

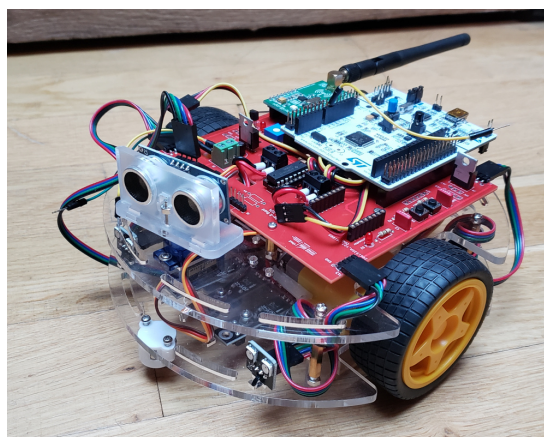
INTERFAÇAGE NUMÉRIQUE

Travaux Pratiques

Semestre 6

Robotique et systèmes embarqués

4 séances



Ce sujet est disponible au format électronique sur le site du LEnsE - <https://lense.institutoptique.fr/> dans la rubrique Année / Première Année / Interfaçage Numérique S6 / Robotique.

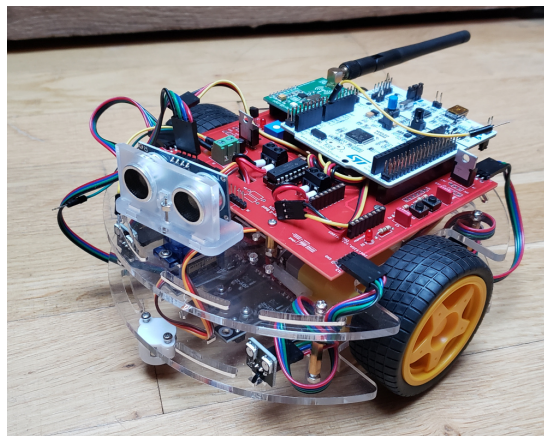
Robotique et systèmes embarqués

À l'issue des séances de TP concernant le **bloc de robotique**, les étudiant-es seront capables de :

- Développer et mettre en œuvre une **solution d'électronique embarquée** pour **mettre en mouvement un robot**
 - Concevoir un **programme embarqué** permettant de **rendre autonome les mouvements d'un robot**
-

Objectifs du mini-projet

L'objectif principal de ce mini-projet est de **développer le code embarqué d'une plateforme robotique** lui permettant de se déplacer de manière autonome le long d'une ligne sans percuter d'obstacle.



Vous aurez à votre disposition une **maquette** basée sur un robot Joy-It Car. Cette maquette est pilotée par une carte Nucléo (contenant un microcontrôleur).

Déroulement du bloc

La liste des étapes à suivre pour la réalisation du programme embarqué de la plateforme de rayonnement lumineux est donnée à titre indicatif. L'ordre et le choix des différentes étapes sont laissés à l'appréciation des différents binômes.

Afin de faciliter la réutilisation des codes, il pourra être intéressant de définir des fonctions pour le pilotage des différents éléments.

Séance 1 / Arduino et Nucléo-STM32 (sans maquette !!)

Le sujet de cette séance est fourni dans un document annexe, disponible aussi sur le site du LEnsE - <https://lense.institutoptique.fr/> dans la rubrique Année / Première Année / Interfaçage Numérique S6 / Bloc Systèmes embarqués / Intro Arduino et STM32.

Etape 0 - 30 min Installer les drivers STM32 et tester un premier programme

Etape 1 - 45 min Piloter des sorties numériques - LED

Etape 2 - 45 min Acquérir des données numériques - Bouton-poussoirs

Etape 3 - 45 min Mettre en œuvre des interruptions sur des événements externes

Etape 4 - 45 min Utiliser des sorties modulées en largeur d'impulsion (PWM) - LEDs

Etape 5 - 60 min Acquérir des données analogiques - Potentiomètre

Séance 2 / Prise en main de la maquette et déplacements élémentaires

Etape 6 - 60 min Piloter l'intensité des LEDs de la maquette

Etape 7 - 90 min Piloter les moteurs à courant continu

Etape 8 - 60 min Acquérir des données des capteurs de ligne

Etape 9 - 60 min Piloter les phares du robot à l'aide de la bibliothèque WS2812 (NeoPixel)

Séances 3 et 4 - Pilotage de haut niveau

Les deux séances suivantes seront consacrées au pilotage du robot pour lui permettre de suivre une ligne ou/et d'éviter les obstacles qu'il rencontre sur son chemin.

