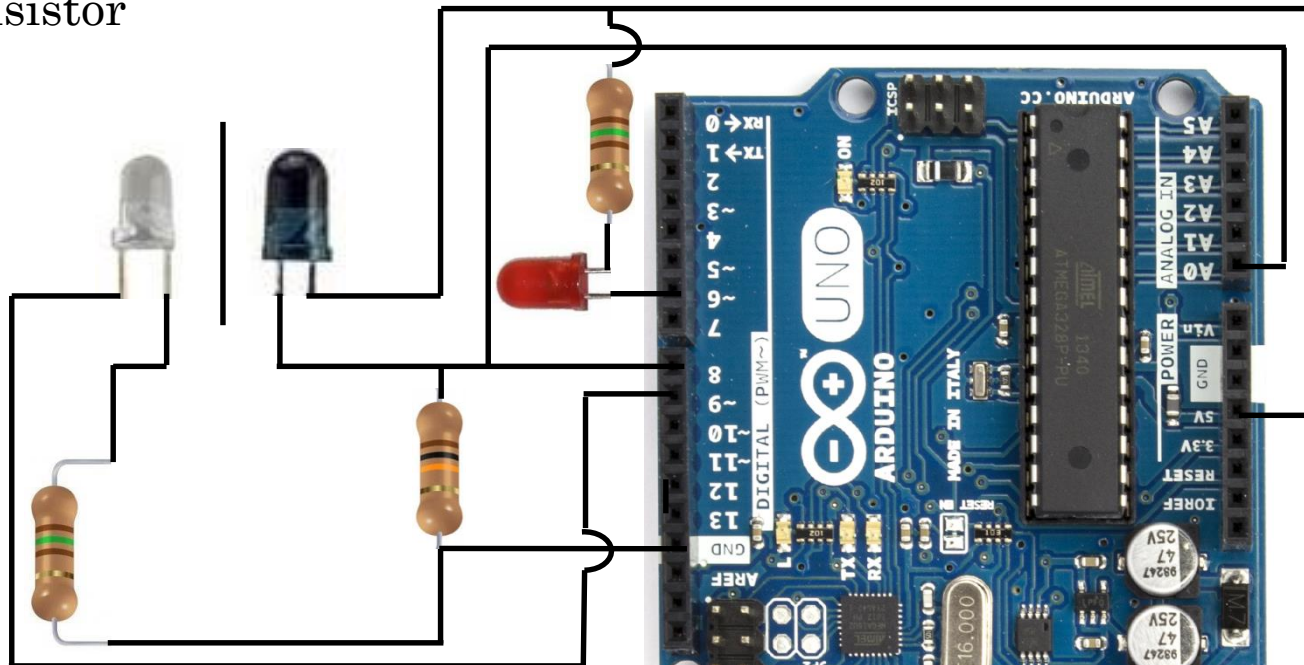
- La LED rouge sur l'IO n°6 indique juste si le phototransistor reçoit une information



3.3. Réalisation d'un détecteur IR

□ Montage final

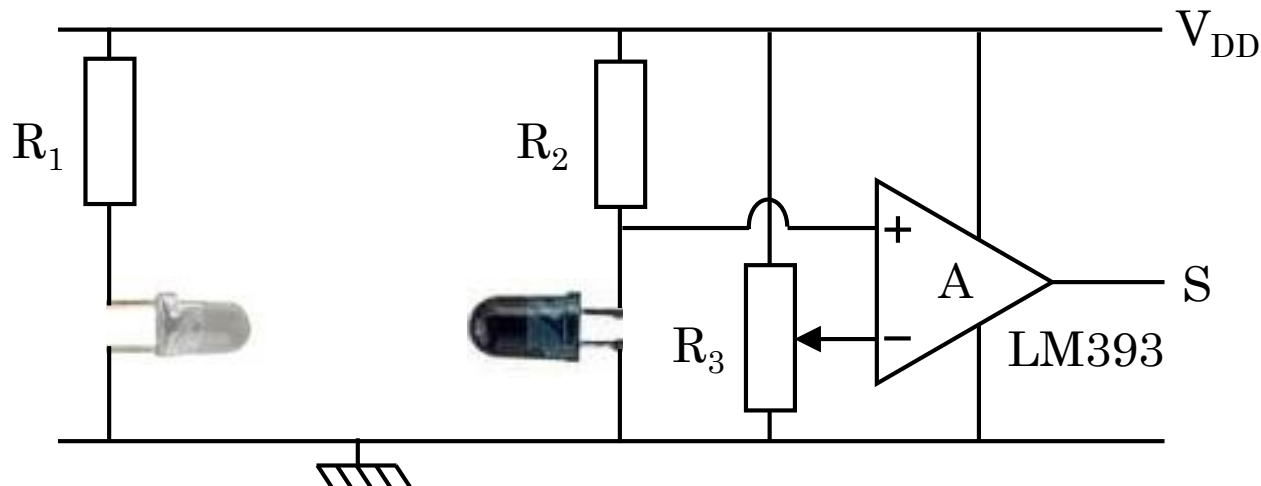
- La LED rouge sur l'IO n°6 indique juste si le phototransistor reçoit une information
- Il reste à orienter la diode et le phototransistor correctement et à ajouter une petite plaque pour éviter que des photons aillent directement sur le phototransistor



