



V2

Capteurs



Ou faire en sorte que l'on puisse faire la différence entre votre robot et Gilbert Montagné.

D'après Liam LOTTE

AREM : CDR



Menu du jour:

Apéritif : Kesako un capteur et une chaîne d'information ?

Entrée : Bases sur microcontrôleurs

Plat Principal : Types de capteurs et leurs sorties

Accompagnés de leurs exemples

Dessert : Les capteurs du robot de l'année dernière

AREM : CDR

AREM



Apéritif

1) Kesako un capteur et une chaîne d'information ?

AREM : CDR



1) Kesako un capteur et une chaîne d'information ?

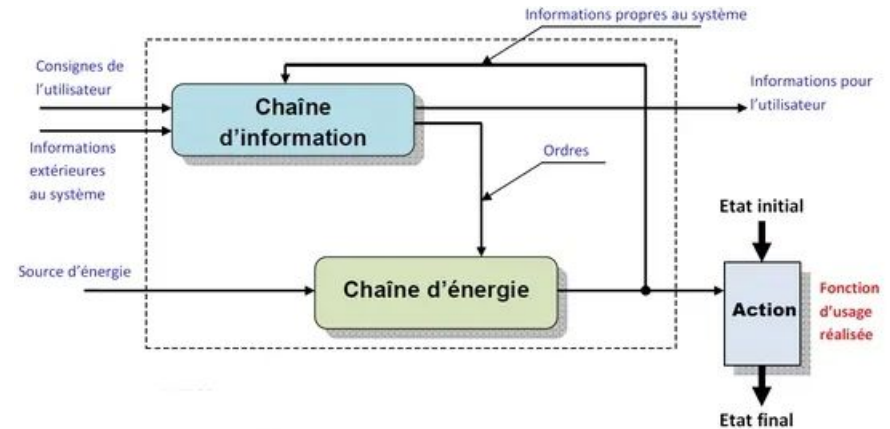
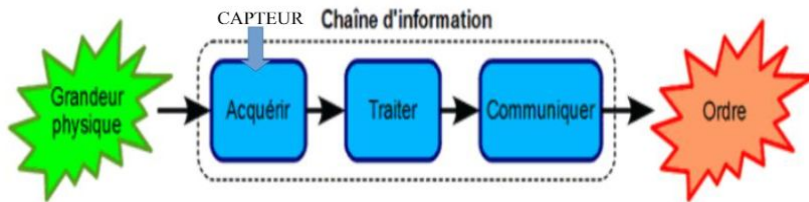
“Un **capteur** est un dispositif transformant l'état d'une **grandeur physique** observée en une grandeur utilisable, telle qu'une **tension électrique**, une **hauteur de mercure**, un **courant électrique** ou la déviation d'une aiguille.”

Source : Wikipédia (t'inquiètes)

AREM : CDR

AREM

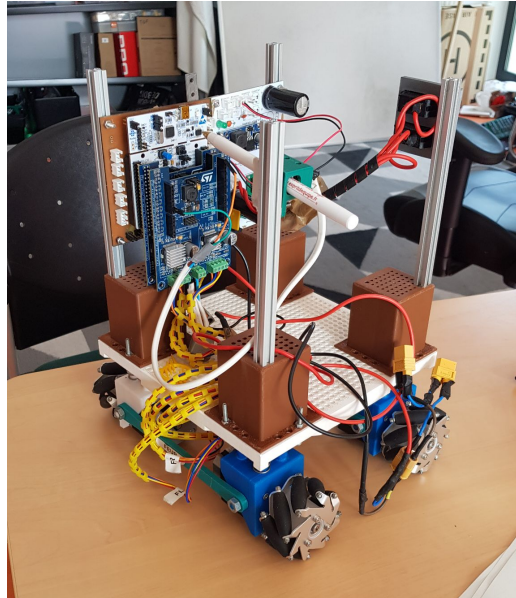
1) Kesako un capteur et une chaîne d'information ?



AREM : CDR



1) Kesako un capteur et une chaîne d'information ?



Cela nous sert à donner la perception du monde au robot, sans ça il est simplement aveugle...

