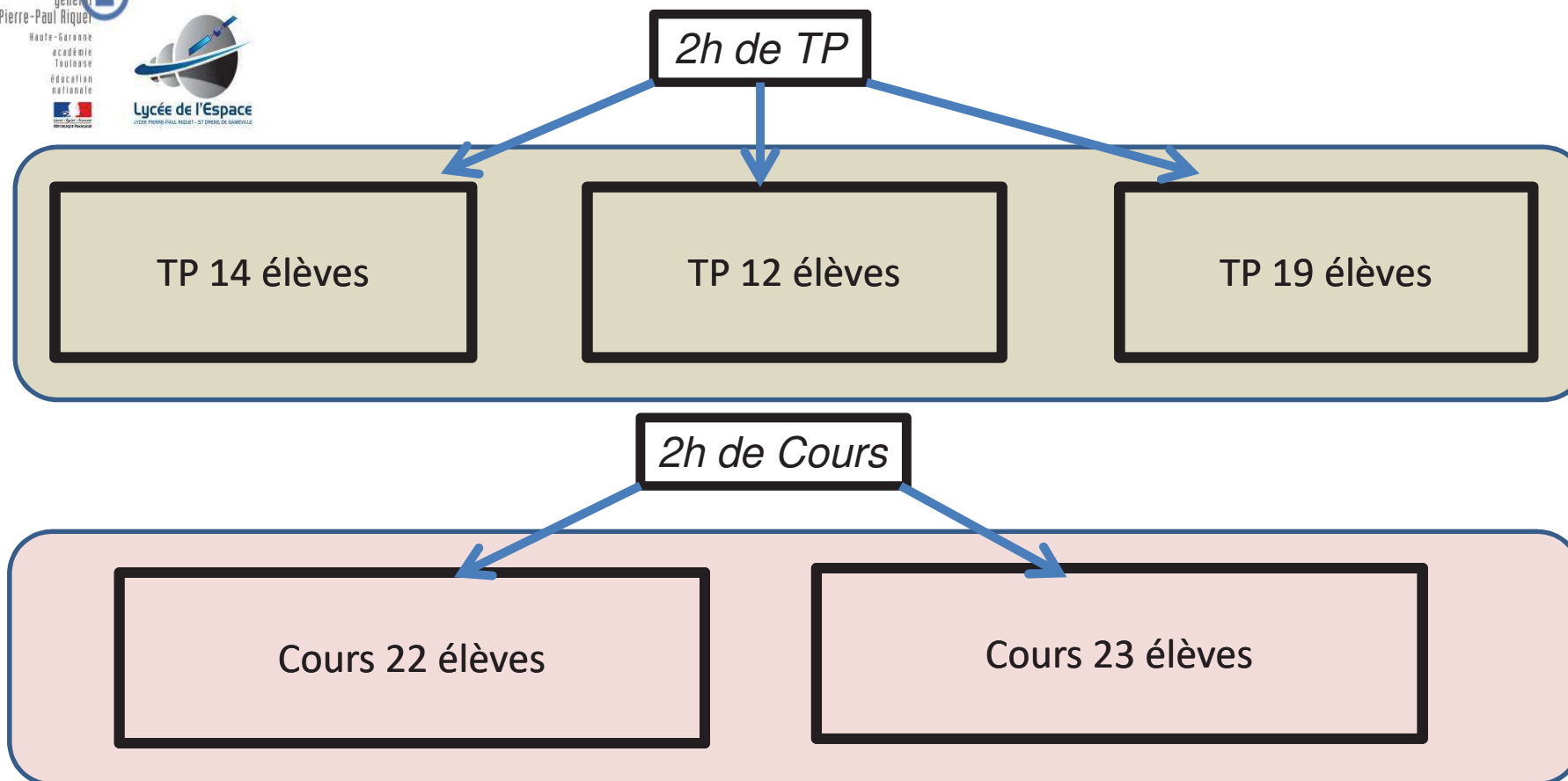




Projet Sciences de l'Ingénieur:

*Lycée Pierre-Paul Riquet
Saint Orens*

Organisation Pédagogique à Pierre Paul Riquet en première Eds SI:



Effectif Total Enseignement de Spécialité Sciences de l'Ingénieur: 45 élèves

Organisation du projet

5 semaines (entre deux périodes de vacances scolaires)

Semaine	Créneau horaire
Semaine 1	3h
Semaine 2	2h
Semaine 3	2h
Semaine 4	2h
Semaine 5	3h
<u>Total</u>	<u>12h</u>