3.4. Le module Keyes IR-01

□ Description du module

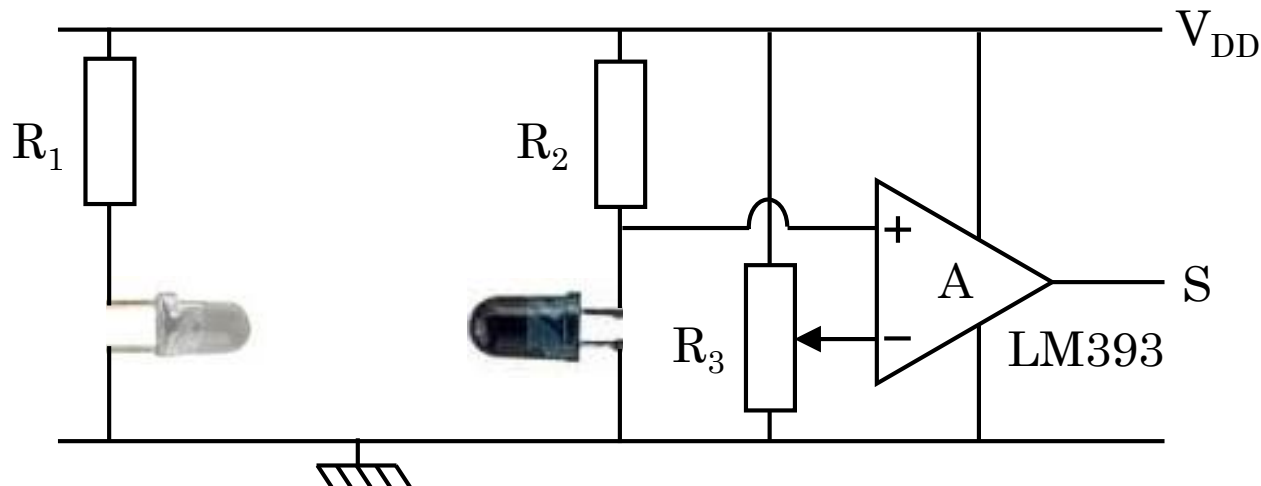
- Il existe des modules qui intègrent déjà la diode IR et le phototransistor avec l'orientation adéquate comme le Keyes IR-01
- Le principe de base de ce module repose sur un comparateur de tension réalisé avec l'AOP LM393.