<- Monstruosité

AREM : CDR



Entrée



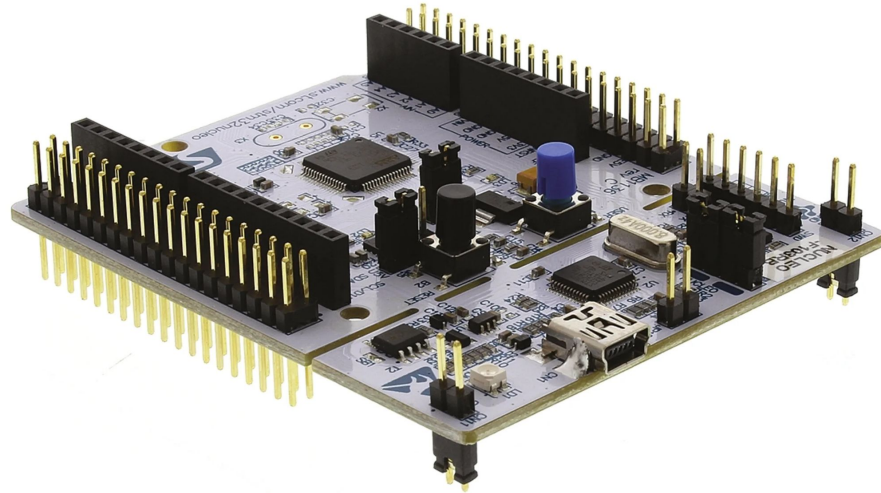
II) Bases sur Microcontrôleurs

AREM : CDR

AREM

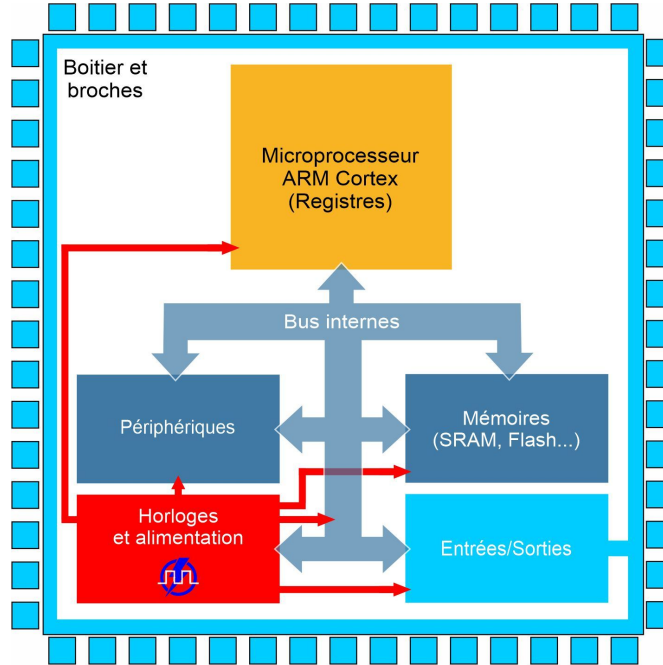
II) Bases sur Microcontrôleurs

On a quoi là dedans ?



AREM : CDR

II) Bases sur Microcontrôleurs



Composition :

Microprocesseur : Exécute le code (easy).

Mémoires : Stocke en général variables et programmes

AREM : CDR

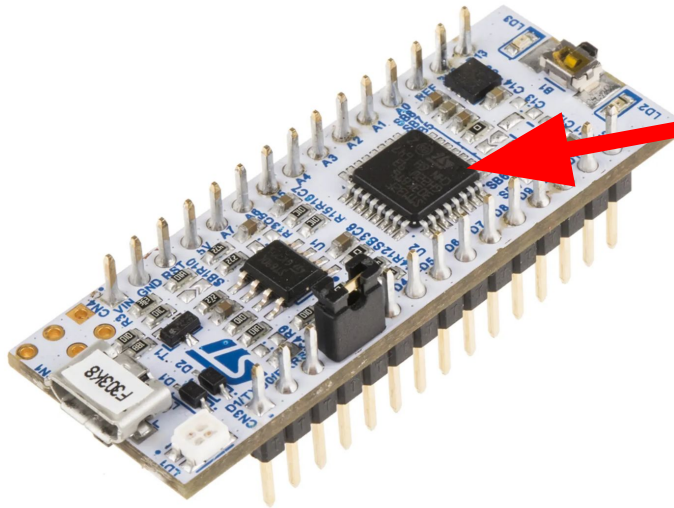
AREM

II) Bases sur Microcontrôleurs

Composition :

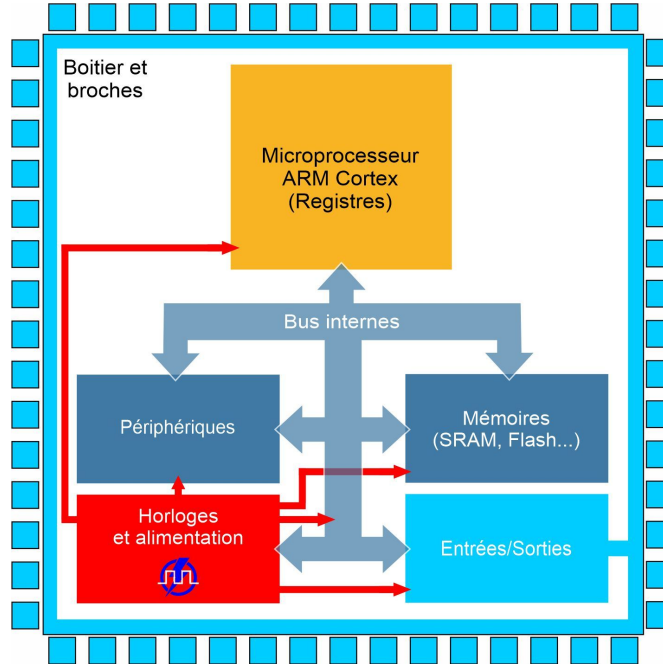
Microprocesseur : Exécute le code (easy).

Mémoires : Stocke en général variables et programmes



AREM : CDR

II) Bases sur Microcontrôleurs



Composition :

Entrée/sorties (ou GPIO) :

Lien avec le monde réel !

Chaque I/O est paramétrable
et permet l'acquisition de
signaux et l'envoi de
commandes. C'est la base !

