الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
People's Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique
Ministry of Higher Education and Scientific Research

LOGO
De l'université

عرض تكوين
ل. م . د

ماستر أكاديمية

برنامج وطني

(تحيين 2026-Bis)

المؤسسة	الكلية/ المعهد	القسم

الميدان	الفرع	التخصص
علوم وتكنولوجيا	الكترونيك	إلكترونيات الأنظمة المضمنة

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
People's Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique
Ministry of Higher Education and Scientific Research

LOGO
De l'université

OFFRE DE FORMATION
L.M.D

MASTER ACADEMIQUE

PROGRAMME NATIONAL

(Mise à jour 2026-Bis)

Etablissement	Faculté / Institut	Département

Domaine	Filière	Spécialité
Sciences et Technologies	*Electronique*	*Electronique des systèmes embarqués*

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
People's Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique
Ministry of Higher Education and Scientific Research

LOGO
De l'université

Training Offer
L.M.D

ACADEMIC MASTER'S DEGREE

NATIONAL PROGRAMME

(2026-Bis Update)

Institution	Faculty / Institute	Department

Domain	Field	Specialisation
Science and Technology	***Electronics***	***Embedded Systems Electronics***

I – Fiche d'identité du Master

Conditions d'accès

(Indiquer les spécialités de licence qui peuvent donner accès au Master)

Filière	Master harmonisé	Licences ouvrant accès au master	Classement selon la compatibilité de la licence	Coefficient affecté à la licence
Electronique	Electronique des systèmes embarqués	Electronique	**1**	**1.00**
		Télécommunications	**2**	**0.80**
		Génie Biomédical	**2**	**0.80**
		Automatique	**3**	**0.70**
		Electrotechnique	**3**	**0.70**
		Electromécanique	**4**	**0.65**
		Autres licences du domaine ST	**5**	**0.60**

II – Fiches d'organisation semestrielles des enseignements de la spécialité

Semestre 1

Unité d'enseignement	Matières	Crédits	Coefficient	Volume horaire hebdomadaire			Volume Horaire Semestriel (15 semaines)	Travail Complémentaire en Consultation (15 semaines)	Mode d'évaluation	
	Intitulé			**Cours**	**TD**	**TP**			**Contrôle Continu**	**Examen**
UE Fondamentale Code : UEF 1.1.1 Crédits : 10 Coefficients : 5	Systèmes à microcontrôleurs	6	3	3h00	1h30		67h30	82h30	40%	60%
	Electronique numérique avancée : FPGA et VHDL	4	2	1h30	1h30		45h00	55h00	40%	60%
UE Fondamentale Code : UEF 1.1.2 Crédits : 8 Coefficients : 4	Traitement avancé du signal	4	2	1h30	1h30		45h00	55h00	40%	60%
	Systèmes asservis numériques	4	2	1h30	1h30		45h00	55h00	40%	60%
UE Méthodologique Code : UEM 1.1 Crédits : 11 Coefficients : 7	TP Systèmes à microcontrôleurs	2	1			1h30	22h30	27h30	100%	
	TP FPGA et VHDL	2	1			1h30	22h30	27h30	100%	
	TP Traitement avancé du signal /TP Systèmes asservis numériques	2	1			1h30	22h30	27h30	100%	
	Programmation C++ embarquée	3	2	1h30		**1h00**	**37h30**	**37h30**	40%	60%
	Programmation avancée en Python	2	2	1h30		1h30	45h00	5h00	40%	60%
UE Découverte Code : UED 1.1 Crédits : 1 Coefficients : 1	**Respect des normes et des règles d'éthique et d'intégrité**	**1**	**1**	**1h30**			**22h30**	**02h30**		**100%**
Total semestre 1		**30**	**17**	**12h00**	**6h00**	**7h00**	**375h00**	**375h00**		

Semestre 2

Unité d'enseignement	Matières	Crédits	Coefficient	Volume horaire hebdomadaire			Volume Horaire Semestriel (15 semaines)	Travail Complémentaire en Consultation (15 semaines)	Mode d'évaluation	
	Intitulé			**Cours**	**TD**	**TP**			**Contrôle Continu**	**Examen**
UE Fondamentale Code : UEF 1.2.1 Crédits : 10 Coefficients : 5	Architecture des Processeurs pour l'Embarqué	6	3	3h00	1h30		67h30	82h30	40%	60%
	Processeurs des Signaux Numériques (DSP)	4	2	1h30	1h30		45h00	55h00	40%	60%
UE Fondamentale Code : UEF 1.2.2 Crédits : 8 Coefficients : 4	Système de Vision Artificielle	4	2	1h30		**1h30**	**45h00**	**55h00**	40%	60%
	Automates Programmables Industriels.	4	2	1h30	1h30		45h00	55h00	40%	60%
UE Méthodologique Code : UEM 1.2 Crédits : 8 Coefficients : 4	TP Architecture des Processeurs pour l'Embarqué / TP Processeurs des signaux numériques	3	1			**2h00**	**30h00**	**45h00**	100%	
	TP Automates programmables industriels	2	1			**1h00**	**15h00**	**35h00**	100%	
	Programmation Java pour les systèmes embarqués	3	2	1h30		**1h00**	**37h30**	**37h00**	40%	60%
UE Transversale Code : UET 1.2 Crédits : 4 Coefficients : 4	Eléments d'IA appliquée	2	2	1h30	1h30		45h00	5h00	40%	60%
	Lean start-up	**2**	**2**	**1h30**	**1h30**		**45h00**	**05h00**	**80% (20% Atl + 60% Prj)**	**20%**
Total semestre 2		**30**	**17**	**12h00**	**6h00**	**7h00**	**375h00**	**375h00**		

Semestre 3

Unité d'enseignement	Matières	Crédits	Coefficient	Volume horaire hebdomadaire			Volume Horaire Semestriel (15 semaines)	Travail Complémentaire en Consultation (15 semaines)	Mode d'évaluation	
	Intitulé			**Cours**	**TD**	**TP**			**Contrôle Continu**	**Examen**
UE Fondamentale Code : UEF 2.1.1 Crédits : 10 Coefficients : 5	Systèmes Temps Réel	6	3	3h00	1h30		67h30	82h00	40%	60%
	Intelligence Artificielle en Embarqué	4	2	1h30	1h30		45h00	55h00	40%	60%
UE Fondamentale Code : UEF 2.1.2 Crédits : 8 Coefficients : 4	Systèmes embarqués	4	2	1h30	1h30		45h00	55h00	40%	60%
	Réseaux et communications industriels	4	2	1h30	1h30		45h00	55h00	40%	60%
UE Méthodologique Code : UEM 2.1 Crédits : 9 Coefficients : 5	TP Intelligence artificielle TP	2	1			1h30	22h30	27h30	100%	
	TP Systèmes embarqués/ TP Systèmes Temps Réel	2	1			1h30	22h30	27h30	100%	
	TP Réseaux industriels	2	1			1h30	22h30	27h30	100%	
	Etude et Réalisation des projets	3	2	1h30		1h30	45h00	30h00	40%	60%
UE Transversale Code : UET 2.1 Crédits : 3 Coefficients : 3	Reserve engineering	2	2	1h30	1h30 Atelier		45h00	5h00	40%	60%
	Recherche documentaire et conception de mémoire	1	1	**1h00**			**15h00**	**10h00**		100%
Total semestre 3		**30**	**17**	**11h30**	**6h00**	**7h30**	**375h00**	**375h00**		

Orientations générales sur le choix des matières de découverte :

Les matières découvertes dans le Référentiel des Matières du Master ''Instrumentation'' (Tableau ci-dessus) sont laissées au libre choix des établissements qui peuvent choisir indifféremment leurs matières parmi la liste présentée ci-dessous en fonction de leurs priorités.

Matières avec programmes détaillés :

- Optoélectronique
- Systèmes énergétiques autonomes
- Electroacoustique et analyses vibratoires
- Compatibilité électromagnétique
- Instrumentation et mesure industrielles
- Sécurité industrielle
- Robotique
- Réglage des Entrainements Electriques
- Bio instrumentation et biocapteurs
- Méthodes et outils pour le control non destructif
- Outils pour la maintenance en instrumentation
- Maintenance industrielle et diagnostic
- Réseaux et communication industriels

Autres matières laissées au libre choix des établissements (programmes ouverts après validation du CPND)

- Systèmes d'affichage(Découverte)
- Instruments de mesure(Découverte)
- Mesures en haute fréquence(Découverte)
- Electroacoustique, son et HIFI(Découverte)
- Télégestion industrielle (SCADA)(Découverte)
- Théorie de la commande des systèmes industriels (Découverte)
- Capteurs intelligents en instrumentation industrielle(Découverte)
- Recherche opérationnelle

Semestre 4

Stage en entreprise ou dans un laboratoire de recherche sanctionné par un mémoire et une soutenance.

	VHS	Coeff	Crédits
Travail Personnel	550	09	18
Stage en entreprise ou dans un laboratoire	100	04	06
Séminaires	50	02	03
Autre (Encadrement)	50	02	03
Total Semestre 4	750	17	30

Ce tableau est donné à titre indicatif

Evaluation du Projet de Fin de Cycle de Master

- Valeur scientifique (Appréciation du jury) /6
- Rédaction du Mémoire (Appréciation du jury) /4
- Présentation et réponse aux questions (Appréciation du jury) /4
- Appréciation de l'encadreur /3
- Présentation du rapport de stage (Appréciation du jury) /3

III - Programme détaillé par matière du semestre S1

Semestre : 1
Unité d'enseignement : UEF 1.1.1
Matière 1 : Systèmes à Microcontrôleurs
VHS : 67h30 (Cours : 3h00, TD : 1h30)
Crédits : 6
Coefficient : 3

Objectifs à atteindre :

Cette matière permet aux étudiants de se familiariser avec les différentes fonctionnalités offertes par les microcontrôleurs dans le monde réel. En effet, le monde s'oriente vers l'aspect de l'intelligence artificielle qui nécessite des conceptions impliquant des microcontrôleurs comme pièces maîtresses pour réaliser des applications de taille industrielle préconisées notamment dans le système embarqué. Ainsi, les étudiants apprendront à manipuler les différents dispositifs d'entrées/sorties, en l'occurrence : les entrées tout ou rien, les modules de capteurs, les écrans de supervision, les moteurs et les actionneurs, ainsi que l'établissement d'une couche de communication appropriée, à la fois simples comme ceux utilisant des protocoles standardisés, UART, I^2C, SPI, Bluetooth, et des protocoles complexes, à savoir : Modbus TCP/IP, MQTT, ESP-NOW, Zigbee . L'apprentissage de base se fait avec le fameux microcontrôleur "ATmega328p" compte tenu du panel d'applications qui peut être proposé à l'utilisateur et qui va de pair avec les plateformes Arduino. En revanche, il a de meilleures performances que celles offertes par les anciens microcontrôleurs telle que la famille "PIC-16Fxx", de plus, il se prête bien à accepter les différents types de compilateurs, notamment le C-Embarqué dans l'environnement convivial appelé "IDE". Ceci étant dit, il permet aux étudiants d'évoluer plus facilement de manière ludique vers les nouvelles générations de microcontrôleurs comme par exemple l'ESP32 ou l'ARM32, nécessitant ainsi un bagage très conséquent en matière de la programmation avancée.
De plus, le logiciel Proteus® permet une simulation interactive en circuit de ces microcontrôleurs "AVR", et des tests explicites du code et du schéma de circuit avant de penser à construire le matériel cible. Lorsque l'application fonctionne correctement, comme il se doit, en mode simulation, le PCB peut être conçu pour valider la conception en question.
La mise en application du contenu de cette matière sera effectuée au sein de l'unité :
"UEM1.1 : TP Systèmes à microcontrôleurs".