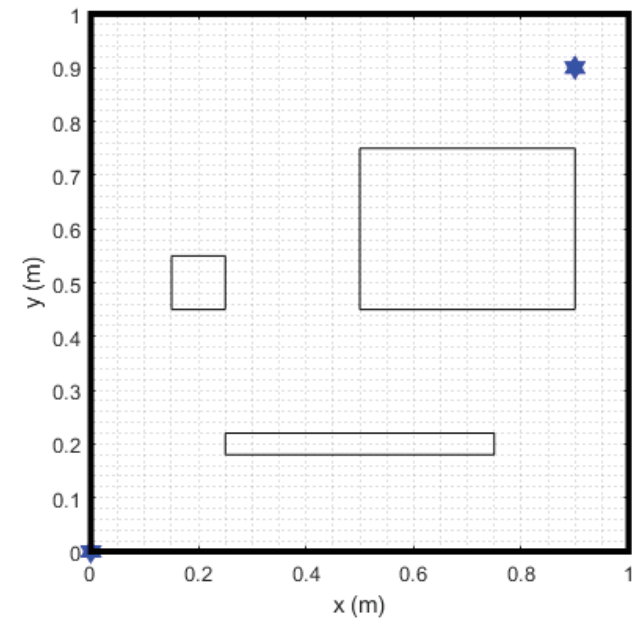
Thématique

Les mobilités des personnes et des biens

Le challenge (ou défi) Robot mobile transporteur autonome



Amener puis déposer un colis d'un point de départ à un point d'arrivée en évitant les différents obstacles fixes ou mobiles lors du parcours.



2 types de robots mis à disposition (1 robot par équipe)

Robot avec élévateur Arduino-Matlab



Il peut transporter le colis
avec un élévateur

Il se programme sous
environnement
Matlab/Simulink/Arduino

motoréducteur
codeur



servomoteur



Capteur à
ultrason



Motoréducteur

Capteurs de ligne

Capteurs de
lumière

Bandes de Leds
Néopixels

Communication Radio pour
signaler des anomalies

Il peut pousser le colis ou le
transporter

Il se programme sous
environnement Python

Robot BitBot micro-python



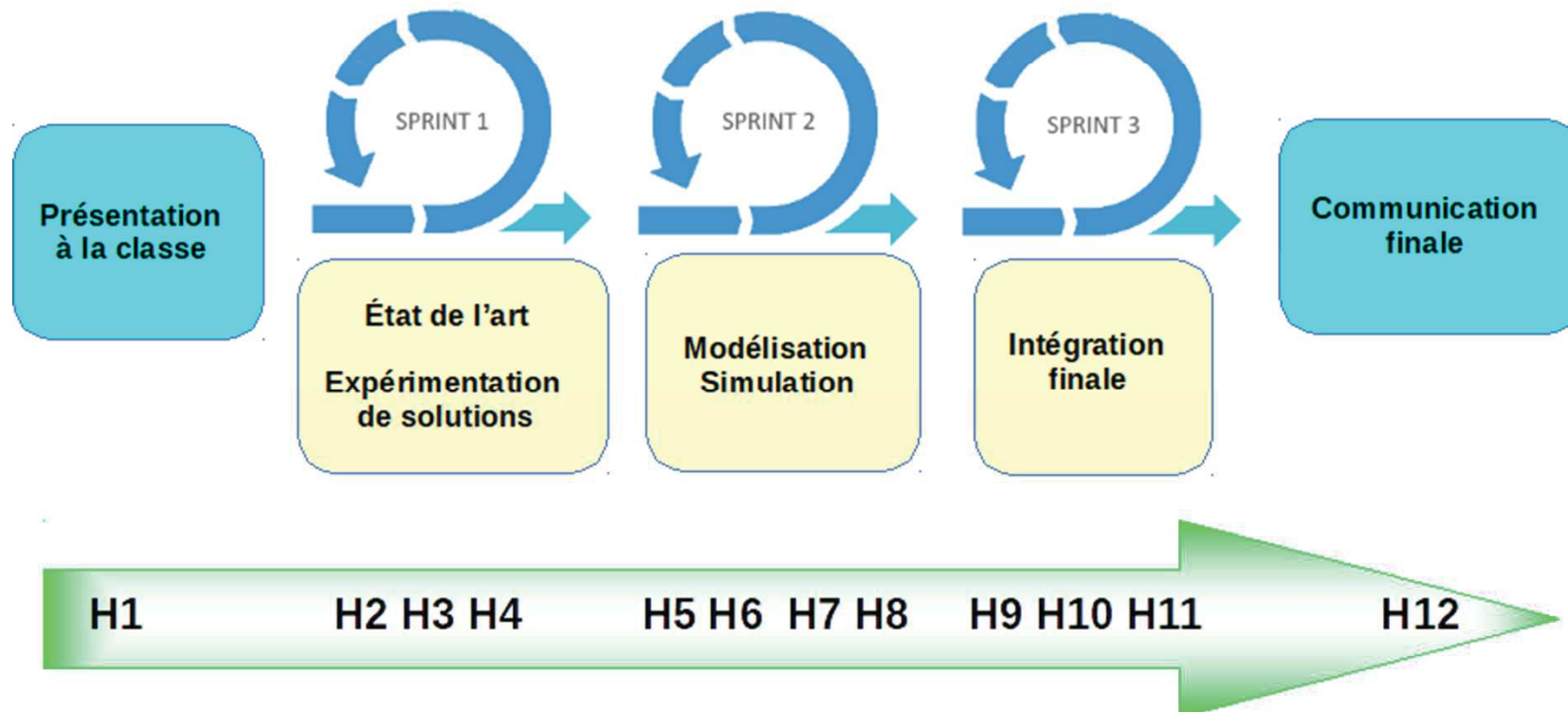
Organisation des équipes de projet et matériels nécessaires