AREM : CDR

AREM

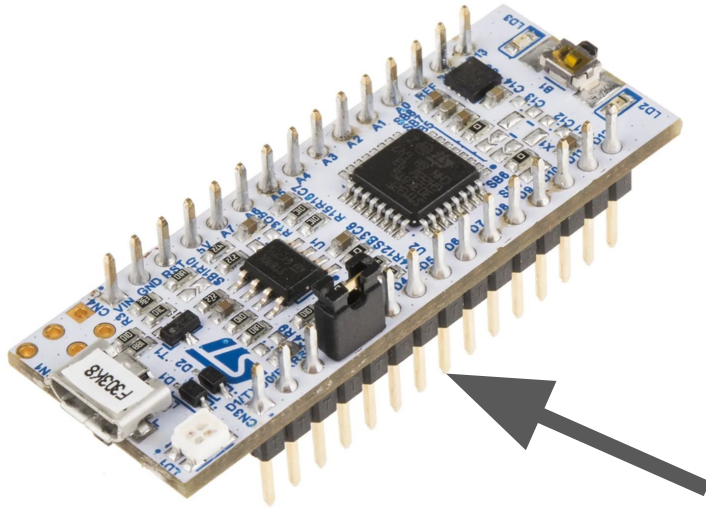
II) Bases sur Microcontrôleurs

Composition :

Entrée/sorties (ou GPIO) :

Lien avec le monde réel !

Chaque I/O est paramétrable
et permet l'acquisition de
signaux et l'envoi de
commandes. C'est la base !



C'est ça !

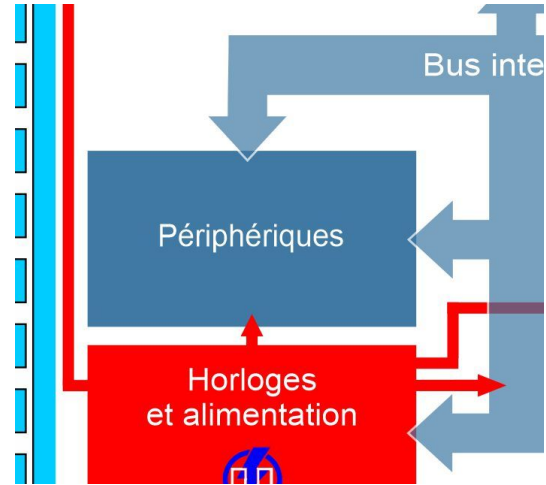
AREM : CDR

II) Bases sur Microcontrôleurs

Périphériques : Circuits intégrés au microcontrôleur qui réalisent des fonctions parfois très spécialisées requises par les applications.

Lorsqu'on veut les utiliser on doit les liés à une **entrée/sortie**.

! Cela va nous aider pour l'acquisition des données des capteurs !





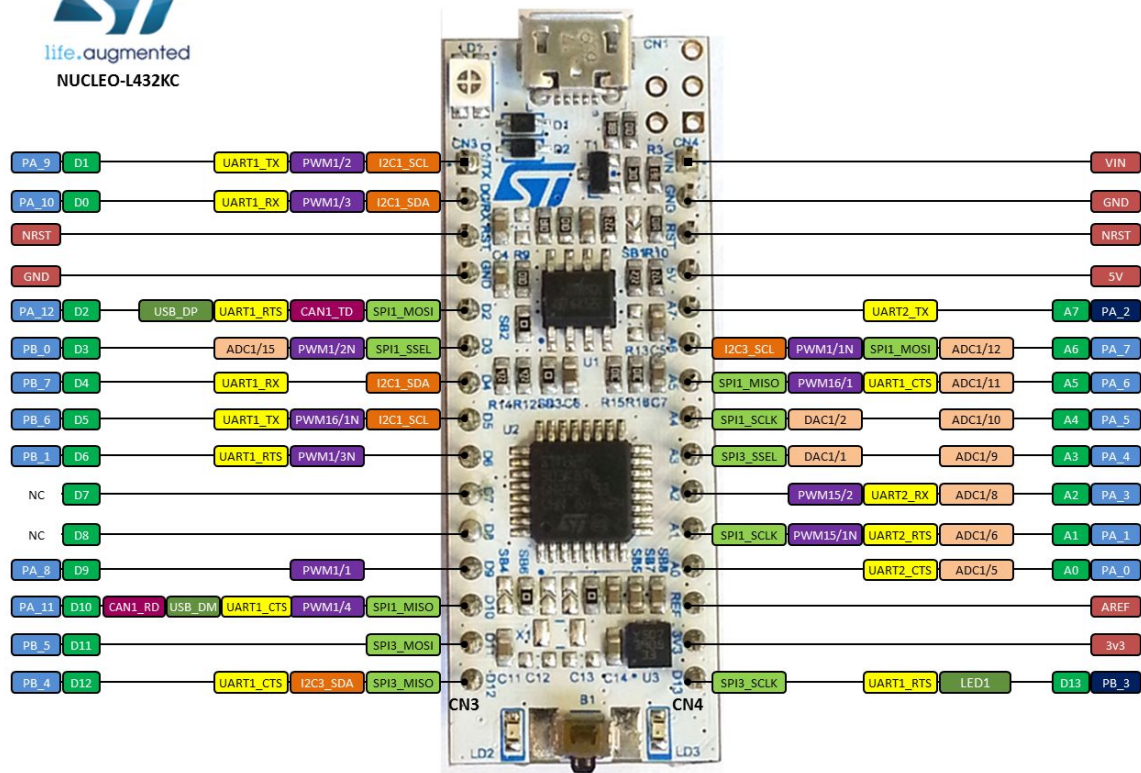
II) Bases sur Microcontrôleurs

En gros :

On appliquera donc les périphériques pour aider le microcontrôleur à parler dans la même langue que les capteurs.