Prérequis recommandés :

Systèmes à microprocesseurs du L3, Electronique numérique, Machine à états finis, Electronique de base, Langage assembleur, Langage C.

Contenu de la matière :

Chapitre 1 Initiation de base du microcontrôleur (5 semaines)

- Architecture générale du microcontrôleur AVR32/PIC32.
- Setting a bit, clearing a bit, checking a bit, Toggling a bit and Macros.
- Architecture de la mémoire AVR (Flash, Boot section; Data memory, RAM, General Registers; EEPROM).

- C-Embarqué dans l'environnement IDE
- L'ossature générale d'un programme Arduino (En-tête, section déclarative, configuration et initialisation, Programme principal et définition des sous-programmes)
- Instructions de bases (Typedef, enum, If-Else, Switch-Case, Struct, Union)
- Applications (Menu déroulant avec trois boutons Up, Down, OK).
- Gestions des sorties numériques (clignotement des leds en utilisant delay() and millis())
- Affichage des données (TM1637 4-digit 7-segment, LCD 2x16)

Chapitre 2. Fonctionnalités avancées des microcontrôleurs (5 semaines)

- Les interruptions occasionnées par des évènements externes (Rising, falling, change).
- Conversion analogique numérique par interruption programmée.
- Interruptions des timers : Timer0, Timer1 et Timer2.
- Générateur MLI par interruption et gestion des Timers (CTC mode et générateur PWM).
- Exemples : encodeur incrémentale dédié à la gestion d'un menu déroulant
- Gradateur de puissance AC avec un relais statique (SSR)
- Exemple : commande d'un servo-moteur avec le mode OC1A/OC1B pour différentes résolutions angulaires.
- Exemple : variateur de la vitesse d'un moteur DC avec le module L298N.
- Utilisation de l'EEPROM interne (Read, Write, Erase).
- Protocoles de communication en mode d'interruption half and full-duplex : UART (manipulation des données type string, byte, int, double)
- Bus I^2C implémentation et utilisation (capteur de température DH22, DS18b20, ds3231)
- Le protocole SPI
- Communication en Full-duplex
- Architecture maître unique – esclave multiple
- Communication à grande vitesse [Hz, kHz ou MHz]
- Configurations maître-esclave.
- Protocole flexible.
- Communication à courte distance.
- Initialisation d'une carte SD et stockage de données.

Chapitre 3. Programmation structurée à haut niveau (5 semaines)

- Systèmes Multi-tâches (pointeurs sur fonctions, implémentation une machine à état fini "FSM" exemple introductif de la gestion d'un feu de signalisation tricolores à deux voies)
- Communication avec Bluetooth (HC-05) en utilisant « MIT APP- Inventor » sur Android et la plateforme Nano possédant le microcontrôleur Atmega328P).
- Le bus CAN est ses applications (conception d'un réseau, exemple tableau de bord d'un véhicule).
- Le bus de terrain Modbus TCP/IP exemple la station S7-1200 avec UNO dotée du module Ethernet Shield.

Mode d'évaluation :

Examen : **60%**, Contrôle Continue : **40%**

Références bibliographiques :

[1] . Arduino Workshop: A Hands-On Introduction with 65 Projects; De John Boxall; No Starch Press 2013.

[2] . C Programming for the Pc the Mac and the Arduino Microcontroller System; De Peter D Minns; AuthorHouse-2013

[3] .Raspberry Pi for Arduino Users: Building IoT and Network Applications and Devices; De James R. Strickland; Apress-2018

[4] .Practical AVR Microcontrollers: Games, Gadgets, and Home Automation with the microcontroller used in the Arduino; De Alan Trevennor; Apress-2012

[5] .Advances in Smart System Technologies: Select Proceedings of ICFSST 2019; Publié par P. Suresh, U. Saravanakumar, Mohammed Saleh Hussein Al Salameh ; Springer Nature.

[6] .INTRENET OF THINGS WITH ARDUINO AND BOLD IOT: With Arduino and Bolt; De Ashwin Pajankar ; Publié par BPB Publications 2018.

[7] .Arduino: A Technical Reference: A Handbook for Technicians, Engineers, and Makers; De J. M. Hughes; "O'Reilly Media, Inc." 2016.

[8] .Building Arduino PLCs: The essential techniques you need to develop Arduino-based PLCs; De Pradeeka Seneviratne; Apress-2017.

[9] .Arduino Robot Bonanza; De Gordon McComb; McGraw Hill Professional-2013.

Semestre: 1
Unité d'enseignement: UEF 1.1.1
Matière 2:Electronique Numérique Avancée : FPGA et VHDL
VHS: 45h00 (Cours: 1h30, TD: 1h30)
Crédits: 4
Coefficient: 2

Objectifs de l'enseignement :

Dans cette matière, les étudiants auront à étudier les différents types de circuits programmables, ainsi que les différentes méthodes de conception en particulier la programmation en utilisant les langages de description matérielle.
L'application du contenu de cette matière s'effectue au niveau de la matière « **UEM1.1 : TP FPGA & VHDL** ».

Connaissances préalables recommandées :

Electronique numérique (combinatoire et séquentielle).

Contenu de la matière:

Chapitre 1. Notion de base sur les circuits programmables (1 semaine)

- Architecture générale des circuits logiques programmables : PAL, GAL, PLD, CPLD
- Exemples de constructeurs et outils de programmation : Altera Quartus II, Xilinx ISE

Chapitre 2. Programmation en VHDL (5 semaines)

- Historique du VHDL.
- Comparaison entre le VHDL et les langages de programmation.
- Différentes descriptions d'une architecture : flot de données, comportemental, structurel.
- Identificateurs et sensibilité à la majuscule.
- Commentaires.
- Représentation des nombres en VHDL
- Structure générale d'un code VHDL : Bibliothèque, Entité, Ports, Architecture.
- Types de données : prédéfinis, définis par l'utilisateur
- Opérateurs : logique, relationnel, de décalage, de concaténation
- Attributs des signaux : EVENT, …
- Signal, variable et constant
- Process
- Component
- Instruction IF-THEN-ELSE
- Instruction CASE-WHEN
- Instruction WHEN-ELSE
- Instruction WITH-SELECT-WHEN

Chapitre 3. Applications sur les circuits FPGA (5 semaines)

- Multiplexeur
- Bascule D
- Additionneur
- Compteur universel avec actions : activation, remise-à-zéro, charge (load).
- Diviseur de fréquence.
- Gestion de la fréquence avec des boutons : sélection, division
- Décodeur 7 segments,

- Affichage série sur plusieurs 7 segments.
- Unité arithmétique-logique 8-bit
- Comparateur 8-bits

Chapitre 4. Conception avancée avec les machines à nombre d'états fini (FSM) (4 semaines)

- Introduction : structure de Mealy et Moore
- Représentation d'une machine FSM
- Exemples de conception de FSM

Mode d'évaluation:

Examen : **60%**, Contrôle Continue : **40%**

Références bibliographiques:

[1] .Volnei A. Pedroni, « Circuit Design with VHDL », MIT Press, 2004.

[2] . Volnei A. Pedroni, « Circuit Design and Simulation with VHDL », 2ème édition, MIT Press, 2010.

[3] . Bryan Mealy, Fabrizio Tappero, « Free Range VHDL », 2018

[4] . Pong P. Chu, « FPGA prototyping by vhdl examples : Xilinx Spartan™-3 Version », John Wiley & Sons, 2008.

[5] . Jacques Weber , Sébastien Moutault, Maurice Meaudre, « Le langage VHDL : du langage au circuit, du circuit au langage », Dunod, 2007.

[6] . Christian Tavernier, « Circuits logiques programmables », Dunod 1992.

Semestre: 1
Unité d'enseignement: UEF 1.1.2
Matière 3:Traitement avancé du signal
VHS: 45h00 (Cours: 1h30, TD: 1h30)
Crédits: 4
Coefficient: 2

Objectifs de l'enseignement :

L'étudiant reçoit les notions de base qui lui permettent de comprendre et d'appliquer des méthodes de traitement de signal concernant les signaux aléatoires et les filtres numériques.

Connaissances préalables recommandées :

Des connaissances sur le traitement numérique des signaux déterministes et les probabilités sont nécessaires pour suivre cette matière. Ces connaissances sont dispensées au niveau de la troisième année licence Electronique.

Contenu de la matière :

Chapitre 1 : Rappels sur les filtres numériques (RIF et RII) (3 semaines)

- Transformée en Z
- Structures, fonctions de transfert, stabilité et implémentation des filtres numériques (RIF et RII)
- Filtre numérique à minimum de phase
- Les méthodes de synthèses des filtres RIF et des filtres RII
- Filtres numériques Multicadences

Chapitre 2 : Signaux aléatoires et processus stochastiques (4 Semaines)

- Rappel sur les processus aléatoires
- Stationnarité
- Densité spectrale de puissance
- Filtre adapté, filtre de Wiener
- Périodogramme, corrélogramme, périodogramme moyenné, périodogramme lissé
- Notions de processus stochastiques
- Stationnarités au sens large et strict et Ergodicité
- Exemples de processus stochastiques (processus de Poisson, processus gaussien et processus Markovien)
- Statistiques d'ordre supérieur (Moments et cumulants, Polyspectres, processus non gaussiens, traitements non linéaires)
- Introduction au filtrage particulaire

Chapitre 3: Analyse spectrale paramétrique et filtrage numérique adaptatif (4 semaines)

- Méthodes paramétriques
- Modèle AR (Lévinson, Yulewalker, Burg, Pisarenko, Music …)
- Modèle ARMA
- Algorithme du gradient stochastique LMS
- Algorithme des moindres carrés récursifs RLS

Chapitre 4 : Analyse temps-fréquence et temps-échelle (4 semaines)

- Dualité temps-fréquence
- Transformée de Fourier à court terme
- Ondelettes continues, discrètes etondelettes dyadiques
- Analyse multi-résolution et bases d'ondelettes
- Transformée de Wigner-Ville
- Analyse Temps-Echelle.