3.4. Le module Keyes IR-01

□ Description du module

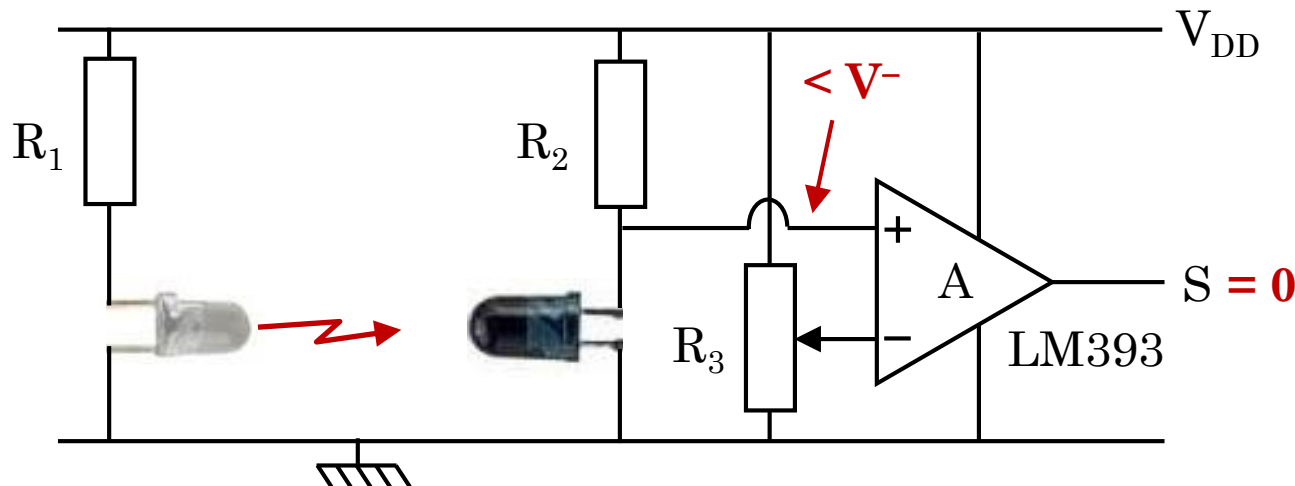
- Il existe des modules qui intègrent déjà la diode IR et le phototransistor avec l'orientation adéquate comme le Keyes IR-01
- Le principe de base de ce module repose sur un comparateur de tension réalisé avec l'AOP LM393.
- La tension de référence sur l'entrée – (V^-) du comparateur peut être ajustée avec le potentiomètre R_3 .



3.4. Le module Keyes IR-01

□ Fonctionnement

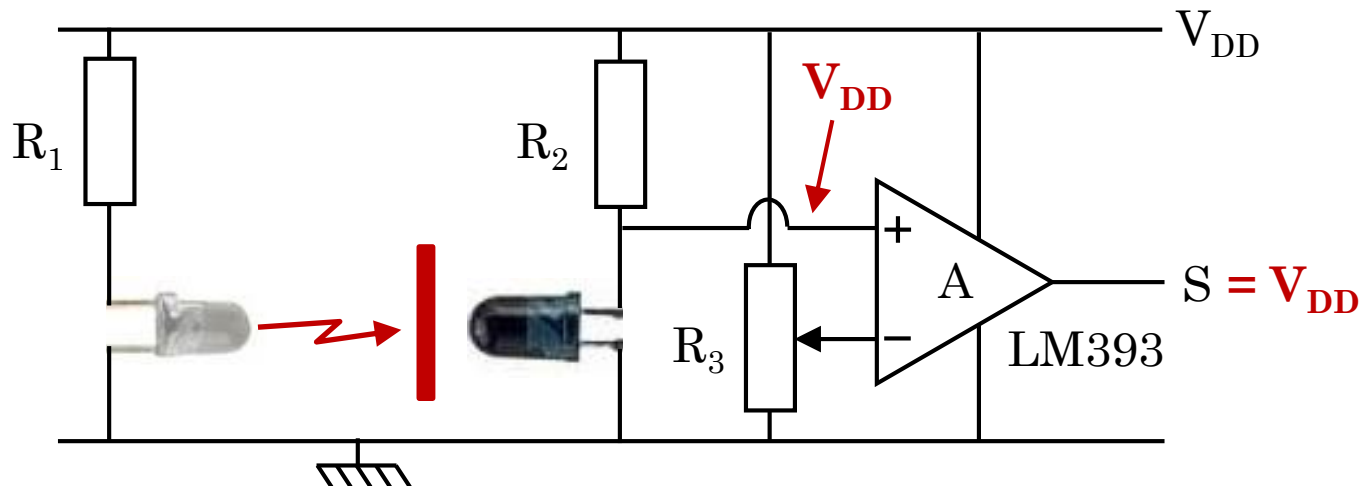
- En présence de lumière, le photo transistor est passant (même proche de la saturation en fonction du nombre de photons) et la tension V^+ devient inférieure à V^- . La tension en sortie est égale à 0