On se posera donc la question de quel périphérique nous permet d'utiliser ce capteur à chaque fois que l'on voudra en ajouter un.

AREM : CDR



On se posera donc la question de quel périphérique nous permet d'utiliser ce capteur à chaque fois que l'on voudra en ajouter un.

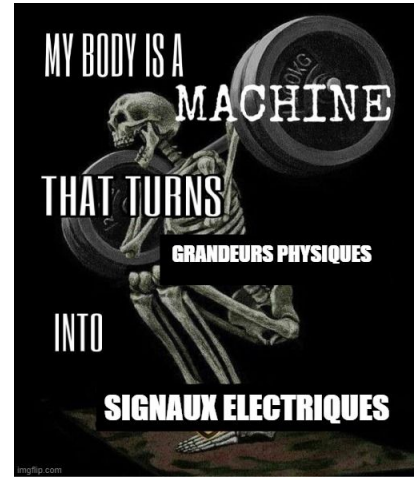
Regarde le **pinout** du microcontrôleur.

AREM



Plat Principal

III) Types de capteur et leurs sorties



AREM : CDR



III) Types de capteurs et leurs sorties

Passif : Pas d'alimentation propre

Exemple : Capteur Résistifs (résistance variant en fonction d'un paramètre donné).

Actif : Alimentation Propre

Exemple : Capteurs Ultrasons

AREM : CDR



III) Types de capteurs et leurs sorties

Passif : Pas d'alimentation propre

Exemple : Capteur Résistifs (résistance variant en fonction d'un paramètre donné).

Actif : Alimentation Propre <- Majorité de nos capteurs

Exemple : Capteurs Ultrasons

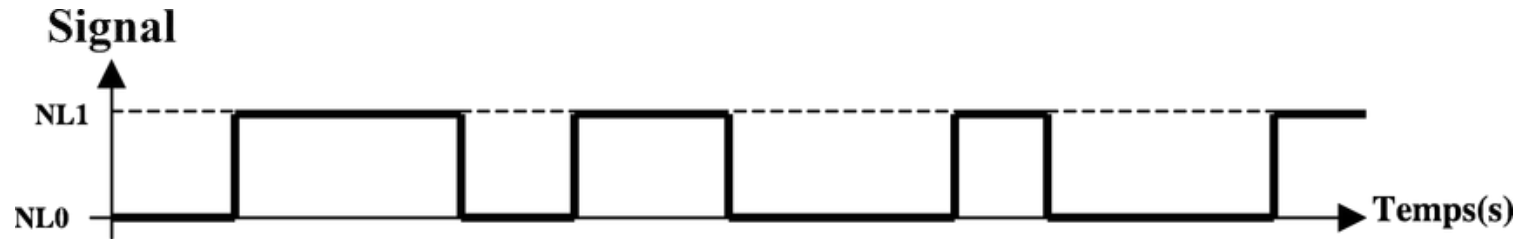
AREM : CDR

AREM



III) Types de capteurs et leurs sorties

Types de sorties : ***TOR - Logique***



AREM : CDR



III) Types de capteurs et leurs sorties

Types de sorties : ***TOR - Logique***

How to deal with ? : Entrée de base du microcontrôleur avec la fonction digitalRead() d'Arduino

Périphérique Hardware : n'importe quel entrée/sortie

Exemple : Bouton/Switch

AREM : CDR



III) Types de capteurs et leurs sorties

Types de sorties : ***Temps de réponse***

How to deal with ? : Vous attendez avec des delay()

(processus bloquant donc pas du tout idéal !)

Il existe des méthodes plus malignes en utilisant les interruptions du microcontrôleur (méthode plus complexe et vous verrez plus tard).