Mode d'évaluation :

Contrôle continu: 40% ; Examen: 60%.

Références bibliographiques :

[1] .Mori Yvon, "Signaux aléatoires et processus stochastiques", Lavoisier, 2014.
[2] .N. Hermann, "Probabilités de l'ingénieur : variables aléatoires et simulations Bouleau", 2002.
[3] .M. Kunt, "Traitement Numérique des Signaux", Dunod, Paris, 1981.
[4] .J. M Brossier, "Signal et Communications Numériques, Collection Traitement de Signal", Hermès, Paris, 1997.
[5] .M. Bellanger, "Traitement numérique du signal : Théorie et pratique", 8e édition, Dunod, 2006.

Semestre: 1
Unité d'enseignement: UEF 1.1.2
Matière 4:Systèmes Asservis Numériques
VHS: 45h00 (Cours: 1h30, TD: 1h30)
Crédits: 4
Coefficient: 2

Objectifs de l'enseignement:

Introduire les propriétés et les représentations des systèmes dynamiques linéaires à tempsdiscret. Donner les éléments fondamentaux de la commande des systèmes linéairesreprésentés sous forme de fonction de transfert en *Z*. Présenter les différentes méthodes desynthèse de correcteurs à temps discrets.

Connaissances préalables recommandées:

Analyse temporelle et fréquentielle des systèmes asservis continus, Représentations graphique et d'état,Synthèse de correcteur.

Contenu de la matière:

Chapitre 1 :Etude de l'échantillonnage d'un signal (5 Semaines)
Transformée en Z et transformée en Z modifiée : Théorème de Shannon, bloqueurs d'ordre zéro et d'ordre un, propriétés de la transformée en Z, Aperçu sur la transformée en Z modifiée et ses propriétés,… Théorème de la valeur initiale et de la valeur finale d'un système échantillonné
Transferts échantillonnés, et équation aux récurrentes : Discrétisation d'un transfert continu, Représentation des systèmes discrets par des équations de récurrences, Propriétés, …
Aperçu sur la transformation bilinéaire d'un transfert échantillonné : Relation entre l'asservissement des systèmes continus et l'asservissement des systèmes échantillonnés (étude de la stabilité d'un système échantillonné par le critère de Routh, …).

Chapitre 2 :Analyse des systèmes échantillonnés dans l'espace d'état (5 Semaines)
Discrétisation de l'équation d'état d'un système continu : Relation entre l'équation d'état d'un système continu et celle d'un système discret.
Représentation et résolution de l'équation d'état d'un système discret : Différentes formes de la matrice d'évolution (diagonale, compagne, observateur, contrôleur, observabilité et contrôlabilité).
Stabilité et précision d'un système discret : Racines de l'équation caractéristique, modes contrôlables, modes observables à partir de la représentation d'état des systèmes échantillonnés, Réponses d'un système échantillonné, Examen de stabilité par le critère de Jurry, …
Notions de gouvernabilité et d'observabilité pour les systèmes SISO et MIMO.

Chapitre 3. Synthèse d'un contrôleur (5 Semaines)
Placement des pôles par retour d'état et par retour de sortie : synthèse de lois de commande simples
Estimateur d'état et de sortie : Cas états du système inaccessibles
Autres méthodes de synthèse : contrôleur PID numérique (structure à 1 degré de liberté), contrôleur RST (structure à 2 degrés de liberté).

Mode d'évaluation:

Contrôle continu: 40% ; Examen: 60%.

Références bibliographiques :

[1] .L. Maret, Régulation Automatique, 1987.
[2] .Dorf & Bishop, Modern Control Systems, Addison-Wesley, 1995

[3] .J. L Abatut, Systèmes et Asservissement Linéaires Echantillonnés, Edition Dunod
[4] .J. Ragot, M. Roesch, Exercices et Problèmes d'Automatique, Edition Masson.
[5] .J. Mainguenaud, Cours d'automatique Tome3, Edition Masson.
[6] .T.J. Katsuhiko, Modern Control Engineering, 5th Edition, Prentice Hall.
[7] .H. Buhler, Réglages Echantillonnés Tome 1, Edition Dunod.
[8] .M. Rivoire, Cours d'Automatique Tome 2, Edition Chihab.
[9] .Th. Kailath, Linear Systems, Prentice-Hall, 1980.

Semestre : 1
Unité d'enseignement : UEM 1.1
Matière 1 : TP Systèmes à microcontrôleurs
VHS: 22h30 (TP: 1h30)
Crédits: 2
Coefficient: 1

Objectifs :

Cette matière est un complément destiné particulièrement à étayer et à bien appréhender la matière précitée **"UEF1.1.1 : Systèmes à microcontrôleurs".** Par ailleurs, elle offre aux étudiants la possibilité de manipuler d'une façon palpable les connaissances acquises à travers des exemples bien ciblés. En outre, il est préférable que l'étudiant puisse tirer parti des conceptions basées sur le microcontrôleur ATmega328P pour créer des applications fiables. On peut se donner pour objectif principal de permettre aux étudiants de créer leurs propres applications en fonction des exigences imposées par la conception elle-même.

Prérequis recommandés :
Systèmes à microprocesseurs, Electronique numérique, FPGA, électronique de base, C embarqué, Grafcet.

Contenu principal :

TP 1 : Tutoriel et prise en main de l'environnement IDE et Proteus (Instruction de base : If-Else, Switch-case, delay(), millis(), typedef, enum, struct, union).

TP 2 : Interruptions externes avec deux boutons poussoirs (pilotage de quatre leds avec des registres à décalage dans les deux sens).

TP 3 : Implémentation d'une machine à état fini en utilisant un simple bouton poussoir pour commander en PWM un moteur DC avec un pont-H à partir d'un module Analogique.

TP 4 : Conception et réalisation d'un gradateur de puissance (relais statique SSR) via l'interruption de Timer1 et l'interruption externe pour détecter le signal du passage par zéro.

TP 5 : Gestion d'un menu déroulant avec trois boutons poussoirs et affichage LCD2x16.

TP 6 : Commande d'un moteur pas-à-pas via Timer1 en mode CTC "*Clear* Timer on *Compare*".

TP 7 : Interfaçage ADC avec les amplificateurs opérationnels et les capteurs analogiques (Manipulation des registres internes).

TP 8 : Gestions multitâches en utilisant un pointeur sur fonction et l'interruption timer1 pour gérer le temps de défilement d'un groupe des diodes Leds.

TP 9 : Lecture/écriture des données d'une EEPROM interne et une EEPROM externes de type 24lc256 en utilisant le bus I^2C.

TP 10 : Commande d'un moteur de type DC via le module Bluetooth "**HC-05"** en utilisant l'application MIT App Inventor «**Android »** .

TP 11 : Implémentation d'un contrôleur PID dédié à la régulation de la température en utilisant le capteur DS 18b20 et un relais statique SSR précité.

TP 12 : Echanges des données entre des plateformes de type Arduino (Slave) UNO et ESP32 (Master) en utilisant le Protocole de communication UART.

TP 13 : Echanges des données entre des plateformes de type Arduino (Clients) UNO et Raspberry pi 4 (Broker) en utilisant le protocole MQTT.

TP 14 : Echange des données entre Arduino et un Automate de type S7-1200 via le protocole Modbus TCP/IP pour commander un variateur de vitesse de type Micro master 420.

Mode d'évaluation :

Contrôle Continue : **100%**

Références bibliographiques :

[1] .Arduino Workshop: A Hands-On Introduction with 65 Projects; De John Boxall; No Starch Press 2013.

[2] .C Programming for the Pc the Mac and the Arduino Microcontroller System; De Peter D Minns; AuthorHouse-2013

[3] .Raspberry Pi for Arduino Users: Building IoT and Network Applications and Devices; De James R. Strickland; Apress-2018

[4] .Practical AVR Microcontrollers: Games, Gadgets, and Home Automation with the microcontroller used in the Arduino; De Alan Trevennor; Apress-2012

[5] .Advances in Smart System Technologies: Select Proceedings of ICFSST 2019;

[6] .Publié par P. Suresh, U. Saravanakumar, Mohammed Saleh Hussein Al Salameh ; Springer Nature.

[7] .INTRENET OF THINGS WITH ARDUINO AND BOLD IOT: With Arduino and Bolt;

[8] .De Ashwin Pajankar ; Publié par BPB Publications 2018.

[9] .Arduino: A Technical Reference: A Handbook for Technicians, Engineers, and Makers; De J. M. Hughes; "O'Reilly Media, Inc." 2016.

[10] . Building Arduino PLCs: The essential techniques you need to develop Arduino-based PLCs; De Pradeeka Seneviratne; Apress-2017.

[11] . Arduino Robot Bonanza; De Gordon McComb; McGraw Hill Professional-2013

[12] . Arduino Sketch for ESP32 Development Workshop; Agus Kurniawan; Publié par PE Press-2018.

Semestre: 1
Unité d'enseignement: UEM 1.1
Matière2 :TP FPGA et VHDL
VHS: 22h30 (TP: 1h30)
Crédits: 2
Coefficient: 1

Objectifs de l'enseignement :

Cette matière est construite autour de la matière « **UEF1.1.1 : Electronique numérique avancée : FPGA et VHDL**» ; elle permet aux étudiants d'appliquer les connaissances acquises sous forme d'exemples précis.
Cette matière permet à l'étudiant de concevoir un système électronique en utilisant le langage de description VHDL et de tester chaque conception sur une carte FPGA.

Connaissances préalables recommandées :
Logique combinatoire et séquentielle.