Les étapes possibles sont les suivantes :

Etape 10 - 120 min Définir et tester une première structure de code permettant de piloter les deux moteurs du robot en fonction de la détection des lignes

Etape 11 - 90 min Acquérir les signaux du capteur ultrason

Etape 12 - 90 min Piloter le servomoteur associé au capteur ultrason

Etape 13 - 180 min Améliorer le programme de contrôle du robot

Vous pourrez également ajouter d'autres éléments présents sur la carte : encodeur de vitesse sur les roues, capteurs de température (analogique ou numérique en I2C), accéléromètre (I2C).

Séance 2 / Prise en main de la maquette et déplacements élémentaires

Etape 6 - 60 min Piloter l'intensité des LEDs de la maquette

Etape 7 - 90 min Piloter les moteurs à courant continu

Etape 8 - 60 min Acquérir des données des capteurs de ligne

Etape 9 - 60 min Piloter les phares du robot à l'aide de la bibliothèque WS2812 (NeoPixel)

Objectifs de la séance

Cette seconde séance est consacrée à la **prise en main de la maquette** et au développement des **fonctionnalités permettant les déplacements élémentaires** de la plateforme.

Description de la maquette

Éléments constitutifs

Alimentation électrique

La tension maximale admissible par les moteurs est de 7 V !

Brochage

Entrées-Sorties standard

| Maquette | Broche Nucléo | Type | Description |
|------------|---------------|-------------------|---------------------------------------|
| LED1 | PC7 | Sortie / PWM | Led active à l' état haut |
| LED2 | PB13 | Sortie / PWM | Led active à l' état bas |
| SW1 | PA11 | Entrée | Bouton-poussoir, par défaut état bas |
| SW2 | PA12 | Entrée | Bouton-poussoir, par défaut état bas |
| USERBUTTON | PC13 | Entrée | Bouton-poussoir, par défaut état haut |
| POT_OUT | PC3 | Entrée analogique | Potentiomètre |

Moteurs

| Maquette | Broche Nucléo | Type | Description |
|----------|---------------|--------------|---------------------------|
| MOT_EN | PA9 | Sortie | Validation des moteurs |
| MOT_L_1 | PB4 | Sortie / PWM | Moteur Gauche direction 1 |
| MOT_L_2 | PA8 | Sortie / PWM | Moteur Gauche direction 2 |
| MOT_R_1 | PA0 | Sortie / PWM | Moteur Droit direction 1 |
| MOT_R_2 | PA1 | Sortie / PWM | Moteur Droit direction 2 |

Phares NeoPixel / SW2812

| Maquette | Broche Nucléo | Type | Description |
|----------|---------------|--------|----------------------|
| DIN_1 | PC0 | Sortie | Phare avant droit |
| DIN_2 | PA10 | Sortie | Phare avant gauche |
| DIN_3 | PC5 | Sortie | Phare arrière gauche |
| DIN_4 | PA13 | Sortie | Phare arrière droit |

Capteurs

| Maquette | Broche Nucléo | Type | Description |
|----------|---------------|-------------------|---------------------------------|
| TEMP_OUT | PC2 | Entrée analogique | Capteur de température MCP9700 |
| LINE_L | PA7 | Entrée | Capteur de ligne Gauche |
| LINE_C | PB6 | Entrée | Capteur de ligne Centre |
| LINE_R | PA5 | Entrée | Capteur de ligne Droit |
| SPEED_L | PC9 | Entrée | Vitesse moteur Gauche |
| SPEED_R | PC8 | Entrée | Vitesse moteur Droit |
| US_TRIG | PB5 | Sortie | Capteur ultrason - Trig |
| US_ECHO | PB3 | Entrée | Capteur ultrason - Echo |
| SERVO | PB7 | Sortie / PWM | Servomoteur du capteur Ultrason |