- Équipe de 4 élèves (5 maxi) pour deux classes de 22 élèves → 11 équipes de 4 élèves

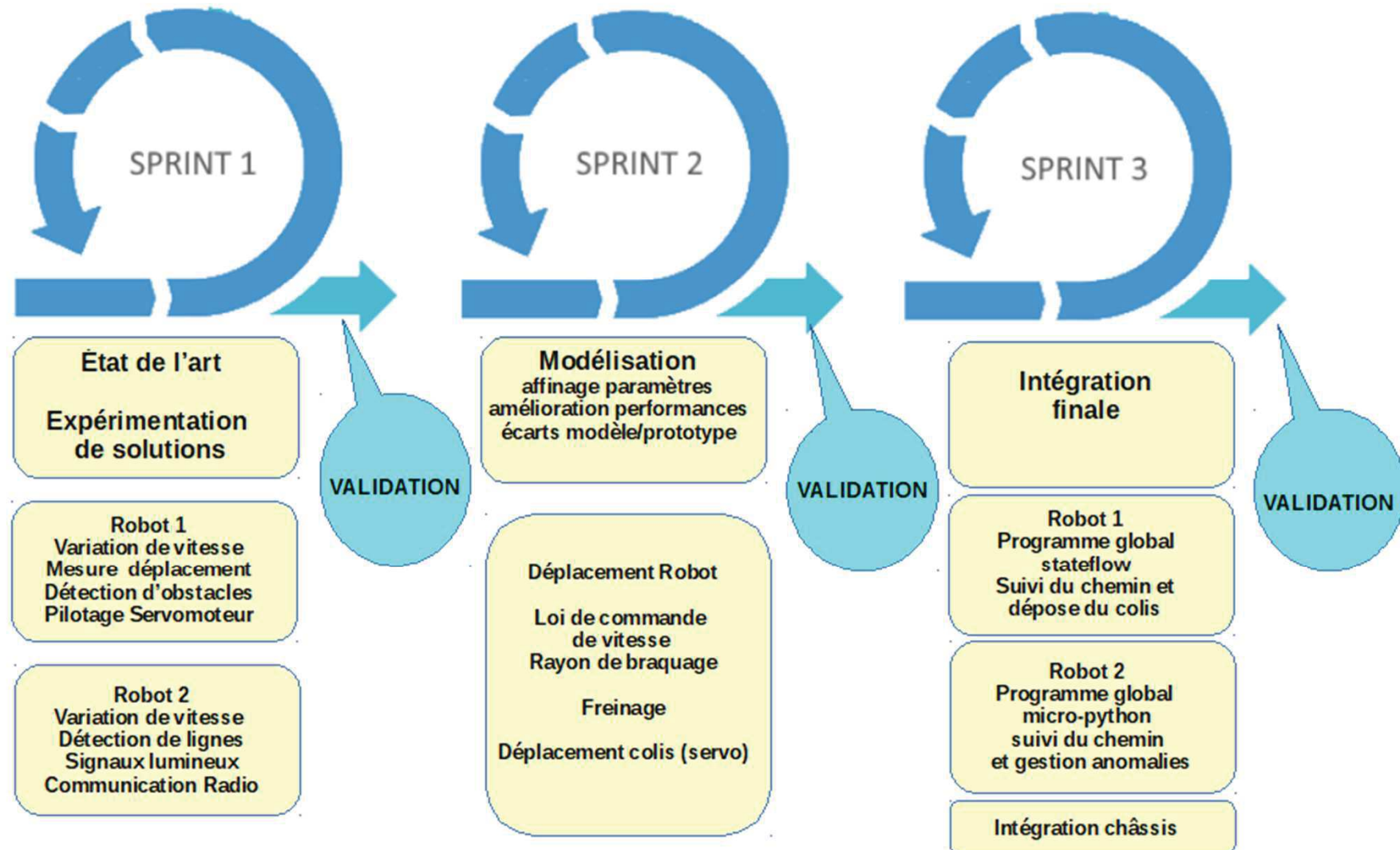
- Matériels à disposition
 - 4 Robots avec élévateur Arduino-Matlab
 - 7 Robots Bit Bot micro-python
 - 22 PC (1 PC par élève)
 - Des guidances pour chaque activité
 - Des ressources pour chaque activité
 - Des modèles (à compléter) pour chaque activité
 - Des appareils de mesure (oscilloscope, multimètre, tachymètre...)