3.4. Le module Keyes IR-01

□ Fonctionnement

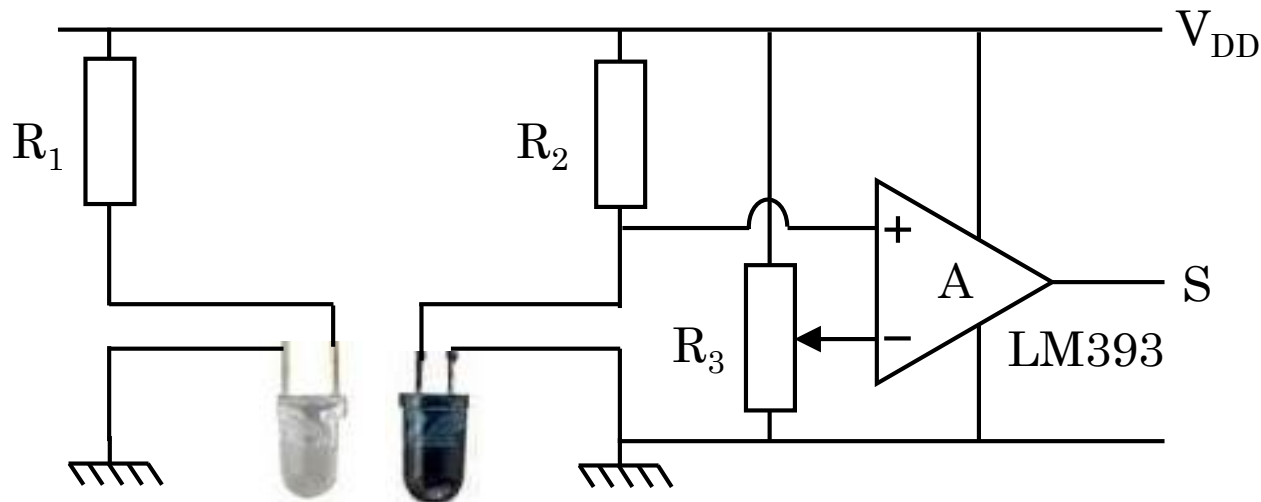
- Si il n'y a pas de lumière, le photo transistor est bloqué (i.e. ne laisse pas passer de courant) et $V^+ = V_{DD}$ donc supérieure à V^- . La tension en sortie est égale à V_{DD}



3.4. Le module Keyes IR-01

□ Fonctionnement

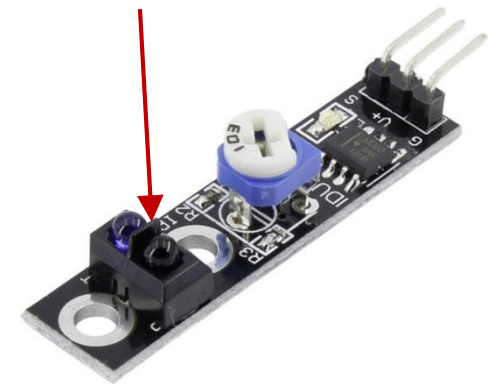
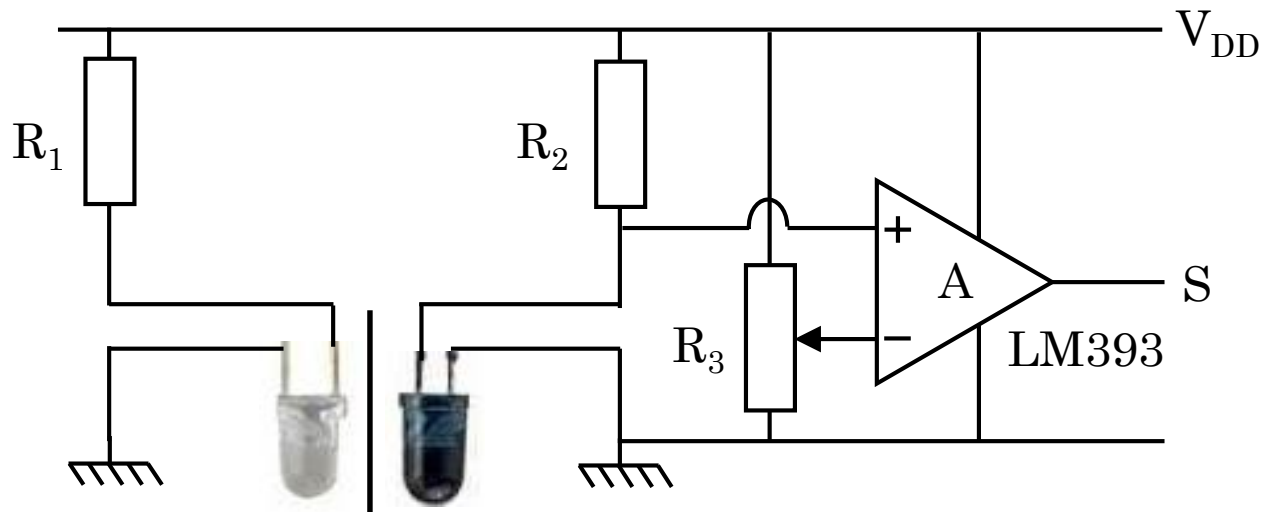
- Pour le module Keyes IR-01, la diode et le phototransistor ne sont pas en face l'un de l'autre mais côte à côte