Contenu de la matière:

TP1 : Présentation de l'outil de développement et de simulation : Altera Quartus II ou Xilinx ISE.
TP2 : Exploitation de la carte de développement à travers un exemple donnée d'un additionneur.
TP3 : Premiers exemples de circuit : multiplexeur, bascule D.
TP4 : Compteur décimal 48 bits simple.
TP5 : Compteur décimal 48 bits avec actions : activation, remise-à-zéro, charge.
TP6 : Feu tricolore de circulation routière.
TP7 : Multiplicateur/diviseur avec les registres à décalage.
TP8 : Affichage série sur plusieurs 7 segments.
TP9 : Horloge numérique.
TP10 : Division de fréquence.
TP11 : Division de fréquence commandable avec des boutons.
TP12 : Affichage VGA.

Mode d'évaluation:
Contrôle Continue : **100%**

Références bibliographiques:
[1] .Volnei A. Pedroni, « Circuit Design with VHDL », MIT Press, 2004.
[2] .Volnei A. Pedroni, « Circuit Design and Simulation with VHDL », 2ème édition, MIT Press, 2010.
[3] .Bryan Mealy, Fabrizio Tappero, « Free Range VHDL », 2018
[4] .Pong P. Chu, « FPGA prototyping by vhdl examples : Xilinx Spartan™-3 Version », John Wiley & Sons, 2008.
[5] .Jacques Weber , Sébastien Moutault, Maurice Meaudre, « Le langage VHDL : du langage au circuit, du circuit au langage », Dunod, 2007.
[6] .Christian Tavernier, « Circuits logiques programmables », Dunod 1992.

Semestre: 1
Unité d'enseignement: UEM1.1
Matière 3:TP Traitement avancé du signal/TP Systèmes asservis numériques
VHS: 22h30 (TP: 1h30)
Crédits: 2
Coefficient: 1

Objectifs de l'enseignement :

Travaux pratiques réalisés sous MATLAB pour donner un aspect pratique à des notions théoriques complexes.

Connaissances préalables recommandées :

Mathématiques (Théorie et calcul des probabilités, Analyse complexe)- Théorie du signal déterministe, Probabilités et statistiques.

Contenu de la matière :

TP Traitement avancé du signal

TP1 : Synthèse et application d'un filtre RIF passe-bas par la méthode des fenêtres (Hanning, Hamming, Bessel et/ou Blackman)
TP2 : Synthèse et application d'un filtre RII passe-bas par transformation bilinéaire
TP3 : Analyse spectrale paramétrique AR et/ou ARMA de signaux sonores (exemple de signaux non-stationnaires)
TP4 : Elimination d'une interférence 50Hz par l'algorithme du gradient LMS
TP5 : Débruitage d'un signal par la transformée en ondelette discrète DWT.

TP Systèmes asservis numériques

Mode d'évaluation:

Contrôle continu: 100%.

Références bibliographiques :

[1] .Mori Yvon, "Signaux aléatoires et processus stochastiques", Lavoisier, 2014.
[2] . 2. N. Hermann, "Probabilités de l'ingénieur : variables aléatoires et simulations Bouleau", 2002.
[3] . 3. M. Kunt, "Traitement Numérique des Signaux", Dunod, Paris, 1981.
[4] . 4. M. Bellanger, "Traitement numérique du signal : Théorie et pratique", 8e édition, Dunod, 2006

Semestre : 1
Unité d'enseignement : UEM 1.1
Matière 4 :Programmation C++ Embarquée
VHS : 37h30 (Cours : 1h30, TP : 1h00)
Crédits : 3
Coefficient : 2

Objectifs de l'enseignement :

C++ fait partie des langages incontournables, c'est l'un des langages les plus utilisés dans l'industrie. Ceci s'explique par la puissance du langage **objet**, sa polyvalence, et par ses performances d'exécution.
Cette matière a pour objectif de découvrir le langage avec un focus sur son utilisation spécifique sur les systèmes embarqués. L'étudiant pourra assimiler les concepts objets, découvrir et mettre en œuvre le langage C++ dans les projets et acquérir un début d'autonomie dans ce langage.

Connaissances préalables recommandées :

Systèmes numériques, programmation C.

Contenu de la matière:

Chapitre 1. Le langage C++ - introduction **(1 semaine)**
Historique, comparaison aux autres langages, utilisation de ce langage dans l'industrie, norme et version de C++ (C++98 • C++03 • C++11 • C++14 • C++17 • C++20), nouvelles fonctionnalités du langage

Chapitre 2. Compilateur **(1 semaine)**
- Compilateur open source GCC/G++
- Compilateur croisé x86/ARM
- Linkage dynamique et statique
- Débogage
- Makefile

Chapitre 3. Types, Constantes, Variables **(1 semaine)**
Références et Pointeurs, déclaration, portée, initialisation, tableau : déclaration, initialisation, espace de nommage, allocation dynamique

Chapitre 4. Notion d'objet **(2 semaines)**
- Du C vers C++, Classes et Objets
- Protection, Accès
- Variable d'instance, Constructeur, Destructeur
- Surcharge
- Opérateur "This"
- Objet et modélisation UML/SysML
- Génération de code automatique

Chapitre 5. Classes dérivées **(1 semaine)**
- Héritage et instanciation
- Amis
- Classe virtuelle
- Héritage multiple

Chapitre 6. Surcharge des Opérateurs **(1 semaine)**
- Opérateur Functions
- Surcharge
- Fonction amie, Classe amie

Chapitre 7. Contrôle de flux **(1 semaine)**
Entrées, Sorties, État, Surcharge, Gestion de fichier

Chapitre 8. Exceptions **(1 semaine)**
Définition d'une exception, Interception

Chapitre 9. Structure de données et STL **(1 semaine)**
Vecteur, Map, List, Pile, Algo standard
Chapitre 10. Particularité du C++ embarqué **(1 semaine)**
- Librairie std
- Compilation et linkage
- Règles de codage spécifique
- Classe virtuelle

Chapitre 11. Multithreading **(2 semaines)**
- Attribut thread_local
- Classe thread, Classe mutex
- Sémaphore vs Mutex
- Conditions, Verrous, future et promise
- Opérateur et opération atomique

Chapitre 12. Smart Pointers **(1 semaine)**
- Dépréciation de auto_ptr
- unique_ptr,
- shared_ptr
- weak_p

Chapitre 13. Principe sur les templates **(1 semaine)**
- Notion de programmation générique
- Classe de modèles
- Fonctions de modèles

Mode d'évaluation :

Examen: **60%**, Contrôle Continue : **40%**

Références bibliographiques:

[1] . Michael Barr, Anthony Massa, Programming embedded systems: in C and C++, 2ème edition, O'Reilly, 2006.
[2] . Igor Viarheichyk, Embedded Programming with Modern C++ Cookbook: Practical recipes to help you build robust and secure embedded applications on Linux, 1st Edition, Kindle Edition, ISBN-13: 978-1838821043
[3] . Arkady Miasnikov, C++ for embedded systems, Kindle Edition, 2015
[4] . Christopher Kormanyos, Real-time C++: Efficient Object-oriented and Template Microcontroller Programming, Springer-Verlag Berlin and Heidelberg GmbH & Co, 2015

Semestre: S1
Unité d'enseignement: UEM 1.1
Matière : Programmation avancée en Python
VHS: 45h00 (Cours 1h30, TP 1h30)
Crédits: 2
Coefficient: 2

Objectifs de la matière :

Compétences visées :

- Utilisation des outils informatiques pour l'acquisition, le traitement, la production et la diffusion de l'information
- Compétences en Python et gestion de projets,
- Compétences en automatisation et visualisation de données.

Objectifs :

- Approfondir la maîtrise du langage Python et initier les étudiants aux bases de l'analyse de données et de l'intelligence artificielle.
- Acquérir les bases de solides en informatique.
- Apprendre à programmer en Python, Excel
- Maitriser l'automatisation de tâches
- Maitriser un logiciel de gestion de projets

Matériels nécessaires :

- Un ordinateur avec Python installé,
- Bibliothèques Python : NumPy, Pandas, Scikit-learn, Matplotlib, os.listdir, os.path.exists, os.mkdir, os.rmdir, Matplotlib, Seaborn, Plitly , Request, Beautiful Soup, Tkinter, PyQT, …
- Tensorflow, PyTorch, …

Prérequis : Programmation Python,

Contenu de la matière :

Chapitre 1 : Rappels sur la programmation en Python (02 Semaines)

1. Introduction : Concepts de base en informatique et outils numériques, installation de Python.
2. Présentation de la notion de système d'exploitation : Roles, types (Linux, Woindows , ..) Gestions des priorités,
3. Présentations des réseaux informatiques (Principe, Adresse IP, DNS, internet, …)
4. Programmation de base : Mode interactif et mode script, Variables, types de données, opérateurs. Structures conditionnelles et boucles (if, for, while).
5. Fonctions et éléments essentiels : Fonctions prédéfinies et création de fonctions. Modules standards (math, random). Chaînes de caractères, listes, manipulation de base des données.
6. Les Fichiers , Listes Tuples, dictionnaires,
7. Exercices :
    - Exercices d'apprentissage de Python
    - Exercices d'utilisation des bibliothèques vus au cours (Math, Random, NumPy, Pandas,…)
    - ….

Chapitre 2 : Programmation et automatisation (04 semaines)

1. Principes d'Automatisation de tâches
    - Bibliothèques Python pour l'automatisation :
        - ✓ Pandas et NumPy.

- ✓ Os, shutil : manipulation de fichiers et dossiers
- ✓ Openpyxl ou pandas : travail avec des fichiers Excel ou CSV

- Définitions et exemples d'automatisation (envoi de mails,…)

2. Manipulation de fichiers avec Python :
   - Utiliser les librairies pour :
     - ✓ Parcourir un dossier (os.listdir)
     - ✓ Vérifier l'existence d'un fichier ou dossier (os.path.exists)
     - ✓ Créer ou supprimer des dossiers (os.mkdir, os.rmdir)
     - ✓ Visualiser des données : Matplotlib, Seaborn, Plitly
     - ✓ Request pour réagir avec des Interface de Programmation d'Application (API)
     - ✓ Beautiful Soup pour le Scraping de données
     - ✓ Tkinter, PyQT pour visualiser des données graphiques
   - Copier ou déplacer des fichiers avec shutil…
   - Recherche, tri et génération de rapports simples.
   - Sérialisation et Désérialisation (Utilisation du module pickle).
   - Sérialisation d'objets et traitement de fichiers volumineux (streaming).
   - ….