Déroulement

3 Sprints (méthode Agile/Scrum)



Déroulement



Présentation H1

Équipe : Appropriation du projet - Découverte du challenge / mise en place des groupes

Expérimentation H2 H3 H4

Mise en œuvre du moteur + réducteur +
roue + variation de vitesse

Mise en œuvre du codeur incrémental
Mise en œuvre capteurs suivi de lignes

Mise en œuvre d'un capteur ultrason
Mise en œuvre du module radio

Mise en œuvre du servomoteur.
Mise en œuvre des Leds Néopixel

Modélisation/Simulations H5 H6 H7 H8

Simulation multi-physique du déplacement du robot et de la stratégie de déplacement

Simulation multi-physique pour prévoir la distance d'arrêt du robot
Simulation multi-physique du servomoteur et chariot élévateur pour déplacer le colis

Intégration finale / Programmation H9 H10 H11

Mise en œuvre du programme global de suivi de chemin + gestion des obstacles + dépose du colis (diagramme d'états Stateflow)

Programme global (micro-python) du suivi du chemin et gestion des anomalies

Intégration châssis (Bonus)

Challenge / Concours H12

Exemple: Comment faire varier la vitesse du robot ?

Expérience découverte principe



Théorie PWM

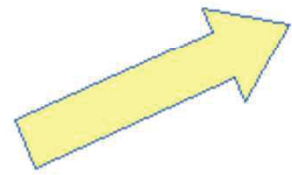
Le réglage d'un moteur, c'est quoi? On ne peut pas l'impressionner. Supposons que l'on dispose un interrupteur S, en série avec le moteur. Si on ferme et ouvre cet interrupteur très rapidement, le moteur n'a pas le temps de s'arrêter mais on constate qu'il tourne plus vite, tantôt on dit que l'interrupteur hache (ou découpe) la tension Vs.

on note souvent cyclique $\frac{t_{on}}{t_{on}+t_{off}}$
 la durée t_{on} et t_{off} la durée $t_{on}+t_{off}$ et 100%

Aux bornes du récepteur, la tension U a l'allure ci-contre. Sa valeur moyenne, notée U_m vaut:
 $U_m = \frac{U_s \cdot t_{on}}{t_{on} + t_{off}}$

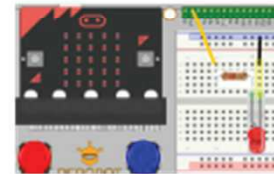
Le hachage permet de faire varier la tension moyenne aux bornes d'un récepteur et la puissance qu'il absorbe en agissant sur PWM (Pulse Width Modulation) ou BLU (Modulation de Largeur d'Impulsion)

C'est comme pour vos notes, si vous n'avez que des 20 votre moyenne est maximum. Si vous avez quelques 10 et quelques 20, la moyenne baisse.

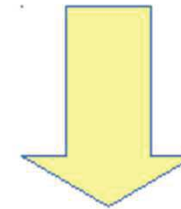
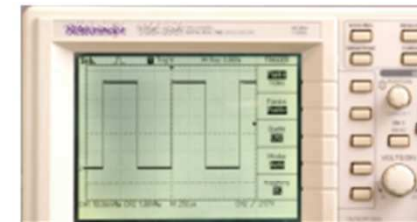


```

14 # Initialisation
15 pin_pwm.set_analog_period(2)
16 valPWM = 0
17 pin_pwm.write_analog(valPWM)
18
19 # Boucle infinie
20 while True:
21     reponse = input("entrer valeur PWM : ")
22     valPWM = int(reponse)
    
```

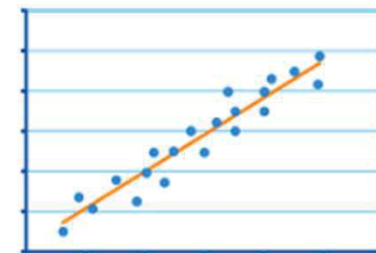
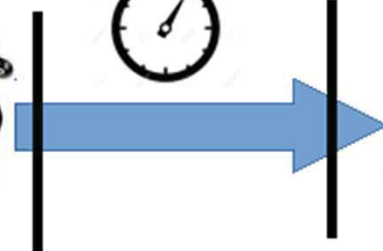
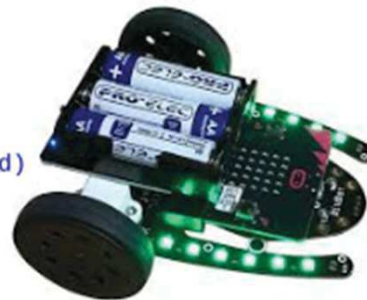


Valider programmation PWM



PWM : Pulse Width Modulation (Modulation par largeur d'impulsion)

speed = 100
 Drive(speed, speed)



Exemple : Valider le modèle de la motorisation

■ 1 guidance



■ des ressources

■ des paramètres à renseigner dans le modèle



■ des paramètres à modifier pour valider le modèle de la motorisation



1^{er} EdS
PROJET ELEVE 1 & 2

Sommaire

1. Déf.
2. Les ressources à disposition
3. Modélisation d'un moteur-ducteur muni de sa roue
 - 3.1 Préparation de la simulation
 - 3.2 Simulation et interprétation
4. Modélisation cinématique du robot sur le sol et simulation
 - 4.1 Préparation de la simulation
 - 4.2 Définition des los de commande des roues afin de suivre une trajectoire
 - 4.3 DÉFI ELEVE 1
 - 4.4 DÉFI ELEVE 2