3.4. Le module Keyes IR-01

□ Fonctionnement

- Pour le module Keyes IR-01, la diode et le phototransistor ne sont pas en face l'un de l'autre mais côte à côte
- Un élément de plastique prévient toute diffusion latéral des photons vers le phototransistor



3.5. Comment détecter une ligne

□ Principe de fonctionnement

- La lumière émise par la diode est réfléchiée par le sol et reçue par le photo transistor



3.5. Comment détecter une ligne

□ Principe de fonctionnement

- La lumière émise par la diode est réfléchiée par le sol et reçue par le phototransistor
- La lumière émise par la diode est absorbée par la ligne noire. Le phototransistor ne reçoit pas de lumière.



3.5. Comment détecter une ligne

□ Principe de fonctionnement

- La lumière émise par la diode est réfléchiée par le sol et reçue par le phototransistor
- La lumière émise par la diode est absorbée par la ligne noire. Le phototransistor ne reçoit pas de lumière.
- La lumière émise par la diode est réfléchiée par le sol et reçue par le phototransistor



3.5. Comment détecter une ligne

□ Principe de fonctionnement

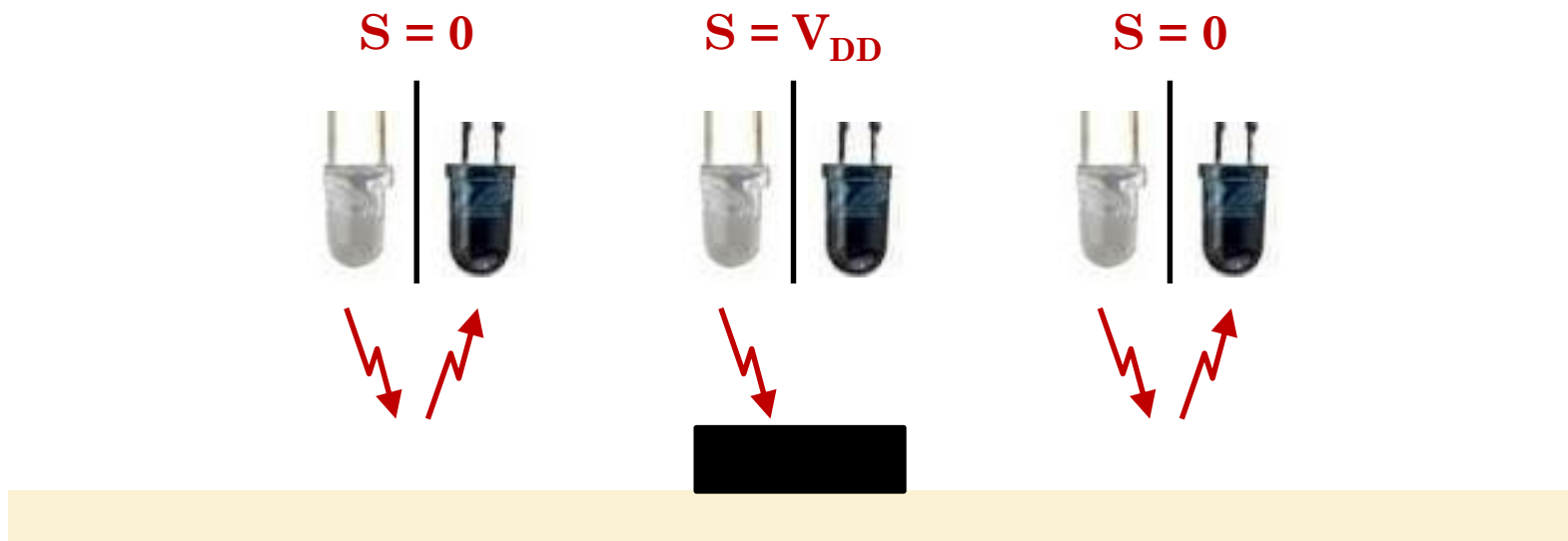
- Cette animation permet de visualiser ce qu'il faut détecter avec la carte arduino pour suivre la ligne



