

Pourquoi cette UE ?

Ce module offre aux étudiants la possibilité de choisir un cours parmi plusieurs et ainsi d'acquérir des compétences plus spécifiques.

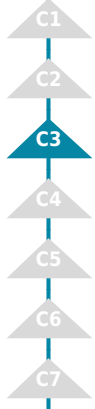
Éléments constitutifs de l'UE

	coefficient	
PRISMsym_10_2-1 Systèmes embarqués	1	
PRISMsym_10_2-2 Développement Android	1	
PRISMsym_10_2-3 Développement LabView	1	
PRISMsym_10_2-4 Traitement d'images numériques	1	
PRISMsym_10_2-5 Robot Operating System (ROS)	1	
Volume d'heures d'enseignement encadré	Volume d'heures de travail personnel	Nombre d'ECTS
200	14	2

Alignement curriculaire

Parmi les compétences visées par la formation, lesquelles sont développées dans cette UE ?

FIG-BC08
PRISM



Contexte et enjeux de l'enseignement

Les systèmes embarqués sont présents dans de nombreux domaines (robotique, avionique, automobile, télécom, domotique...). Ils sont au cœur des systèmes mécatroniques, des procédés qu'ils contrôlent/pilotent. Chaque application nécessite une architecture hardware et software adaptées afin de répondre au mieux aux besoins, comme par ex : la capacité de calcul, les contraintes temps réel, les interfaces d'entrées/sorties (ctrlr/cmde), le traitement du signal (DSP), la gestion de la consommation, la connectivité, besoin ou pas d'un OS, etc. Ils donnent accès à une panoplie extrêmement large de services englobant des fonctions de vision, de localisation, de communication à courte et longues distance, de connexion internet, d'IA... L'objectif du cours est de permettre aux étudiants de mettre en œuvre un robot mobile embarquant entre autres, une carte à microcontrôleur, une carte Linux, dans le cadre d'un projet incluant localisation, visualisation et services web.

Prise en compte des dimensions socio-environnementales

ODD9 - Industrie, innovation et infrastructure

Prérequis

Cours Architecture des microcontrôleurs, Base de la programmation objet (JAVA). Ces cours sont enseignés avant l'EU "Systèmes Embarqués".

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	4
Cours intégré (cours + TD)	
TD	
TP	6
Projets	30
Travail en autonomie encadré	
Contrôles et soutenances	
Travail personnel	

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

Connaitre les architectures et caractéristiques des systèmes embarqués.
 Connaitre les commandes de base sous Linux. Ce cours permet aux élèves d'aller plus loin s'ils le souhaitent.
 Installer et utiliser les outils de développement (VScode, ssh, compilateur JAVA, ressources ...).
 Apprendre à concevoir et développer une application embarquée, depuis un cahier des charges jusqu'à sa validation finale.

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

TP Linux : (6h) .
 Projet de développement d'une application de pilotage (via une connexion WIFI) d'une plateforme mobile (embarquant Linux). (3h).

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Code source de l'application commenté/explicé.
 Un retour personnalisé est fait aux élèves qui le demandent.

Notez que :

- en plus des modalités d'évaluation prévues, des évaluations non inscrites à l'emploi du temps pourront également avoir lieu,
- en cas de dysfonctionnement avéré, les évaluations de groupe peuvent-être individualisées.

Plan de cours

1. Cours d'introduction sur les systèmes embarqués (caractéristiques, architectures ...)
2. Introduction à Linux + TP : les commandes Shell de base.
3. Présentation de l'architecture du robot, des outils de développement (Visual Studio Code, API JAVA PI4), SSH)
4. Projet : développement d'un applicatif embarqué de mise en œuvre d'une plateforme mobile ...

Ressources et références

Cours_SystemesEmbarqués.pdf ; presentationLinux.pdf.
IntroSystèmesTempsReel.pdf et ArchitectureDesFPGA.pdf : non présentés en cours.

Contexte et enjeux de l'enseignement

Les applications nomades se sont développées avec les outils de communication personnels tels que les smartphones ou tablettes. Ce cours électif offre aux élèves qui le suivent la possibilité d'apprendre à développer des applications pour le système d'exploitation Android.

Prise en compte des dimensions socio-environnementales

ODD9 - Industrie, innovation et infrastructure

Prérequis

Bases de la programmation

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	
Cours intégré (cours + TD)	
TD	
TP	
Projets	40
Travail en autonomie encadré	
Contrôles et soutenances	
Travail personnel	

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

Concevoir/développer une application sous environnement Android.
Développer une interface homme machine pour le pilotage d'un robot mobile.

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

La démarche est très pragmatique, et consiste à accompagner les élèves dans le développement d'une application depuis le cahier des charges jusqu'au test de validation finale.

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Rapport du projet + code source de l'application

Notez que :

- en plus des modalités d'évaluation prévues, des évaluations non inscrites à l'emploi du temps pourront également avoir lieu,
- en cas de dysfonctionnement avéré, les évaluations de groupe peuvent-être individualisées.

Plan de cours

- 1) Environnement de développement 'Flutter' connu pour sa capacité à concevoir des applications natives multiplateforme pour Android et iOS.
- 2) Méthodes de développement (création des vues, interfaces de communications...)