Première prise en main du modèle

1^{er} EdS
Ressource projet
Page 10 sur 30

Lorsque vous travaillez avec des moteurs, vous aurez besoin d'une source externe pour alimenter les pilotes de moteur et alimenter les moteurs. Vous pouvez le faire en connectant une batterie LiPo au connecteur de batterie ou en utilisant une source d'alimentation externe et en la connectant à l'entrée VIN sur le boîtier. Il est recommandé d'effectuer ces opérations avec l'interrupteur d'alimentation en position OFF. Une fois l'alimentation externe connectée à la carte, l'interrupteur d'alimentation peut être activé.

Dans les projets, nous utiliserons le Motor Carrier avec un MKR1000 en raison des capacités de cette carte. Grâce à une configuration spéciale au sein de MATLAB, la carte peut communiquer directement avec le logiciel tant qu'il se situe sur le même réseau sans fil.



2.6 Contrôle des moteurs

100:1 Micro Metal Gearmotor HP 6V

Pololu Item #: 1101
Brand: Pololu
Status: Active and Preferred
Free shipping in USA
Free add-on shipping in USA

Price break: Unit price (USD)

1	15.95
10	13.50
50	11.96

Quantity: 1
Add to cart

This gearmotor is a miniature high-power, 6 V brushed DC motor with a 100:37:1 metal gear D-shaped gearbox output shaft is 9 mm long and 3 mm in diameter.

Key specifications:

voltage	no-load performance	stall extrapolation
6 V	310 RPM, 70 mA	1.7 kg cm (24 oz in), 1.6 A

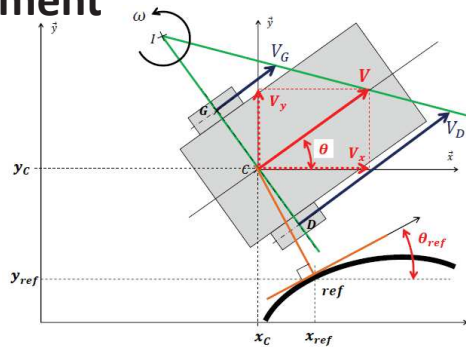
```

Parametre_model.m
7 % voie du robot - distance entre les centres des roues
8 l=28e-3;
9 % masse du robot (à mesurer réellement)
10 masse=0.25;
11 % définition des caractéristiques géométriques et cinétiques du robot 3D
12 % issu de Solidworks (ne rien modifier ici)
13 rover_em_DataFile;
14 %% caractéristiques des moteurs
15 viscous_friction_moteur=4.3e-7;
16 inductance=0.5;
17 inertie_moteur=2e-8;
18 couple_resistant=12e-4;
19 No_Load_Speed=7500;
20 Stall_torque=0.014709975;
21 No_Load_current=90e-3;
22
23 %% Caractéristiques des réducteurs
24 red=100;
    
```

The screenshot shows a Simulink model of a robot's motorization system. It includes two motor control blocks labeled 'ALIMENTATION ROUE DROITE' and 'ALIMENTATION ROUE GAUCHE'. The Simulink interface displays various control elements like gain blocks, summing junctions, and motor blocks. To the right, the Mechanics Explorer window shows a 3D model of the robot with a red wheel and a blue gear. Two graphs show the speed of the right and left motors in rpm over time, both showing a step response from 0 to approximately 3000 rpm. The time is set to T=0.01s.

Valider la stratégie de déplacement

- Renseigner dans le modèle la loi entrée-sortie de la cinématique du robot sur le sol



```

Editor - Bloc: modele_rover_1_PROF/loi de commande consigne en vitesse
Parametre_modele.m loi de commande consigne en vitesse des mote
1
2 function [Wgref,Wdref] = fcn(Vref,Wref,r,b)
3
4 Wdref = (Vref+0.5*(Wref*b))/r;
5 Wgref = (Vref-0.5*(Wref*b))/r;