3.5. Comment détecter une ligne

□ Sens de croisement de la ligne

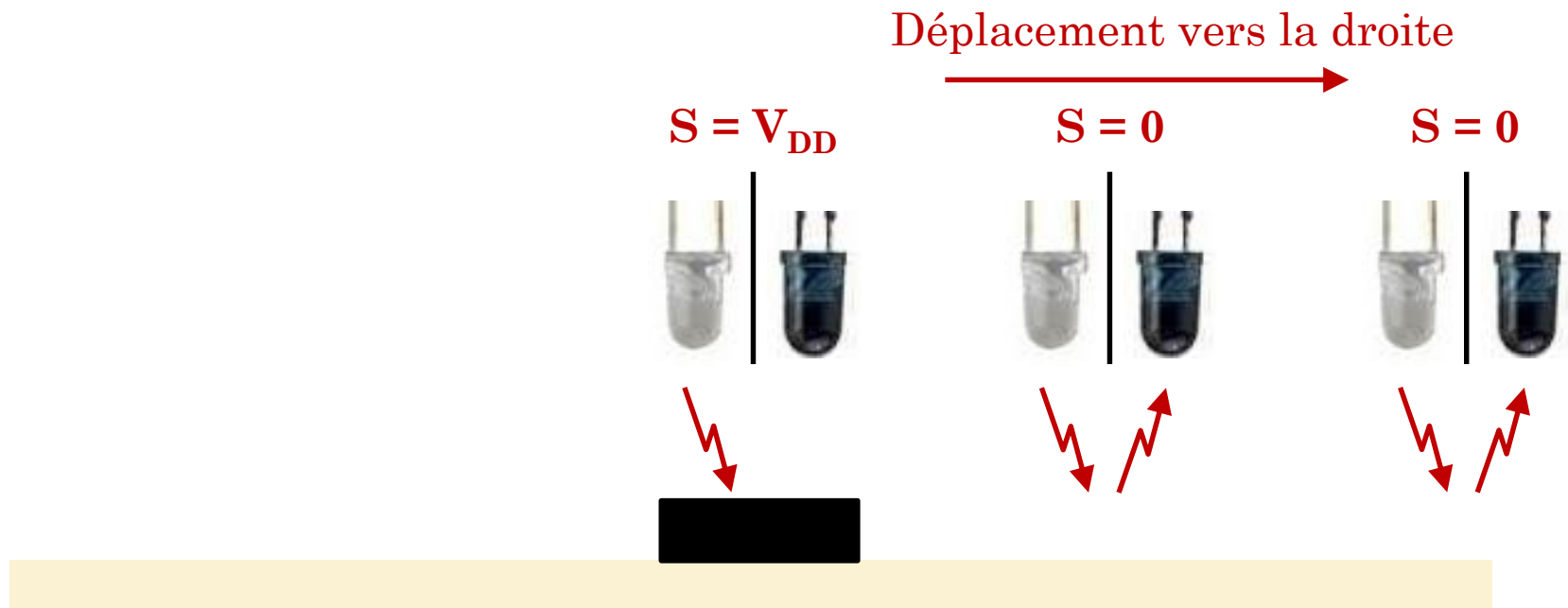
- L'information « être sur la ligne ou non » n'est pas suffisante
- Il faut aussi savoir de quel côté elle a été traversée
- Pour cela on peut utiliser 3 modules et déterminer celui qui se trouve au dessus de la ligne