3. **Exercices** :
   - Utilisation de openpyxl et pandas pour lire, modifier et écrire des fichiers Excel ou CSV pour :
     - ✓ Créer des rapports automatiques
     - ✓ Extraire automatiquement des données
     - ✓ ….
   - Ecriture de scripts pour :
     - ✓ traiter des fichiers textes (recherche, tri)
     - ✓ automatiser des calculs techniques
     - ✓ gérer des rapports simples (PDF, Excel)
     - ✓ ….
   - Algorithmes de tri, de recherche et de tri par insertion
   - Implémenter une fonction de recherche dans une liste.
   - Opération sur les fichiers
   - Navigation sécurisée (configuration de réseaux simples, gestion des mots de passe)
   - ….

Chapitre 3 : Apprentissage avancé d'Excel (02 semaines)

1. Principes des macros et création d'une macro simple,
2. Tableaux croisés dynamiques,
3. Histogrammes,
4. Diagrammes en barres,
5. Araignée,
6. Etc.
7. **Exercices** Excel ….

Chapitre 4 : Apprentissage de GanttProject (02 semaines)

1. Introduction à la gestion de projets :

- Qu'est-ce qu'un projet ?
- Quels sont les enjeux de gestion d'un projet ?
- Interface de GanttProject

2. Les tâches (création, modification ,organisation)
3. Gestion du temps (dates de début ou de fin de projet)
4. Gestion des ressources
5. **Exercices** sur Gantt Project

Chapitre 4 : Programmation orientée objet avancée (03 semaines)

1. Organisation du code :
   - Fonctions personnalisées, paramètres, valeur de retour.
   - Modules, importations et packages.
2. Structures de données complexes :
   - Listes, tuples et dictionnaires : création, modification, suppression, parcours.
3. Concepts fondamentaux de la Programmation orientée objet (POO) :
   - Classes, objets, attributs et méthodes.
   - Attributs publics, privés et protégés.
4. Méthodes spéciales :
   - **init**, **str**, **repr**, **len**.
5. Concepts avancés :
   - Encapsulation, abstraction, héritage, polymorphisme.
   - Héritage avancé, décorateurs, design patterns, métaclasses.
6. **Exercices**

Chapitre 5 : Introduction aux données pour l'IA (02 semaines)

1. Introduction aux Datasets courants en IA :
   - Iris, MNIST, CIFAR-10, Boston Housing, ImageNet.
2. Prétraitement des données pour le Machine Learning:
   - Nettoyage, normalisation, encodage, séparation des données.
   - Validation croisée (cross-validation).
3. Techniques de Feature Engineering :
   - Sélection, création de caractéristiques, réduction de dimension.
4. Bibliothèques essentielles pour le développement des modèles IA:
   - scikit-learn, TensorFlow, Keras, PyTorch
5. **Exercices**

Travaux pratiques :

TP 01 : Maîtriser les bases de la programmation en Python

(Structures de contrôle, types, boucles, fonctions simples)

1. Initiation
2. Lire et traiter des fichiers textes
3. Gérer des rapports simples (PDF, Excel)

TP 02 :

- Elaborer un cahier de charges d'un mini projet d'automatisation de tâches avec Python consistant à identifier et à envoyer automatiquement des rapports par email avec Python :

1. Charger les données depuis un fichier (ex : mesures expérimentales),
2. Effectuer des statistiques simples sur les données (moyenne, écart-type avec interprétation),
3. Générer un graphique,
4. Envoi du résultat avec Python.

TP 03 :

1. Programmation ex Excel du tableau de bord vu en TD
2. Création de tableaux Excel automatisés
3. Macros simples,
4. Formules conditionnelles,
5. Recherche V.

TP 04 :

organiser une réunion en Ganttproject

1. Créer un nouveau projet :
    - Nom du projet : « Réunion …..
    - Date de début : Date et heure de la réunion
    - Durée estimée : durée totale de la réunion
2. Définition des tâches
    - Points à l'ordre du jour (chaque point de l'ordre du jour devient une tâche)
    - Sous-taches : Si un point est composé, créer alors les sous-tâches correspondantes
    - Tâches initiales et finales (par exemple : « Accueil de participants », « clôture de la réunion »)
3. Définition des ressources :
    - Participants (chaque participant est une ressource)
    - Matériel (ordinateur, datashow…)
4. Estimation des durées :
    - Durée de chaque point : temps nécessaire pour chaque point de l'ordre du jour
    - Temps de transition d'un point à l'autre
5. Création du diagramme de Gantt :
    - Visualiser l'ordre du jour
    - Identifier les points clés
6. Suivre l'avancement en temps réel (projection du Diagramme de Gantt)

TP 05 : Structures avancées et organisation du code

(Fonctions personnalisées, dictionnaires, modules et organisation modulaire

TP 06 : Programmation orientée objet avancée en Python

(Encapsulation, héritage, méthodes spéciales, design patterns simples)

TP 07 : Manipulation de fichiers et analyse de données

(Lecture/écriture de fichiers, traitement de texte, introduction à Pandas et NumPy)

TP 08 : Préparation et traitement de données pour l'intelligence artificielle

(Chargement de datasets IA, nettoyage, transformation, sélection de caractéristiques)

Projet final

Titre : Analyse et visualisation d'un jeu de données + modèle prédictif simple
Compétences mobilisées : Lecture de données, POO, structures avancées, Pandas, Scikit-learn. (Présentation orale + rapport écrit).

Mode d'évaluation :

examen 60% , CC=40%

Bibliographie

[1] . E.Schultz et M.Bussonnier (2020) : Python pour les SHS. Introduction à la programmation de données. Presses Universitaires de Rennes.

[2] . C.Paroissin, (2021) : Pratique de la data science avec R : arranger, visualiser, analyser et présenter des données. Paris : Ellipses, DL 2021.

[3] . S.Balech et C.Benavent : NLP texte minig V4.0, (Paris Dauphine – 12/2019) : lien : https://www.researchgate.net/publication/337744581_NLP_text_mining_V40_-_une_introduction_-_cours_programme_doctoral

[4] . Allen B. Downey Think Python: How to Think Like a Computer Scientist, O'Reilly Media, 2015;

[5] . Ramalho, L.. Fluent Python. " O'Reilly Media, Inc.", 2022;

[6] . Swinnen, G.. Apprendre à programmer avec Python 3. Editions Eyrolles, 2012;

[7] . Matthes, E. Python crash course: A hands-on, project-based introduction to programming. no starch press, 2019

[8] . Cyrille, H. (2018). Apprendre à programmer avec Python 3. Eyrolles, 6ème édition. ISBN: 978-2212675214

[9] . Daniel, I. (2024). Apprendre à coder en Python, J'ai lu

[10] . Nicolas, B. (2024). Python, du grand débutant à la programmation objet Cours et exercices corrigés, 3eme édition, Ellipses

[11] . Ludivine, C. (2024). Selenium Maîtrisez vos tests fonctionnels avec Python, Eni

Ressources en ligne :

- Documentation officielle Python : docs.python.org
- Exercices Python sur Codecademy : codecademy.com/learn/learn-python-3
- W3Schools Python Tutorial : w3schools.com/python/

Semestre : 1
Unité d'enseignement : UET 1.1
Matière : Respect des normes et des règles d'éthique et d'intégrité.
VHS : 22h30 (Cours : 1h30)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement:

Développer la sensibilisation des étudiants au respect des principes éthiques et des règles qui régissent la vie à l'université et dans le monde du travail. Les sensibiliser au respect et à la valorisation de la propriété intellectuelle. Leur expliquer les risques des maux moraux telle que la corruption et à la manière de les combattre, les alerter sur les enjeux éthiques que soulèvent les nouvelles technologies et le développement durable.

Connaissances préalables recommandées :

Ethique et déontologie (les fondements)

Contenu de la matière :

A.Respect des règles d'éthique et d'intégrité,

1. **Rappel sur la Charte de l'éthique et de la déontologie du MESRS :** Intégrité et honnêteté. Liberté académique. Respect mutuel. Exigence de vérité scientifique, Objectivité et esprit critique. Equité. Droits et obligations de l'étudiant, de l'enseignant, du personnel administratif et technique,
2. **Recherche intègre et responsable**

- Respect des principes de l'éthique dans l'enseignement et la recherche
- Responsabilités dans le travail d'équipe : Egalité professionnelle de traitement. Conduite contre les discriminations. La recherche de l'intérêt général. Conduites inappropriées dans le cadre du travail collectif
- Adopter une conduite responsable et combattre les dérives : Adopter une conduite responsable dans la recherche. Fraude scientifique. Conduite contre la fraude. Le plagiat (définition du plagiat, différentes formes de plagiat, procédures pour éviter le plagiat involontaire, détection du plagiat, sanctions contre les plagiaires, …). Falsification et fabrication de données.

3. **Ethique et déontologie dans le monde du travail :**

Confidentialité juridique en entreprise. Fidélité à l'entreprise. Responsabilité au sein de l'entreprise, Conflits d'intérêt. Intégrité (corruption dans le travail, ses formes, ses conséquences, modes de lutte et sanctions contre la corruption)

B- Propriété intellectuelle

I- Fondamentaux de la propriété intellectuelle

1- Propriété industrielle. Propriété littéraire et artistique.
2- Règles de citation des références (ouvrages, articles scientifiques, communications dans un congrès, thèses, mémoires, …)

II- Droit d'auteur

1. **Droit d'auteur dans l'environnement numérique**

Introduction. Droit d'auteur des bases de données, droit d'auteur des logiciels.Cas spécifique des logiciels libres.

2. **Droit d'auteur dans l'internet et le commerce électronique**

Droit des noms de domaine. Propriété intellectuelle sur internet. Droit du site de commerce électronique. Propriété intellectuelle et réseaux sociaux.

3. Brevet

Définition. Droits dans un brevet. Utilité d'un brevet. La brevetabilité. Demande de brevet en Algérie et dans le monde.

III- Protection et valorisation de la propriété intellectuelle

Comment protéger la propriété intellectuelle. Violation des droits et outil juridique. Valorisation de la propriété intellectuelle. Protection de la propriété intellectuelle en Algérie.

C. Ethique, développement durable et nouvelles technologies

Lien entre éthique et développement durable, économie d'énergie, bioéthique et nouvelle technologies (intelligence artificielle, progrès scientifique, Humanoïdes, Robots, drones,

Mode d'évaluation :

Examen : 100 %

Références bibliographiques:

[1] . Le site de l'Organisation Mondiale de la Propriété Intellectuelle www.wipo.int

[2] . Charte d'éthique et de déontologie universitaires, https://www.mesrs.dz/documents/12221/26200/Charte+fran__ais+d__f.pdf/50d6de61-aabd-4829-84b3-8302b790bdce

[3] . Arrêtés N°933 du 28 Juillet 2016 fixant les règles relatives à la prévention et la lutte contre le plagiat

[4] . L'abc du droit d'auteur, organisation des nations unies pour l'éducation, la science et la culture (UNESCO)

[5] . E. Prairat, De la déontologie enseignante. Paris, PUF, 2009.