```

```

%% coordonnées des cibles à atteindre
% coordonnées cible départ
X1=0;
Y1=0;
% coordonnées cible arrivée
X2=0.9;
Y2=0.9;

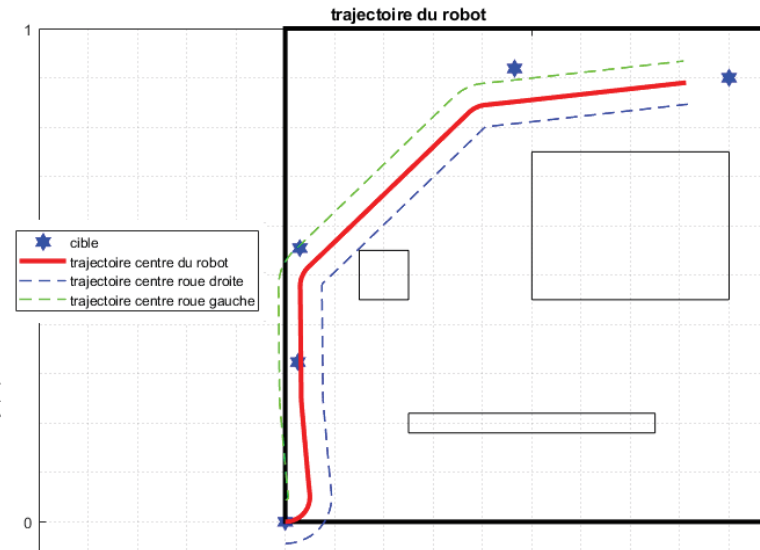
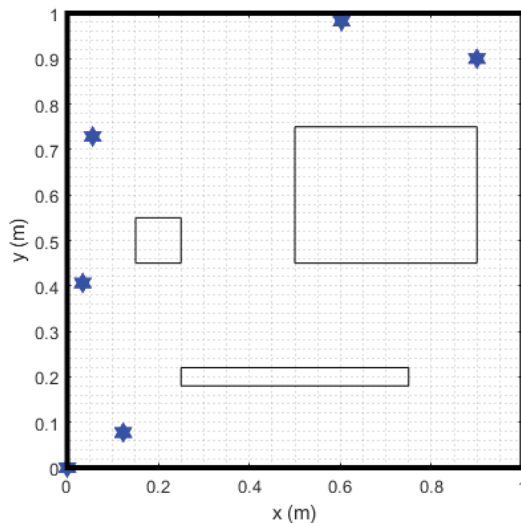
% coordonnées des points de passage
X3=0.2;
Y3=0.5;

% création du tableau de valeur des coordonn
Coordonnee_cible=[X1 Y1 ; X2 Y2];

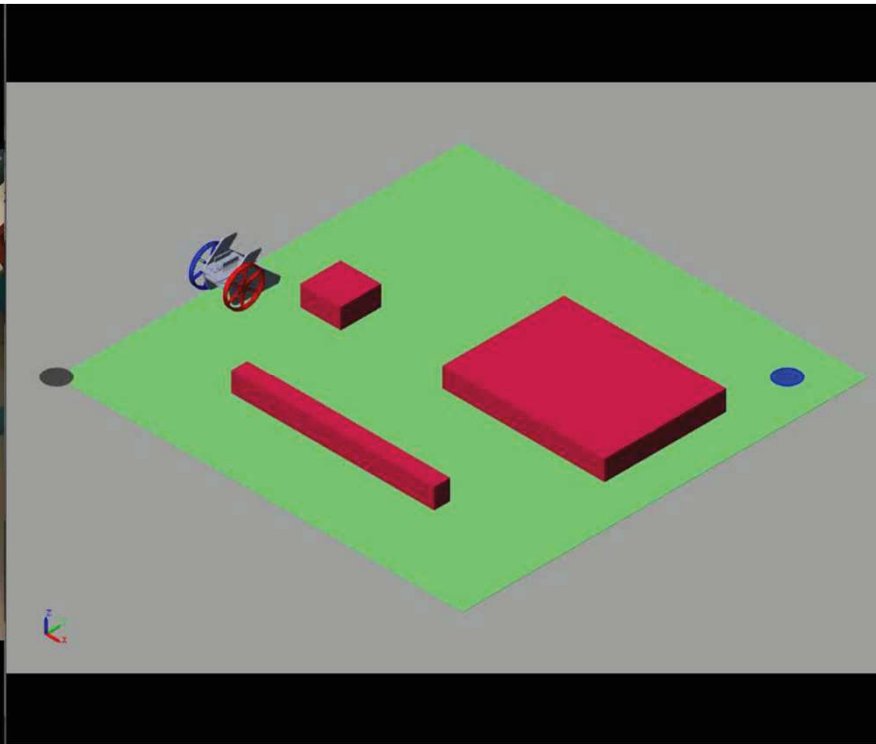
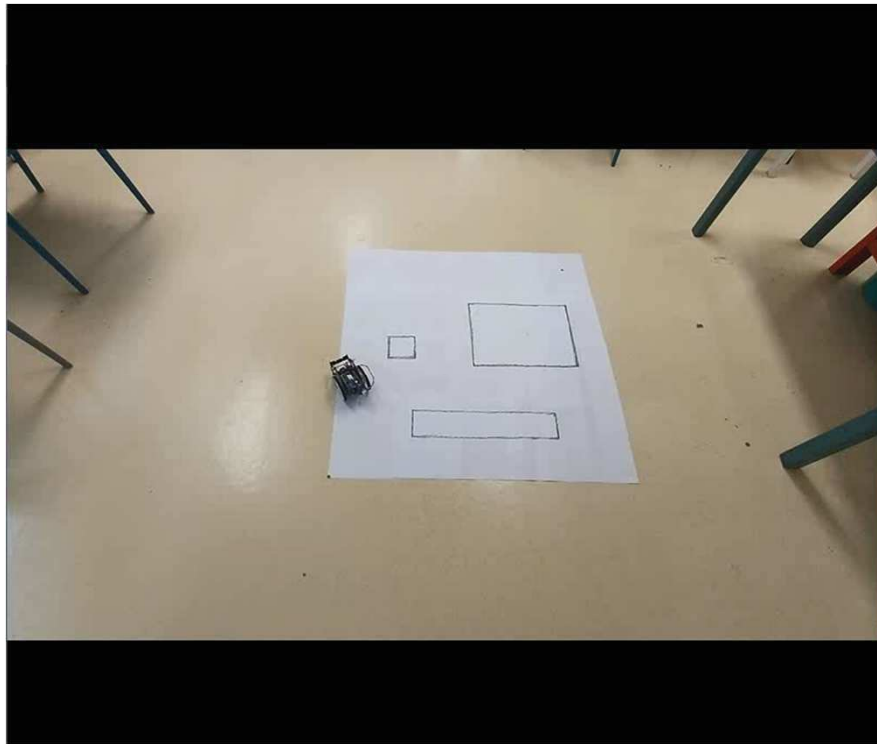
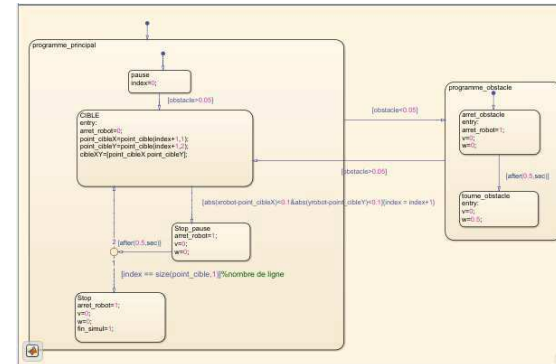
```