AREM : CDR

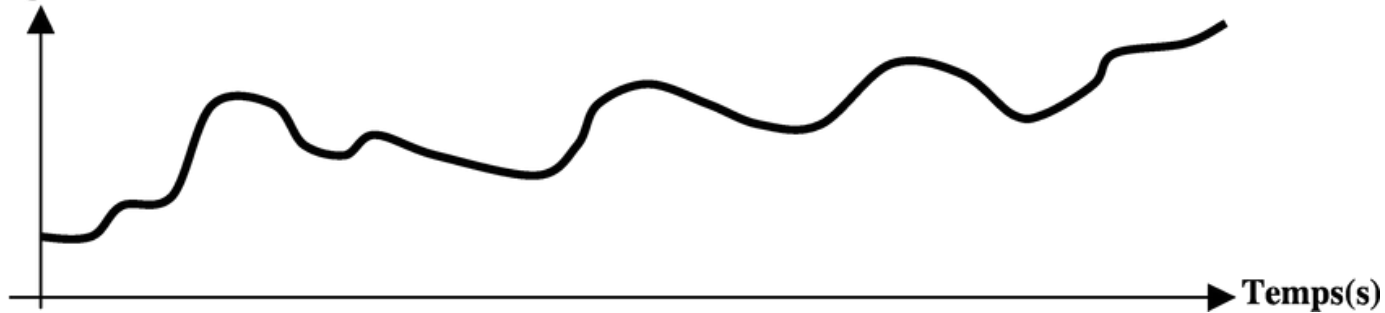
AREM



III) Types de capteurs et leurs sorties

Types de sorties : *Analogique*

Signal



AREM : CDR



III) Types de capteurs et leurs sorties

Types de sorties : ***Analogique***

How to deal with ? : Convertisseur Analogique Numérique du μ C grâce à la fonction Arduino “analogRead()”

Périphérique hardware lié :

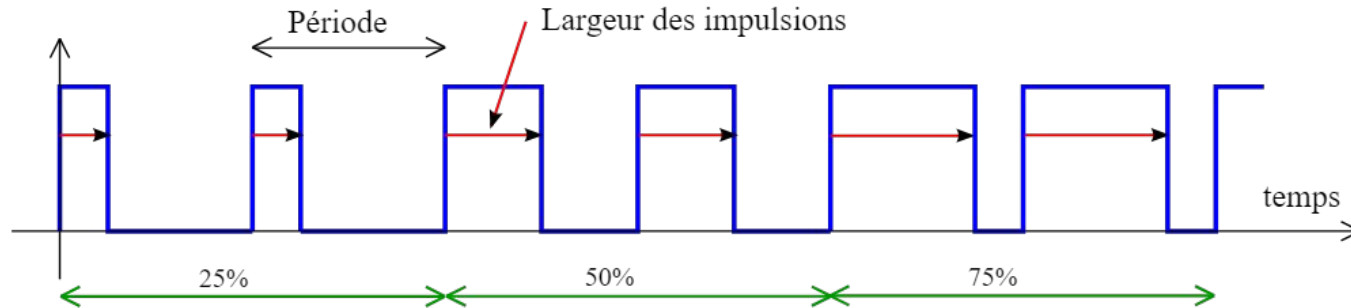
ADC (Analog to Digital Converter)

CAN en français

AREM : CDR

III) Types de capteurs et leurs sorties

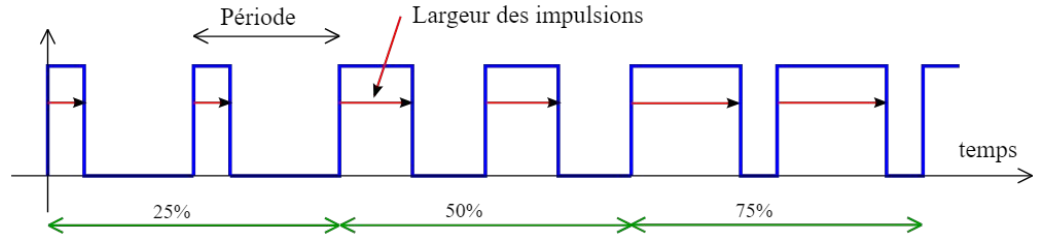
Types de sorties : **PWM** (Pulse Width Modulation)



AREM : CDR

III) Types de capteurs et leurs sorties

Cela nous permet en gros de changer la tension moyenne de notre signal !



$$U_{\text{moy}} = U * \alpha$$

Avec le rapport cyclique α .

$$\alpha = t_{\text{impulsion}} / T$$

AREM : CDR

III) Types de capteurs et leurs sorties

Types de sorties : **PWM** (Pulse Width Modulation)

How to deal with ? : Convertisseur Analogique Numérique du μ C grâce à la fonction Arduino “analogRead()”... again

Périphérique hardware lié : Le même que pour l’acquisition logique

ADC (Analog to Digital Converter)

CAN en français

AREM : CDR



III) Types de capteurs et leurs sorties

Types de sorties : Plein de protocoles de communications...

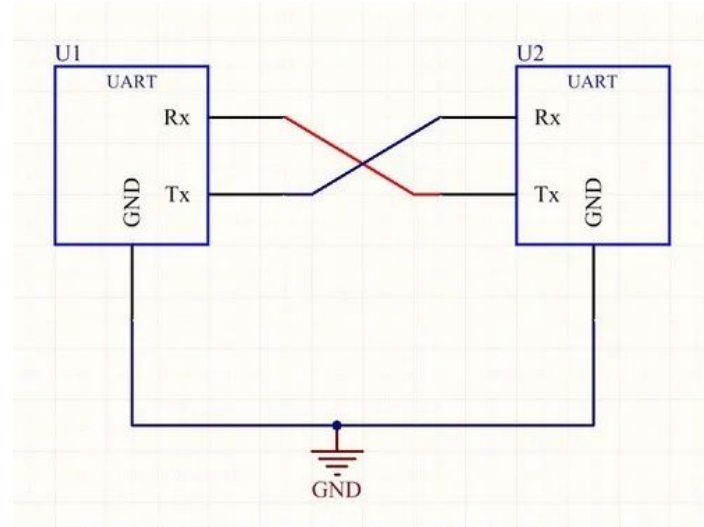
How to deal with ? : En général trouver une bibliothèque... mais on va donner des exemples !