3.5. Comment détecter une ligne

□ Sens de croisement de la ligne

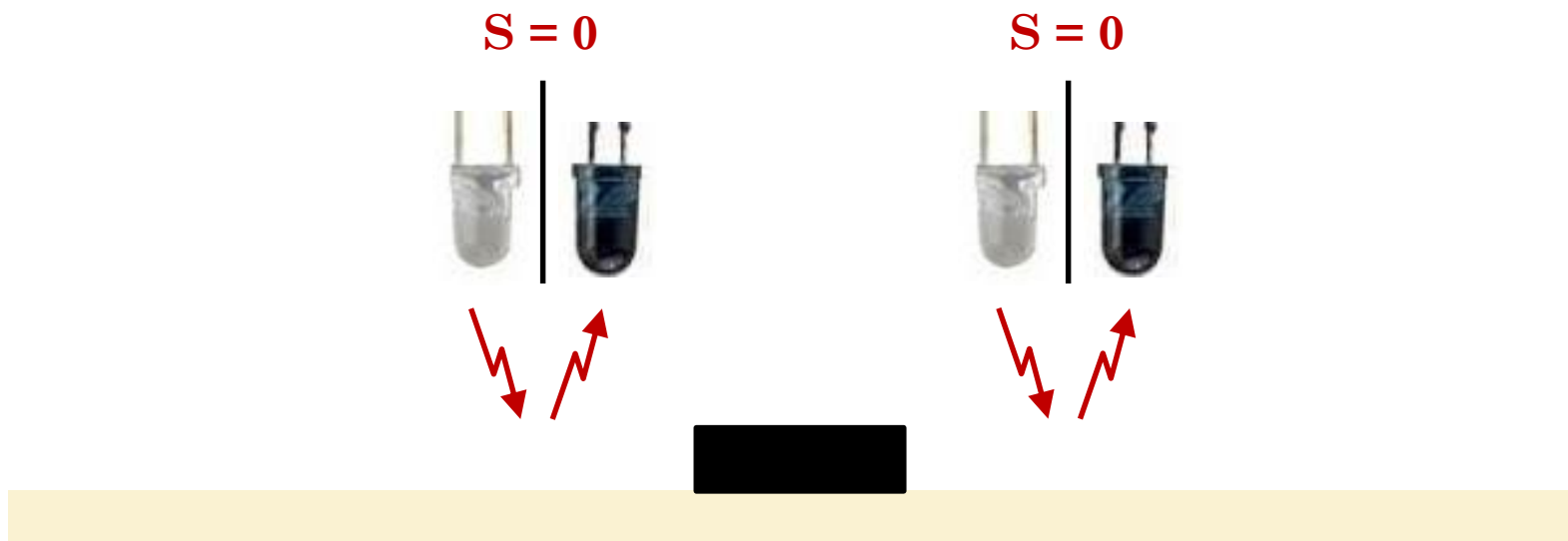
- L'information « être sur la ligne ou non » n'est pas suffisante
- Il faut aussi savoir de quel côté elle a été traversée
- Pour cela on peut utiliser 3 modules et déterminer celui qui se trouve au dessus de la ligne



3.5. Comment détecter une ligne

- Sens de croisement de la ligne

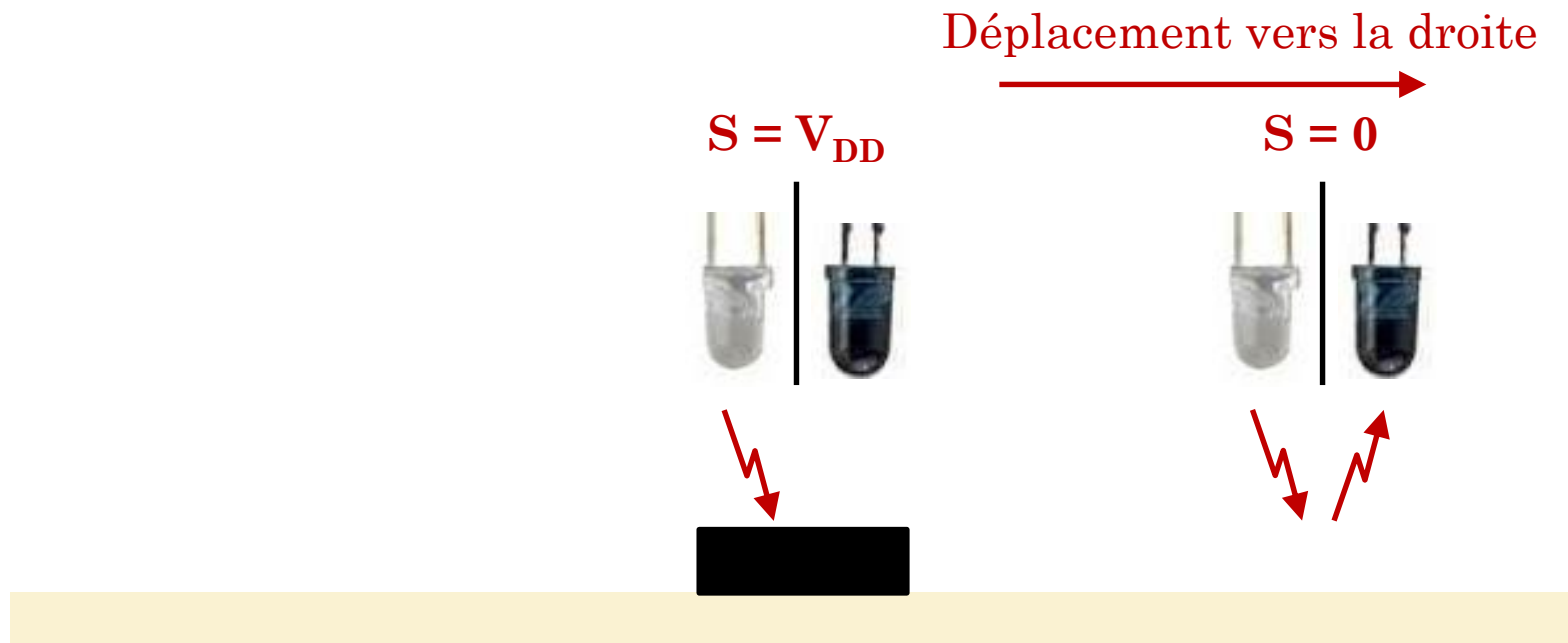
- Plus économique, on peut utiliser seulement 2 modules