Accéléromètre / MikroE-6DOF IMU 3 Click

Ce module fonctionne à l'aide du protocole I2C.

| Maquette | Broche Nucléo | Type | Description |
|-----------|---------------|---------------|----------------------------------|
| SDA | PB9 | Entrée-Sortie | Signal de données bidirectionnel |
| SCL | PB8 | Sortie | Signal d'horloge |
| RESET | PC4 | Sortie | Reset matériel du composant |
| Interrupt | PB10 | Entrée | Interruption sur réception |

Capteur température numérique / TC74A2

Ce module fonctionne à l'aide du protocole I2C.

| Maquette | Broche Nucléo | Type | Description |
|----------|---------------|---------------|----------------------------------|
| SDA | PB9 | Entrée-Sortie | Signal de données bidirectionnel |
| SCL | PB8 | Sortie | Signal d'horloge |

Communication nRF24L01

Ce module fonctionne selon le protocole SPI. *Il doit nécessairement être utilisé avec un second module afin de pouvoir transmettre des données entre deux microcontrôleurs.*

| Maquette | Broche Nucléo | Type | Description |
|-------------|---------------|--------|-------------------------------------|
| SPI | | | |
| <i>SCK</i> | PC10 | Sortie | Signal d'horloge |
| <i>MISO</i> | PC11 | Entrée | Données entrantes |
| <i>MOSI</i> | PC12 | Sortie | Données sortantes |
| <i>CS</i> | PA14 | Sortie | Sélection du composant |
| <i>CE</i> | PD2 | Sortie | Validation du composant (puissance) |
| <i>INT</i> | PA15 | Entrée | Interruption sur réception |

Utilisation de la sortie modulée PB7

```
1 LL_GPIO_SetAFPin_0_7(GPIOB, GPIO_PIN_7, GPIO_AF1_TIM2);
```

Etape 9 - Acquérir des données de l'accéléromètre (I2C)

Temps conseillé : 90 min

Le composant que nous allons étudier est un **accéléromètre et magnétomètre** intégrés sur une même puce de silicium. Sa référence est **FXOS8700CQ**. Ce composant est intégré au module **MikroE DOF6 - IMU Click**.

Protocole I2C

DESCRIPTION PROTOCOLE et CONNECTIQUES !

ATTENTION ! Les broches utilisées sur la carte Nucléo pour l'I2C ne sont pas celles par défaut. Il est indispensable de préciser les broches SDA et SCL à l'aide des méthodes suivantes :

```
1 Wire.setSDA( PB9 );  
2 Wire.setSCL( PB8 );
```

→ M Ouvrir le code *09_accelero.ino* fourni. Compiler ce code et téléverser ce code dans la carte Nucléo.

Ce code contient les fonctions *test_FXOS()* et *read_i2c_buffer()*, ainsi que des définitions des registres internes du composant.

→ M

Configuration

Récupération des données

These registers contain the X-axis, Y-axis, and Z-axis 14-bit left-justified sample data expressed as 2's complement numbers. [NXP Doc p.52 of 113]

Traceur Série

```
1 Serial.print(valeur1);  
2 Serial.print(",");  
3 Serial.print(valeur2);  
4 Serial.print(",");  
5 Serial.print(valeur3);  
6 Serial.print(",");  
7 Serial.print(valeur4);  
8 Serial.println();
```

Robot Joy-It Car / Présentation du matériel

La tension maximale admissible par les moteurs est de 7 V !

INTERFAÇAGE NUMÉRIQUE

Travaux Pratiques

Semestre 6

Ressources

Bloc Robot

Liste des ressources

- [Schéma de la carte du robot Joy-It Car](#)
- [PCB de la carte du robot Joy-It Car](#)