AREM : CDR

III) Types de capteurs et leurs sorties

UART :

Communication avec un autre élément
en binaire.

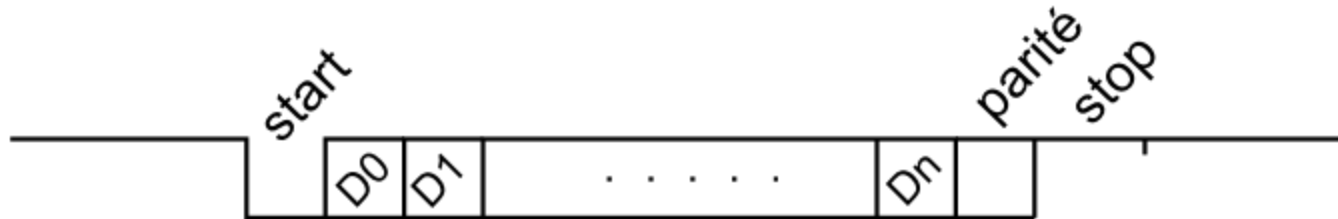


AREM



III) Types de capteurs et leurs sorties

UART :



AREM : CDR

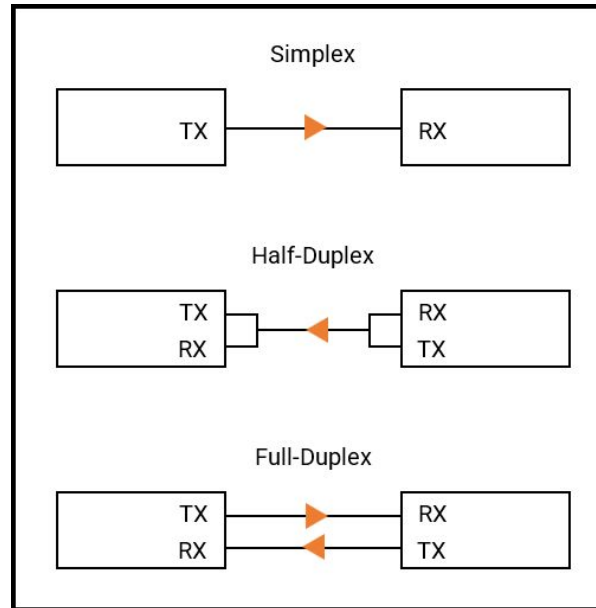
III) Types de capteurs et leurs sorties

UART :

Différence

entre full-duplex

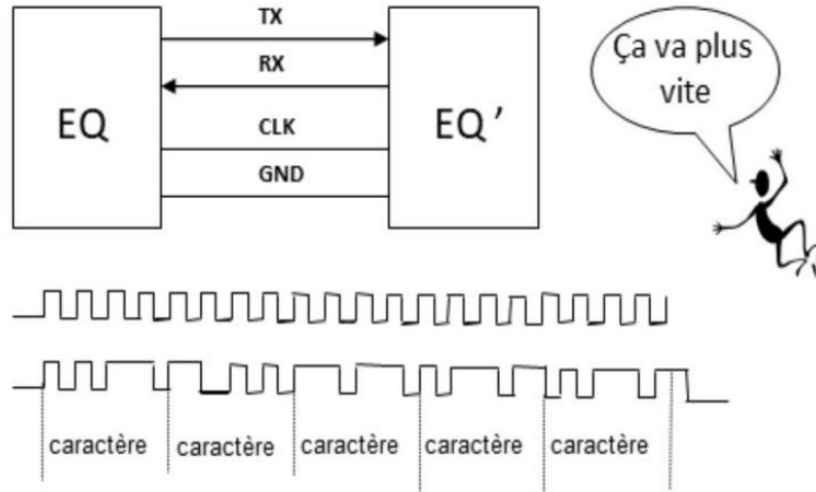
et half-duplex



AREM

III) Types de capteurs et leurs sorties

USART :



merci Marques

AREM : CDR



III) Types de capteurs et leurs sorties

UART et USART :

Comment on fait mon capitaine ? : Bibliothèque <Serial.h> native sur Arduino.

Périphérique hardware lié : UART (damn)

AREM : CDR

AREM

III) Types de capteurs et leurs sorties

Problème avec l'UART/USART :