[6] . Racine L., Legault G. A., Bégin, L., Éthique et ingénierie, Montréal, McGraw Hill, 1991.

[7] . Siroux, D., Déontologie : Dictionnaire d'éthique et de philosophie morale, Paris, Quadrige, 2004, p. 474-477.

[8] . Medina Y., La déontologie, ce qui va changer dans l'entreprise, éditions d'Organisation, 2003.

[9] . Didier Ch., Penser l'éthique des ingénieurs, Presses Universitaires de France, 2008.

[10] . Gavarini L. et Ottavi D., Éditorial. de l'éthique professionnelle en formation et en recherche, Recherche et formation, 52 | 2006, 5-11.

[11] . Caré C., Morale, éthique, déontologie. Administration et éducation, 2e trimestre 2002, n°94.

[12] . Jacquet-Francillon, François. Notion : déontologie professionnelle. Le télémaque, mai 2000, n° 17

[13] . Carr, D. Professionalism and Ethics in Teaching. New York, NY Routledge. 2000.

[14] . Galloux, J.C., Droit de la propriété industrielle. Dalloz 2003.

[15] . Wagret F. et J-M., Brevet d'invention, marques et propriété industrielle. PUF 2001

[16] . Dekermadec, Y., Innover grâce au brevet: une révolution avec internet. Insep 1999

[17] . AEUTBM. L'ingénieur au cœur de l'innovation. Université de technologie Belfort-Montbéliard

[18] . Fanny Rinck et léda Mansour, littératie à l'ère du numérique : le copier-coller chez les étudiants, Université grenoble 3 et Université paris-Ouest Nanterre la défense Nanterre, France

[19] . Didier DUGUEST IEMN, Citer ses sources, IAE Nantes 2008

[20] . Les logiciels de détection de similitudes : une solution au plagiat électronique? Rapport du Groupe de travail sur le plagiat électronique présenté au Sous-comité sur la pédagogie et les TIC de la CREPUQ

[21] . Emanuela Chiriac, Monique Filiatrault et André Régimbald, Guide de l'étudiant: l'intégrité intellectuelle plagiat, tricherie et fraude… les éviter et, surtout, comment bien citer ses sources, 2014.

[22] . Publication de l'université de Montréal, Stratégies de prévention du plagiat, Intégrité, fraude et plagiat, 2010.

[23] . Pierrick Malissard, La propriété intellectuelle : origine et évolution, 2010.

[24] . http://www.app.asso.fr/

[25] . http://ressources.univ-rennes2.fr/propriete-intellectuelle/cours-2-54.html

IV - Programme détaillé par matière du semestre S2

Semestre : 2
Unité d'enseignement : UEF 1.2.1
Matière 1 : Architecture des Processeurs pour l'Embarqué
VHS : 67h30 (Cours : 3h00, TD : 1h30)
Crédits : 6
Coefficient : 3

Objectifs de l'enseignement :

Cette matière permet aux étudiants d'apprendre en moins de temps comment exploiter l'architecture des processeurs ARM très utilisée dans les systèmes embarqués. Une première partie montrant comment configurer et travailler avec l'environnement de développement de la plate-forme STM32. Une seconde partie présente les bases de la programmation et les principaux aspects de l'officiel HAL (Hardware Abstraction Layer). Vers la fin, une partie plus avancée couvrant des aspects tels que l'utilisation d'un système d'exploitation en temps réel, des applications avancés comme l'IoT et l'USB.

L'application du contenu de cette matière s'effectue au niveau de la matière « **UEM1.2 : TP Architecture de microprocesseurs pour l'Embarqué** ».

Connaissances préalables recommandées :

Systèmes à microprocesseurs, Systèmes à microcontrôleurs, Programmation en C/C++ embarqué, Electronique numérique : FPGA & programmation VHDL.

Contenu de la matière:

Chapitre 1. Introduction aux processeurs ARM-Cortex (4 semaines)

- Les différents types de processeur ARM-Cortex
- Architecture des processeurs ARM Cortex
- Les registres
- La mémoire
- Le pipeline
- Les interruptions et les exceptions
- Le jeu d'instructions
- Les performances
- Introduction aux familles STM32.
- L'environnement de développement SM32CubeIDE.
- Aperçu sur la carte de développement Nucléo.

Chapitre 3. Exploitation des processeurs ARM-Cortex (8 semaines)

- Configuration des ports d'entrées/sorties
- Gestion des interruptions
- Communication UART
- Gestion de la DMA
- Gestion de l'horloge
- Utilisation des Timers
- Conversion analogique numérique
- Conversion numérique analogique
- Communication I²C
- Communication SPI

- Calcul des CRC (Cyclic Redundancy Check)
- Utilisation du Watchdog Timer
- L'horloge temps réel

Chapitre 4. Applications avancées des processeurs ARM-Cortex (3 semaines)

- FreeRTOS
- Développement IoT
- Développement USB

Mode d'évaluation:

Examen : **60%**, Contrôle Continue : **40%**

Références bibliographiques:

[1] . Carmine Noviello, « Mastering STM32 : A step-by-step guide to the most complete ARM Cortex-M platform, using the official STM32Cube development environment », 2ème édition, Lean Pub, 2022.

[2] . Donald Norris, « Programming With STM32: Getting Started With the Nucleo Board and C/C++ », McGraw-Hill, 2018.

[3] . Yifeng Zhu, « Embedded systems with ARM Cortex-M Microcontrollers in assembly language and C », 3ème édition, E-Man Press LLC, 2017.

[4] . Joseph Yiu, « Definitive guide to ARM Cortex-M23 and Cortex-M33 processors », Elsevier, 2021.

[5] . Joseph Yiu, « The definitive guide to ARM Cortex-M0 and Cortex-M0+ Processors », 2ème édition, Elsevier, 2015.

[6] . Joseph Yiu, « The definitive guide to ARM Cortex-M3 and Cortex-M4 Processors », 3ème édition, Elsevier, 2014.

[7] . Joseph Yiu, « The definitive guide to ARM Cortex-M0 », Elsevier, 2011.

[8] . Joseph Yiu, « The definitive guide to ARM Cortex-M3 », 2ème édition, Elsevier, 2010.

Semestre: 2
Unité d'enseignement: UEF 1.2.1
Matière 2: Processeurs numériques du signal (DSP)
VHS: 45h00 (Cours: 1h30, TD: 1h30)
Crédits: 4
Coefficient: 2

Objectifs de l'enseignement:

Connaître l'architecture interne d'un DSP et la plateforme matérielle intégrant ce DSP ainsi que l'environnement de développement sur une plateforme à base de DSP. A l'issue de cette matière, l'étudiant doit maîtriser le flot de conception et doit être également en mesure de faire une adéquation algorithme-architecture pour l'implémentation d'algorithmes sur une plateforme à base de processeurs DSP.

Connaissances préalables recommandées :

Systèmes à microprocesseurs. Traitement numérique avancé du signal. Programmation en langage assembleur et C.

Contenu de la matière:

Chapitre 1 : Arithmétique à virgule fixe et à virgule flottante (2 semaines)
Généralités sur les processeurs DSP, Rappels sur la numérisation des signaux (échantillonnage, quantification), formats de représentations des nombres, codage des nombres entiers (entiers positifs ou non signés, complément à 1, complément à 2), représentation des nombres réels dans un calculateur (virgule fixe, virgule flottante)

Chapitre 2 : Architecture des DSP TMS320C6x (4 semaines)
Architecture interne du C6000, le processeur, cartographie mémoire, unités fonctionnelles, paquets *fetch* et exécution, architecture pipeline, les registres, les registres de contrôle, les périphériques (timers, PLL, interruptions, HPI, GPIO), la liaison série (*multichannel buffered serial port*), présentation du jeu d'instructions

Chapitre 3 : Algorithmes de traitement du signal sur DSP (5 semaines)
Adéquation algorithme-architecture. Problèmes de quantification, contraintes temps-réel, gestion des entrées/sorties. Présentation des modes d'adressage (indirect, circulaire, inversé). Implémentation Filtrage RIF et RII. Buffers à décalage et circulaire, Implémentation de la FFT sur DSP (Adressage inversé).

Chapitre 4 : Techniques avancées de gestion mémoire DSP (4 semaines)
Présentation des mémoires internes (niveaux L1 et L2) et des mémoires externes (SRAM, Flash, DDRAM, ...). Gestion des sections mémoires via le fichier *.cmd (organisation des sections). Gestion de la mémoire externe par L'EMIF (*External Memory InterFace*). Technique de transfert par blocs. Organisation des données pour l'EDMA. Paramètres et options pour l'EDMA. Exemple de transfert de données en utilisant le contrôleur DMA.

Mode d'évaluation:

Contrôle continu: 40% ; Examen: 60%.

Références bibliographiques:

[1] . R. Chassaing, D. Reay, Digital Signal Processing and Applications with the TMS320C6713 and TMS320C6416 DSK, John Wiley & Sons, 2008.

[2] . D. Reay, Digital Signal Processing and Applications with the OMAP-L138 eXperimenter, John Wiley & Sons, 2012.

[3] . T.B. Welch, C.H.G. Wright and M.G. Morrow, Real-Time Digital Signal Processing from MATLAB to C with TMS320C6x DSPs, CRC Press, 2012.

[4] . Steven A Tretter, Communication System Design Using DSP Algorithms, Springer 2008.

[5] . N. Dahnoun, Digital Signal Processing Implementation using the TMS320 C6000 DSP platform, Prentice Hall, 2000.

[6] . N. Kehtarnaz, N. Kim, Real Time Digital Signal Processing Based on TMS320C6000, Newnes, 2004.

[7] . N. Kehtarnaz, M. Keramat, DSP System Design using TMS320C6000, Prentice Hall, 2006.

[8] . S. W. Smith, The Scientist and Engineer's Guide to Digital Signal Processing.

[9] . G. Baudoin et F. Virolleau, Les DSP : famille TMS320C54x. Développement d'applications.

[10] . L. Correvon, DSP et Temps réel : Application Industrielle, Haute Ecole d'Ingénierie du Canton de Neuchâtel.

[11] . P. Laspsley , J. Bier , A. Shoham, E. A. Lee, DSP Fundamentals: Architecture and Features, Berkley Design Technology, Inc, 1994.