Ressources et références

Références sur internet

Contexte et enjeux de l'enseignement

Dans l'automatisation industrielle, la rapidité et l'efficacité dans la conception de systèmes de mesure et de contrôle sont cruciales. Ce cours initie à LabView, un outil reconnu pour son approche graphique et sa robustesse industrielle qui permet de programmer rapidement des systèmes d'acquisitions, de traitement et de pilotage en parallèle d'une IHM.

Prise en compte des dimensions socio-environnementales

ODD9 - Industrie, innovation et infrastructure

Prérequis

Bases de la programmation objet

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	4
Cours intégré (cours + TD)	
TD	
TP	
Projets	36
Travail en autonomie encadré	
Contrôles et soutenances	
Travail personnel	

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

Connaître l'environnement de développement LaBView.
Acquérir des signaux de grandeurs physiques au moyen de carte E/S.
Traiter des données acquises pour les mettre en forme et les sauvegarder.
Communiquer et récupérer des informations par protocole SERIE.
Concevoir une UI esthétique et ergonomique sous LabView.

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

4h de CM
36h de projet découpé en 3 sous projets.

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Code source commenté + manuel d'utilisation du logiciel développé.
Un retour est fait à chaque élèves sur demande.

Notez que :

- en plus des modalités d'évaluation prévues, des évaluations non inscrites à l'emploi du temps pourront également avoir lieu,
- en cas de dysfonctionnement avéré, les évaluations de groupe peuvent-être individualisées.

Plan de cours

Cours d'introduction au développement sur l'environnement LabView.

Projet 1 : Acquérir et traiter un signal généré par un joystick avec une carte d'acquisition E/S (type NI-DAQmx).

Projet 2 : Développer une IHM pour le pilotage d'un robot Zumo avec une communication SERIE (type ARDUINO).

Projet 3 : Combiner les projets 1 et 2 afin de piloter le robot au moyen du joystick. Acquérir les données générées et les traiter.

Ressources et références

Support de cours introduction à LabView.

Référence sur internet.

Contexte et enjeux de l'enseignement

L'objectif de cette UE est de découvrir l'imagerie numérique et ses applications. Malgré le nombre d'heure limité et la densité du cours, un projet sera à présenter lors de la dernière séance. Chaque groupe (1 ou 2 élèves) aura un sujet différent. Ces sujets seront proposés par l'enseignant ; cependant, les étudiants pourront suggérer un projet qui les intéresse s'il est en rapport avec l'UE.

Prise en compte des dimensions socio-environnementales

ODD3 - Bonne santé et bien-être ODD9 - Industrie, innovation et infrastructure
ODD14 - Vie aquatique ODD15 - Vie terrestre

Prérequis

Savoir coder un minimum, traitement de signal.

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	12
Cours intégré (cours + TD)	
TD	12
TP	
Projets	14
Travail en autonomie encadré	
Contrôles et soutenances	2
Travail personnel	14

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

Comme les TD seront sous Matlab, il faut savoir coder un minimum. Le premier TD est consacré à la prise en main de l'outil, donc même une personne novice motivée peut élaborer un projet final exploitable.

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

Les TD seront sous matlab (contenant la toolbox Image Processing).
Les projets varient selon les années.

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Un contrôle final et la présentation d'un projet.

Notez que :

- en plus des modalités d'évaluation prévues, des évaluations non inscrites à l'emploi du temps pourront également avoir lieu,
- en cas de dysfonctionnement avéré, les évaluations de groupe peuvent-être individualisées.

Plan de cours

Les base de l'imagerie numérique
Les histogrammes (seuillage, détection et suivi d'objets...)
Le filtrage des images
Transformée de Fourier rapide et ses applications (si contenu précédent assimilé)
Détection de contours et de points d'intérêts
Projet.

Ressources et références

Les slides de cours seront disponibles
1 TD pour les 6 premières séances (6/10 séances)
L'aide en ligne de matlab peut s'avérer utile pour les TD.

Contexte et enjeux de l'enseignement

L'objectif de ce cours est dans un premier temps d'appréhender les possibilités de ROS et son écosystème, puis d'apporter les bases théoriques initiales utiles à son implémentation dans son projet robotique. Dans un deuxième temps un cas d'application sera mené sur une plateforme robotique.

Prise en compte des dimensions socio-environnementales

ODD9 - Industrie, innovation et infrastructure

Prérequis

Programmation python ou C++, Environnement Linux Cours Systèmes Embarqués

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	
Cours intégré (cours + TD)	
TD	
TP	
Projets	40
Travail en autonomie encadré	
Contrôles et soutenances	
Travail personnel	

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

- Maîtrise de l'outils ROS
- Maîtrise de l'utilisation des ressources mises à disposition par l'écosystème
- Autonomie dans l'analyse, recherche et l'implémentation de solution

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

Le cours est constitué de 12 heures de cours théorique / TD, 24h de TP en semi autonomie en groupe de 3 et 4h de restitution commune avec soutenance.

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

4h de restitution de projet commune avec soutenance.

Notez que :

- en plus des modalités d'évaluation prévues, des évaluations non inscrites à l'emploi du temps pourront également avoir lieu,
- en cas de dysfonctionnement avéré, les évaluations de groupe peuvent-être individualisées.

Plan de cours

- Contexte, principe et architecture
- Outils intégrés (Rviz, Gazebo, bag, tf...)
- Gestion projet ROS (Gitlab)

Ressources et références

Poly de référence disponible au format numérique