1 capteur = 1 périphérique -> **On a rapidement plus de places**

Malika Doray

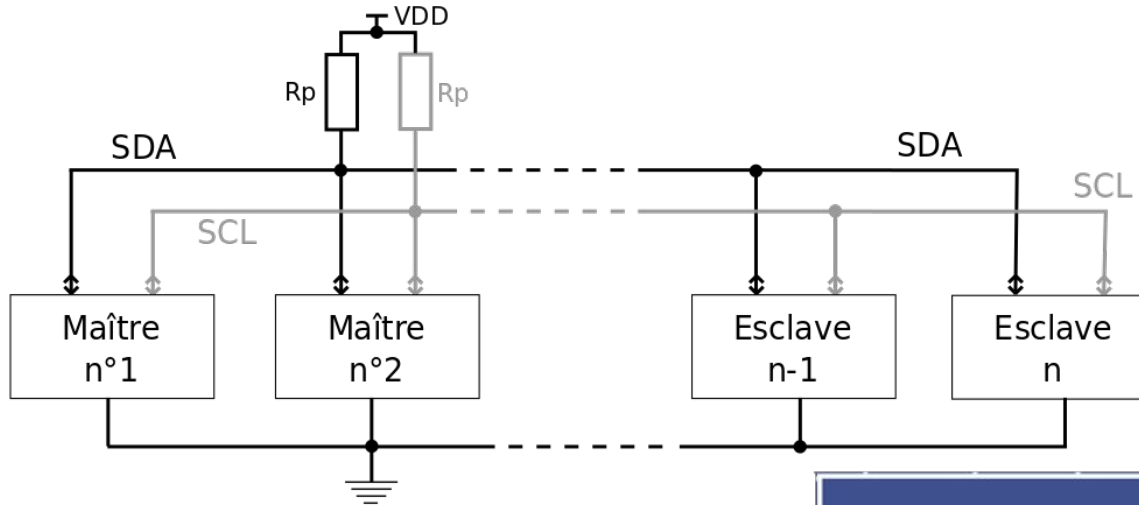
Y'A PLUS DE PLACE !



AREM : CDR

III) Types de capteurs et leurs sorties

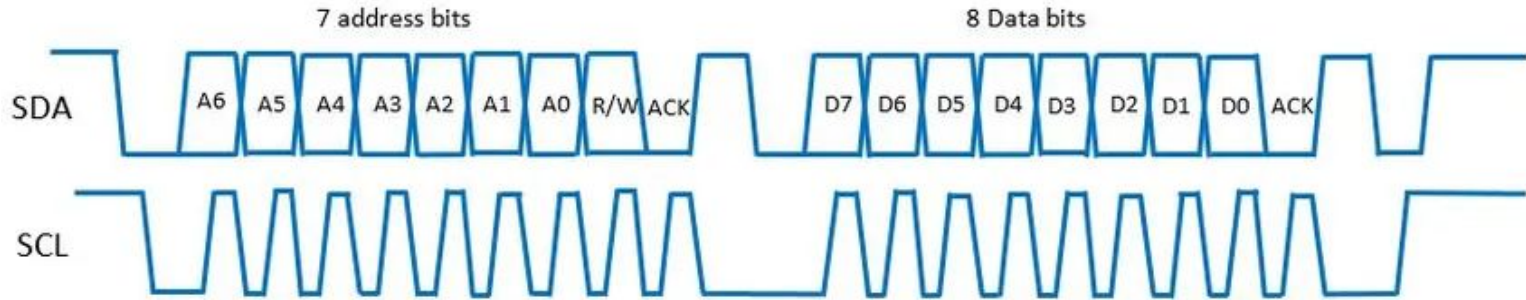
I²C :



AREM : CDR

III) Types de capteurs et leurs sorties

I²C :



Entre 100 ou 400 kHz en général

-> Assez lent en réalité

AREM : CDR



III) Types de capteurs et leurs sorties

I²C :

Comment on fait mon capitaine ? : Bibliothèque <Wire.h> native sur Arduino.

Périphérique hardware lié : I²C (c'est crazy)

AREM : CDR



III) Types de capteurs et leurs sorties

I²C : Limite

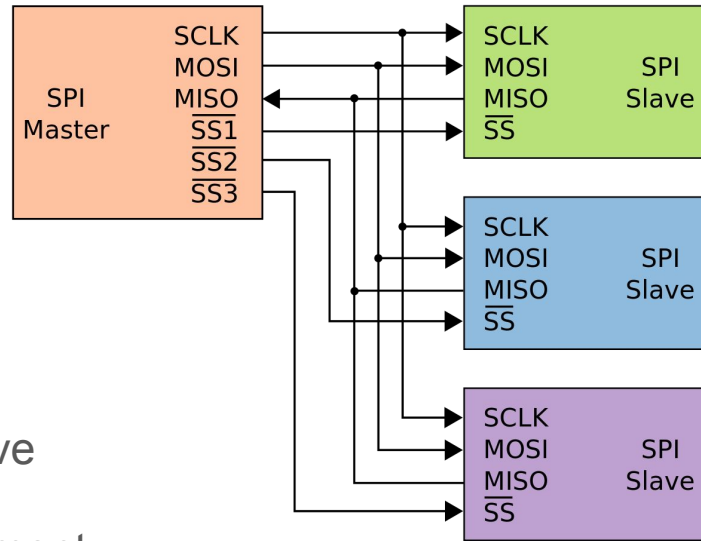
Le protocole est très simple, trop simple en fait... Sur de longues distance (plusieurs mètres) il ne fonctionnera plus à cause du bruit électronique. Dans ce cas, ne vous battez pas. Changez de protocole (CAN* par exemple)

*Pas le convertisseur électronique analogique mais le protocole mon gâté.

AREM : CDR

III) Types de capteurs et leurs sorties

SPI :



- Maître/Esclave
- Full Duplex
- Pas Acquittement
- Choix des tailles de packets

AREM



III) Types de capteurs et leurs sorties

SPI : en gros l'I²C mais en rapide



AREM : CDR



III) Types de capteurs et leurs sorties

SPI :

Comment on fait mon capitaine ? : Bibliothèque <SPI.h> native sur Arduino.