3.5. Comment détecter une ligne

□ Sens de croisement de la ligne

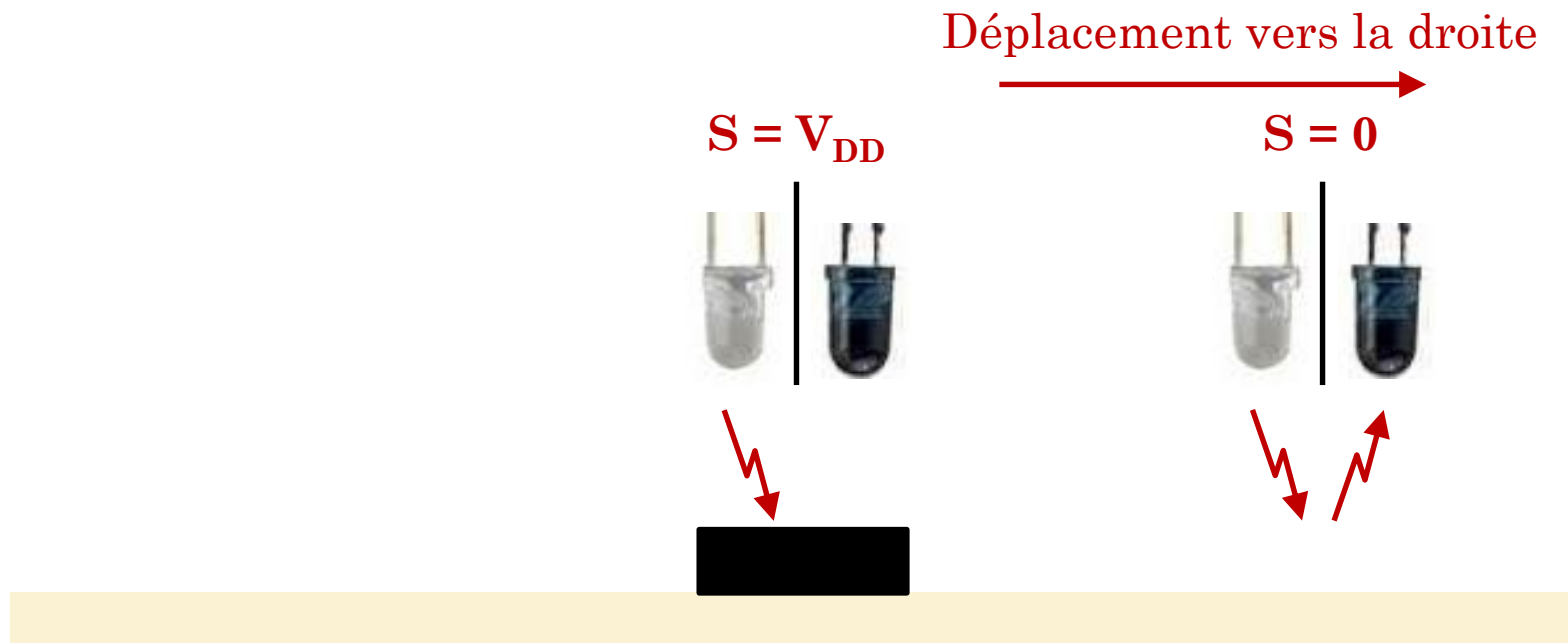
- Plus économique, on peut utiliser seulement 2 modules



3.5. Comment détecter une ligne

□ Sens de croisement de la ligne

- Plus économique, on peut utiliser seulement 2 modules



4.1. Introduction

- XXX
- XXX

EN COURS D'ECRITURE