[12] . Oktay Alkin, Digital Signal Processing: A Laboratory Approach using. PC-DSP, Prentice Hall.

[13] . Digital Control Applications with the TMS320 Family: Selected Application notes, Texas Instruments, 1991.

[14] . M. Pinard, Les DSP, famille ADSP 218X: Principes et applications, Dunod, 2000.

[15] . B. Bouchez, Applications audio-numériques des DSP: théorie et pratique du traitement numérique du son, Publitronic, 2003.

[16] . Texas Instruments, TMS320C6000 Code Composer Studio Tutorial (Rev. C), http://www.ti.com/lit/ug/spru301c/spru301c.pdf, 2000.

[17] . Texas Instruments, Code Composer Studio Development Tools v3.3 Getting Started Guide (Rev. H), http://www.ti.com/lit/ug/spru509h/spru509h.pdf, 2008.

[18] . Texas Instruments, TMS320C6000 Programmer's Guide (Rev. K), http://www.ti.com/lit/ug/spru198k/spru198k.pdf, 2011.

[19] . Texas Instruments, TMS320C6000 CPU and Instruction Set Reference Guide (Rev. G), http://www.ti.com/lit/ug/spru189g/spru189g.pdf, 2006.

[20] . Texas Instruments, TMS320C6000 Chip Support Library API Reference Guide (Rev. J), http://www.ti.com/lit/ug/spru401j/spru401j.pdf, 2004.

[21] . Texas Instruments, TMS320C1X User's Guide. Juillet 1991.

Semestre: 2
Unité d'enseignement: UEF 1.2.2
Matière : Système de vision artificielle
VHS: 45 h (Cours: 01h30 , TP: 01h30)
Crédits:4
Coefficient:2

Objectifs de l'enseignement :

Connaitre les composantes principales d'un système de vision artificielle ainsi que le rôle de chaque composante dans la conception de ce dernier. Etudier les outils permettant la reproduction automatique des taches effectuées par le système visuel humain et interprétées par le cerveau.

Connaissances préalables recommandées:

- ✓ Traitement du signal ;
- ✓ Traitement d'image.

Contenu de la matière :

Chapitre 1. Acquisition et numérisation de l'image (02 semaines)
Composition fonctionnelle d'un système de vision artificielle, Définition de l'image, Capteurs d'images, numérisation de l'image**,** Vision humaine**,** Modélisation 3D et calibration de la caméra.

Chapitre 2. Rappels sur le traitement d'image (02 semaines)
Opérations ponctuelles (Transformation logarithmique, Inversion du contraste, modification de l'histogramme), Opérations locales (Filtrage spatial, filtrage fréquentiel)

Chapitre 3. Contours et segmentation (03semaines)
Détection de contours (Définition d'un contour, approche gradient, approche lapalucien, Filtrage de Canny, Filtrage LOG, Contours actifs) Segmentation (Seuillage d'histogramme, Approches régions)

Chapitre 4. Mouvement (02semaines)
Estimation de mouvement, flux optique, méthode de Horn & Schunk, Mise en correspondance par bloc Approche Gunnar Farneback

Chapitre 5. Détection de caractéristiques (02 semaines)
Définition d'un point d'intérêt, transformations géométriques, Détecteur de Moravec Détecteur de Harris ,SIFT

Chapitre 6. Apprentissage machine (03semaines)
Définition de l'apprentissage machine, classification supervisée (K-PPV, Naive bayes , SVM), classification non supervisée (K-Means, Fuzzy C-Means)

Contenu de la matière TP:

TP1 : introduction à l'utilisation de OpenCV

- Représentation des images et vidéo sous OpenCV
- Traitements sur la Couleur et palette

TP2 : Calibration de camera et reconstruction 3D

TP3 : Détection de contours et segmentation

TP4 : Détection et estimation de mouvement

TP5 : Détection de caractéristiques

TP6 : Poursuite d'objets

TP7: Machine Learning

Mode d'évaluation:

Contrôle continu: 100%

Références bibliographiques

Mode d'évaluation:

Contrôle continu TP : 40 % ; Examen : 60 %.

Références bibliographiques :

[1] .R.Gonzalez, Digital Image Processing, *Upper Saddle River, N.J, Prentice Hall, 2002 (ISBN 978-0-201-18075-6).*

[2] . M.Bergounioux, Introduction au traitement mathématique des images : méthodes déterministes*, vol. 76, Berlin, Heidelberg, Springer Berlin Heidelberg, coll. « Mathématiques et Applications », 19 mars 2015*

[3] . A. Herbulot, Mesures statistiques non-paramétriques pour la segmentation d'images et de vidéos et minimisation par contours actifs, Thèse de Doctorat soutenue à Université de Nice - Sophia Antipolis, le 10 octobre 2007

[4] . C.Grava , Compensation de mouvement par réseaux neuronaux cellulaires : Application en imagerie médicale, Thése de doctorat, soutenue à l'INSA de Lyon 2003.

[5] . J-P. Cocquerez et S. Phillip, Préface H.Maitre Analyse d'images : Filtrage et segmentation, Edition Masson , 1995

[6] . H. P. Moravec, "Towards Automatic Visual Obstacle Avoidance". In Proceeding of Fifth International Joint Conference on Artificial Intelligence, Cambridge Massachusetts USA, pp. 584-587, 1977.

[7] . C. Harris and M. Stephens. A combined corner and edge detector. In Proceedings of The Fourth Alvey Vision Conference, pages 147

[8] . D.Lowe, Object recogniton from local scale invariant features. In Proceedings of The IEEE international conference on computer vision, 1999

[9] . I. Rish, "An empirical study of the naive Bayes classifier". IJCAI Workshop on Empirical Methods in Artificial Intelligence

[10] . V.Vapnik, « Support Vector Regression Machines ». Advances in Neural Information Processing Systems 9, NIPS 1996, 155-161, MIT Press. 1997

[11] . E. Lebarbier, T. Mary-Huard, Notes de cours, Classification non supervisée, AgroParisTech, 2017

Semestre : 2
Unité d'enseignement : UEF 1.2.2
Matière4 : Automates Programmables Industriels
VHS : 45 h 00 (Cours : 1h 30, TD : 1h 30)
Crédits : 4
Coefficient : 2

Objectifs de l'enseignement :

Ce cours permet à l'étudiant de comprendre l'organisation matérielle et logicielle des API, de choisir un API et les composants associés selon l'application souhaitée et d'utiliser un langage de programmation adapté pour l'API.

Connaissances préalables recommandées :

Logique combinatoire et séquentielle, Microprocesseurs, Microcontrôleurs, Capteurs, Réseaux et communications industriels.

Contenu de la matière :

Chapitre 1 : API : Automates Programmables Industriels (2 semaines)
Définition d'un API, Architecture interne et externe d'un API et caractéristiques. Choix de l'API. Types des Entrées/Sorties d'un API et ses caractéristiques.
Chapitre 2 : Matérialisation de processus industriels par les API (3 semaines)
Définition d'un système automatisé. Les parties essentielles d'un système automatisé (PO, PC, HMI, Interfaçage). Principe de fonctionnement d'un API et d'un système automatisé ordres-informations. Câblage. Notions capteur-actionneur, réseaux industriels, ...
Chapitre 3 : Programmation de l'API (5 semaines)
Introduction au Grafcet. Introduction aux langages : LD, IL, FBD, SFC, SCL. Application : définition des parties PO-PC, élaboration du grafcet, programmation en Ladder. Exercices d'application.
Chapitre 4 : Visualisation de processus (3 semaines)
Introduction aux systèmes IHM (Interface Human Machine) et SCADA, représentation et commande de processus, affichage d'alarmes, recettes, archivage, gestion des utilisateurs, …. Exercices d'application.
Chapitre 5 : Automate programmable dédié à la sécurité (2 semaines)
Architecture, commande de processus et de machine, gestion des fonctions de sécurité.

Mode d'évaluation :

Contrôle continu : 40 % ; Examen : 60 %.

Références bibliographiques :

[1] . Frank D. Petruzella, Programmable Logic Controllers, 4th edition, Ed. Mc Graw Hill 2004.
[2] . William Bolton, Les automates programmables industriels, Editions Dunod, l'Usine Nouvelle, 2010.
[3] . Ian G. Warnock, Programmable Controllers: Operation and Application, Prentice Hall.
[4] . Gilles Michel, Architecture et applications des automates programmables industriels, Dunod.
[5] . G. Michel, Automates Programmables industriels, Dunod, 1979.
[6] . S. Thelliez et J.M.Toullote, Grafcet et logique industrielle programmée, Eyrolles, 1980.
[7] . J.C Bossy, P. Brard, P. Faugère, C. Merlaud, Le Grafce : sa pratique et ses applications, Educalivre Ed. Casteilla, 1995.
[8] . Henri Ney, Eléments d'automatismes, Collection Electrotechnique et normalisation, Edition Nathan, 1996.

[9] . M. Diaz, Les Réseaux de Pétri - Modèles fondamentaux. Traité IC2 - Série Informatique et Systèmes d'Information, Hermès Science 2001

[10] . A. Choquet-Geniet, Les réseaux de Pétri – Un outil de modélisation, Dunod, 2006. P a g e | 50

[11] . P. Ladet, Outils de modélisation des automatismes séquentiels, Les réseaux de Pétri, Techniques de l'ingénieur, 1990.

[12] . Support d'apprentissage/ de formation siemens, Module 041-101 TIA Portal WinCC Basic avec KTP700 Basic et SIMATIC S7-1200.

[13] . CEI 61508-2 Sécurité fonctionnelle des systèmes électriques/ électroniques/électroniques programmables relatifs à la sécurité.

Semestre : 2
Unité d'enseignement : UEM 1.2
Matière 1 : TP Architecture des Processeurs pour l'Embarqué/ TP Processeurs Numériques du signal (DSP)
VHS : 30h00 (TP : 2h00)
Crédits : 3
Coefficient : 1

TP Architecture des Processeurs pour l'Embarqué

Objectifs de l'enseignement :

Cette matière est construite autour de la matière « **UEF1.2.2 : Architecture de Microprocesseurs pour l'Embarqué** » ; elle permet aux étudiants d'appliquer les connaissances acquises sous forme d'exemples précis.

Connaissances préalables recommandées :

Systèmes à microprocesseurs, logique combinatoire et séquentielle, électronique fondamentale, électronique de puissance.