Périphérique hardware lié : SPI (ok je l'ai)

AREM : CDR

III) Types de capteurs et leurs sorties

Protocole :	U(S)ART	I ² C	SPI
Avantage :	<ul style="list-style-type: none">- Rapide- Simple- Bi Directionnel	<ul style="list-style-type: none">- Simple d'utilisation- Commun	<ul style="list-style-type: none">- Rapide- Modulaire
Inconvénient :	<ul style="list-style-type: none">- Uniquement point à point	<ul style="list-style-type: none">- Lent	<ul style="list-style-type: none">- Plus de câbles que l'I²C

Point à Point / Série

AREM : CDR



IV) Les capteurs du robot de l'année dernière

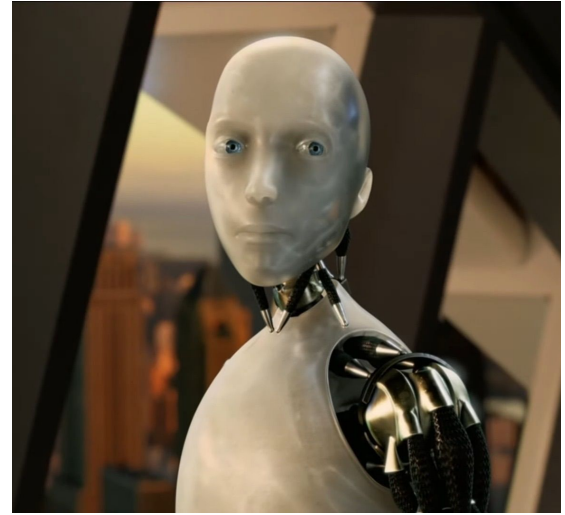
- **Lidar**
 - Utilisation : Pour la vision du robot
 - Protocole : UART
 - Périphérique : UART
- **Capteur de fin de course (bouton à lamelle)**
 - Utilisation : Pour détecter le retrait de la tirette
 - Protocole : UART
 - Périphérique : UART

AREM : CDR

AREM

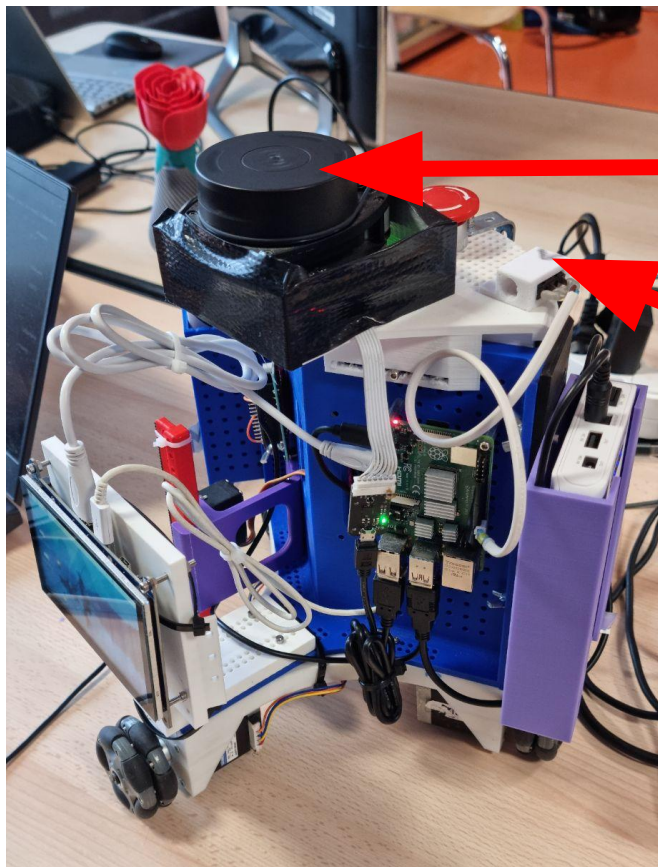


Dessert



IV) Les capteurs du robots de l'année dernière

AREM : CDR



Lidar

Capteur de
fin de course

AREM : CDR



IV) Les capteurs du robot de l'année dernière

Capteur des PAMI's :

- **TOF (Time Of Flight)** : Capteur de distance fonctionnant à base de rebond de lumière (donc rapide). Il a pour particularité d'avoir un processeur dessus qui calcul en continue la distance et renvoie le résultat en I²C.

AREM : CDR



La règle maîtresse

Pour chaque chose que vous avez à faire, faites les recherches de la *datasheet* de votre capteur et *cherchez si des bibliothèques existent déjà* !

Vous gagnerez du temps



AREM : CDR



AREM

La règle maîtresse

la suite

Si vous utilisez des capteurs, **documentez les** ! Pour stocker votre documentation, utilisez le saint GitHub !

Chad mec qui documente son travail



Virgin codeur fou



AREM : CDR



[*https://github.com/AREM-Proiets/documentation-capteurs*](https://github.com/AREM-Proiets/documentation-capteurs)



AREM : CDR



KA-CHOW!

Go faire des robots les zamis