- Déterminer le chemin afin de rallier le point de départ au point d'arrivée en évitant les obstacles

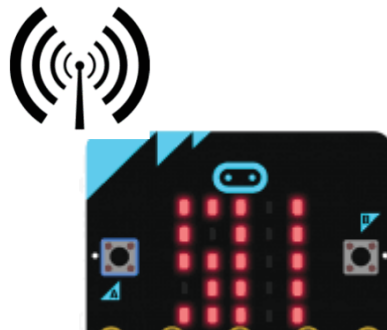
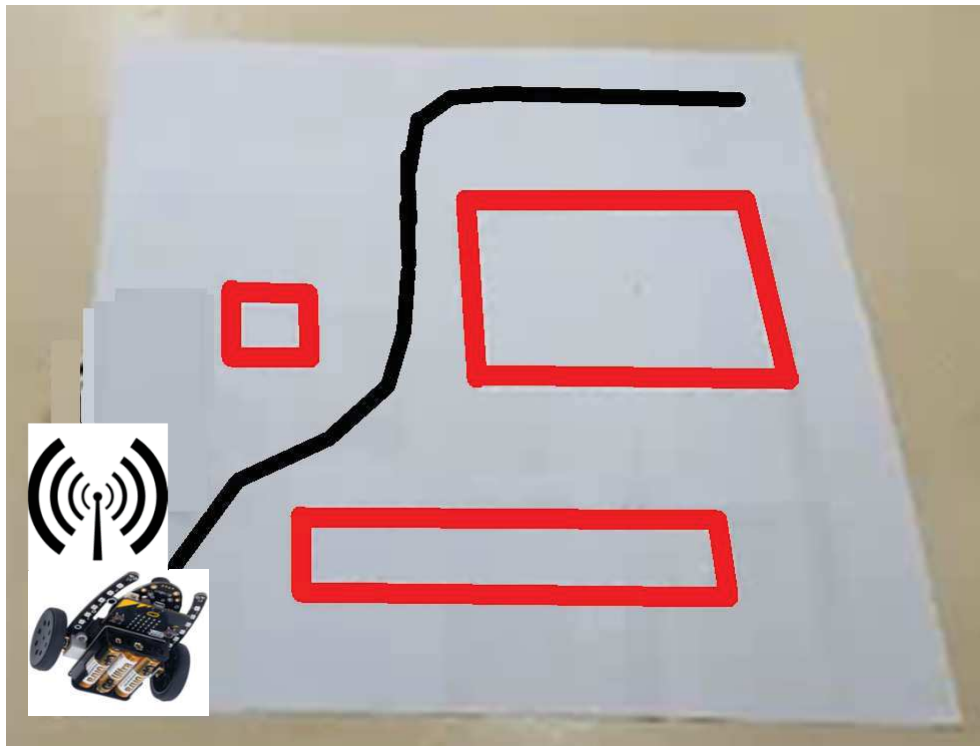
- Simuler le déplacement du robot et valider la stratégie adoptée



- Mise en œuvre du programme global



- Mise en œuvre du programme global



```

Mu 1.1.0.alpha.2 - Challenge12h_SuiviLigne_V05.py
Mode Nouveau Charger Enregistrer Flasher Fichiers REPL Graphique Zoomer Dé-zoomer
PWM v0.py Challenge12h_SuiviLigne_V05.py
67     pwm_left = 1023 + valid_speed(speed_left)
68
69     if speed_right >= 0:
70         dir_right = 0
71         pwm_right = valid_speed(speed_right)
72     else:
73         dir_right = 1
74         pwm_right = 1023 + valid_speed(speed_right)
75     pin_dirMotL.write_digital(dir_left)
76     pin_pwmL.write_analog(pwm_left)
77     pin_dirMotR.write_digital(dir_right)
78     pin_pwmR.write_analog(pwm_right)
79
80     # beep du buzzer de durée "tme"
81     def beep(tme):
82         pin_buzzer.write_digital(1)
83         sleep(tme)
84         pin_buzzer.write_digital(0)
85
86     def clignote():
87         # rouge
88         for pix in range(0, len(np)):
89             np[pix] = (bright, 0, 0)
90         np.show()
91         sleep(500)
92         # vert
93         for pix in range(0, len(np)):
94             np[pix] = (0, bright, 0)
    
```



Compétences générales	Repère	Compétences développées	Connaissances associées	Classe
INNOVER	I3	Imaginer une solution originale, appropriée et esthétique	Cartes heuristiques Scénarios d'usage et expériences utilisateurs Design d'interface et d'interaction	1e
ANALYSER	A6	Analyser le comportement d'un objet à partir d'une description à événements discrets	Diagramme états-transitions	Tale
	A8	Analyser les principaux protocoles pour un réseau de communication et les supports matériels	Algorithme Support filaire et sans fil	Tale 1e
			Écart de performance absolu ou relatif, et interprétations possibles	1e
			Erreurs et précision des mesures expérimentales ou simulées	1e
			Traitement des données : tableaux, graphiques, valeurs moyennes, écarts types, incertitude de mesure	1e
			Modèle associé aux composants élémentaires de transformation, de modulation, de conversion ou de stockage de l'énergie	1e
MODÉLISER ET RÉSOUDRE	M3	Associer un modèle aux composants d'une chaîne de puissance	Comportement séquentiel	1e
	M4	Traduire le comportement attendu ou observé d'un objet	Structures algorithmiques (variables, fonctions, structures séquentielles, itératives, répétitives, conditionnelles)	1e
	M6	Modéliser sous une forme graphique une structure, un mécanisme ou un circuit	Diagramme d'états-transitions	1e
	M13	Déterminer les grandeurs géométriques et cinématiques d'un mécanisme	Circuit électrique	1e
			Positions, vitesses et accélérations linéaire et angulaire sous forme vectorielle	1e
EXPÉRIMENTER ET SIMULER	E1	Prévoir l'ordre de grandeur de la mesure	Gamme d'appareils de mesure et capteurs	1e
	E2	Conduire des essais en toute sécurité à partir d'un protocole expérimental fourni	Règle de raccordement des appareils de mesure et des capteurs	1e
	E4	Instrumenter tout ou partie d'un produit en vue de mesurer les performances	Carte micro - contrôleur	Tale
	E6	Relever les grandeurs caractéristiques d'un protocole de communication	Caractéristiques des signaux	1e
	E7	Modifier les paramètres influents et le programme de commande en vue d'optimiser les performances du produit	Processus itératif d'amélioration des performances	Tale
	E8	Mettre en oeuvre une simulation numérique à partir d'un modèle multi-physique pour qualifier et quantifier les performances d'un objet réel ou imaginé	Paramètres de simulation : durée, incrément temporel, choix des grandeurs affichées, échelles adaptées à l'amplitude et la dynamique des grandeurs simulées	Tale
COMMUNIQUER	C2	Rendre compte de résultats	Tableau, graphique, diaporama, carte mentale	1e
	C3	Collecter et extraire des données Comparer, traiter, organiser et synthétiser les informations pertinentes	ENT, moteurs de recherche, internet, blog, base de données, dossiers techniques	1e
	C6	Travailler de manière collaborative Trouver un tiers expert Collaborer en direct ou sur une plateforme, via un espace de fichiers partagés	Espaces partagés et de stockage, ENT	1e

Intégration finale / Programmation

H5 H6 H7 H8

Équipe

▪ En bonus

Intégration d'une coque sur le châssis afin de protéger les composants

Concours / Challenge

H12

Équipe

▪ Présentation des résultats obtenus

▪ Démonstration du fonctionnement



BILAN

Projet Sciences de l'Ingénieur:
Lycée Pierre-Paul Riquet
Saint Orens

- Nombre d'activités proposés	14
- Nombre d'activités réalisés pour chaque élève	3

Retour des élèves après 2 semaines de projet

- Projet concret et motivant avec un fil conducteur.
- Mise en pratique des cours théoriques.
- Les activités pratiques ont un lien.