Contenu de la matière:

TP 1 : Introduction à la carte de développement STM Nucleo
TP 2 : Prise en main de l'environnement SM32CubeIDE.
TP 3 : Implémentation d'un projet simple (clignotement d'une LED) sur la carte STM Nucleo
TP 4 : Développement d'un projet avec plusieurs entrées/sorties : poussoirs, LED, …
TP 5 : Développement d'un projet avec des interruptions
TP 6 : Configuration et utilisation des périphériques Timers.
TP 7 : Transmission et réception avec une communication série USART
TP 8 : Conversion analogique numérique
TP 9 : Variation de la largeur d'impulsion PWM
TP 10 : Démonstration du fonctionnement DMA
TP 11 : Conversion numérique analogique

Mode d'évaluation:
Contrôle Continue : **100%**

Références bibliographiques:

[1] . Carmine Noviello, « Mastering STM32 : A step-by-step guide to the most complete ARM Cortex-M platform, using the official STM32Cube development environment », 2ème édition, Lean Pub, 2022.
[2] . Donald Norris, « Programming With STM32: Getting Started With the Nucleo Board and C/C++ », McGraw-Hill, 2018.
[3] . Yifeng Zhu, « Embedded systems with ARM Cortex-M Microcontrollers in assembly language and C », 3ème édition, E-Man Press LLC, 2017.
[4] . Joseph Yiu, « Definitive guide to ARM Cortex-M23 and Cortex-M33 processors », Elsevier, 2021.
[5] . Joseph Yiu, « The definitive guide to ARM Cortex-M0 and Cortex-M0+ Processors », 2ème édition, Elsevier, 2015.
[6] . Joseph Yiu, « The definitive guide to ARM Cortex-M3 and Cortex-M4 Processors », 3ème édition, Elsevier, 2014.
[7] . Joseph Yiu, « The definitive guide to ARM Cortex-M0 », Elsevier, 2011.

[8] . Joseph Yiu, « The definitive guide to ARM Cortex-M3 », 2ème édition, Elsevier, 2010.

TP Processeurs Numériques du Signal (DSP)

Objectifs de l'enseignement:

Mettre en pratique les connaissances théoriques acquises durant les séances de cours et de TD. Familiariser l'étudiant aux programmes d'applications sur une plateforme DSP moyennant une adéquation algorithme architecture. Les programmes peuvent être aussi exécutés à l'aide du simulateur fourni avec CCS.

Connaissances préalables recommandées:

Systèmes à microprocesseurs. Traitement avancé du signal. Programmation en langage assembleur et C.

Contenu de la matière:

Les équipes de formation sont priées de réaliser au moins 4 TPs (voire plus, si cela est possible) en fonction du type de la plateforme DSP disponible. Par ailleurs, il est permis de rajouter ou remplacer quelques TPs de la liste jointe par d'autres TPs en relation avec la matière.

***Précision** : Tout changement apporté à cette liste doit être signalé au CPND de manière à en faire profiter les autres établissements.*

TP1: Prise en main de CCS et découverte de la carte d'évaluation DSKTMS320C6x
Environnement de développement intégré "Code Composer Studio (CCS)", compilation, chargement, exécution et débogage de programmes simples.
TP2: Acquisition, traitement et restitution de signaux audio avec la DSKTMS320C6x
Echantillonnage, repliement de spectre, quantification, transfert de données du/au Codec et utilisation en mode scrutation ou en mode interruption.
TP3: Génération de signaux avec la DSKTMS320C6x
Génération d'ondes, onde sinusoïdale, modulation AM et modulation FM.
TP4: Implémentation de filtres numériques par la DSKTMS320C6x
Filtres IIR et RIF.
TP 5Implémentation de la Transformée de Fourier discrète et rapide
TFD et TFR
TP6: Utilisation de Matlab avec la DSK TMS320C6x
Simulations : Matlab ou Simulink, génération automatique du code pour la DSK en utilisant Simulink, Real Time Workshop et Code Composer Studio.

Mode d'évaluation:

Contrôle continu: 100%

Références bibliographiques:

[1] . R. Chassaing, D. Reay, Digital Signal Processing and Applications with the TMS320C6713 and TMS320C6416 DSK, John Wiley & Sons, 2008.
[2] . T.B. Welch, C.H.G. Wright and M.G. Morrow, Real-Time Digital Signal Processing from MATLAB to C with TMS320C6x DSPs, CRC Press, 2012.
[3] . Steven A Tretter, Communication System Design Using DSP Algorithms, Springer 2008.

Semestre: 02
Unité d'enseignement: UEM 1.2
Matière :TP Automates Programmables Industriels
VHS: 15h00 (TP: 1h00) – Une fois par quinzaine
Crédits: 2
Coefficient: 1

Objectifs de l'enseignement:

Apprendre à l'étudiant comment installer, programmer et utiliser un API. Lui montrer commentanalyser et respecter les contraintes technologiques et sécuritaires liées à l'interfaçage des différentséléments industriels avec un API. L'initier à maitriser les tâches d'édition et de débogage desprogrammes ainsi que la correction des erreurs détectées.

Connaissances préalables recommandées:

- ✓ Microcontrôleurs ;
- ✓ API ;
- ✓ Grafcet ;
- ✓ Ladder.

Contenu de la matière: Choisir au moins 5 TPs.

TP01 : Prise en main de l'environnement API : Simulation d'un système automatisé, Revue des différents logiciels. Introduction au logiciel Step7 de Siemens
TP02 : Mise en oeuvre d'un API : Configuration Hardware. Initiation à la programmation en Ladder (Marche-Arrêt d'un actionneur avec maintien). Utilisation des entrés/sorties TOR : Utilisation des relais, contacteur, … (faire éventuellement les câblages nécessaires).
TP03 : Contrôle du niveau d'un réservoir. Utilisation des blocs spéciaux (interruptions)
Exemples d'applications : Réaliser au moins 2 TPs parmi la liste des TPs suivantes
TP04 : Contrôle d'un vérin pneumatique
TP05 : Contrôle de feux de signalisation tricolores pour une simple intersection
TP06 : Contrôle d'une unité de remplissage et de transfert de bouteilles
TP07 : Contrôle d'une perceuse automatisé
TP08 : Transfert et tri de pièces de différentes dimensions
TP09 : Contrôle d'une unité de matriçage
TP10 : Contrôle d'une unité de Fabrication de Médicaments
TP11 : Contrôle d'un Four Tunnel

Mode d'évaluation:

Contrôle continu : 100%.

Références bibliographiques:

[1] . Frank D. Petruzella, Programmable Logic Controllers, 4th edition, Ed. Mc Graw Hill 2004.
[2] . William Bolton, Les automates programmables industriels, Editions Dunod, l'Usine Nouvelle, 2010.
[3] . Ian G. Warnock, Programmable Controllers: Operation and Application, Prentice Hall.
[4] . Gilles Michel, Architecture et applications des automates programmables industriels, Dunod.
[5] . G. Michel, Automates Programmables industriels, Dunod, 1979.
[6] . S. Thelliez et J.M.Toullote, Grafcet et logique industrielle programmée, Eyrolles, 1980.
[7] . J.C Bossy, P. Brard, P. Faugère, C. Merlaud, Le Grafce : sa pratique et ses applications, Educalivre Ed. Casteilla, 1995.

Semestre: 3
Unité d'enseignement: UEM 1.2
Matière : Programmation Java pour les systèmes embarqués
VHS: 45h00 (Cours: 1h30 , TP: 1 h00)–Choisir entre Python ou Java
Crédits: 3
Coefficient: 2

Objectifs de l'enseignement:

Selon l'IEEE Python et Java sont les Meilleurs langages de programmation en 2021. Python est un langage de programmation très répandu et facile à apprendre. Il est adapté aussi bien aux débutants qu'aux experts pour sa simplicité, sa syntaxe lisible et sa variété d'usage. Il s'impose dans tous les domaines: l'embarqué, un développeur web, un analyste, un data scientist ou un expert marketing,..etc. Java est un langage objet pur, c'est le langage le plus approprié à la programmation du réel. Il est présent dans les noyaux logiciels de presque tous les appareils technologiques actuels. Cette matière permet à l'étudiant d'accéder à un niveau appréciable en programmation Java qui lui permet de faire face à cet aspect de la technologie moderne.

Connaissances préalables recommandées:

- ✓ Programmation (langage Pasacal/Matlab,/Langage C) ;
- ✓ Infomatique 1, Informatique 2, Informatique 3 ;

Contenu de la matière:

Java :

Chapitre 1:Introduction à Java (1 semaine)
Chapitre 2: Les instructions de contrôle (3 semaines)
A) Les instructions de choix : L'instruction *if* (Le *if* simple, *Le* if *avec partie* else, *ifs* imbriqués et *if-else if*), 2.4 Opérateurs logiques, L'instruction *switch,* Opérateurs conditionnels, 2.7 Priorité et associativité des opérateurs
B)Les boucles
2.9 La boucle *while*
2.10 La boucle *do...while*
2.11 La boucle *for*
2.12 Boucles imbriquées
2.13 Les mots-clés *break* et *continue*
L'instruction *break*
L'instruction *break* avec étiquette
L'instruction *continue*
L'instruction *continue* avec étiquette
Chapitre 3: Fonction mathématiques, Caractères et Chaînes de caractères (3 semaines)
3.2 Fonctions mathématiques couramment utilisées
3.2.1 Méthodes trigonométriques
3.2.2 Méthodes des exposants (les Exponentielles et les puissances)
3.2.3 Les méthodes d'arrondis
3.2.4 Les méthodes min, max et abs
3.2.5 La méthode random
3.3 Type de données de caractères et opérations
3.4 Le type String
3.4.1 Obtention de la longueur d'une chaîne
3.4.2 Obtention de caractères d'une chaîne
3.4.3 Concaténation des chaînes de caractères
3.4.4 Conversion des chaînes de caractères
3.4.5 Lecture d'une chaîne depuis le clavier

3.4.6 Lecture d'un caractère depuis le clavier
3.5 Sortie formatée sur écran (instruction System.out.printf)
Chapitre 4: Les méthodes (2 semaines)
4.2 Définition d'une méthode
4.3 Appel d'une méthode
4.4 Méthodes *void* et méthodes permettant un retour de valeur
4.5 Passage des paramètres par valeurs
4.6 Portée des variables
4.7 La surcharge de méthode
Chapitre 5: Les tableaux (3 semaines)
A) Tableaux unidimensionnels
5.2 Notions de base sur les tableaux
5.2.1 Déclaration des tableaux
5.2.2 Création de tableaux
5.2.3 Taille du tableau et valeurs par défaut
5.2.4 Accès aux éléments du tableau
5.2.5 Initialisateurs de tableaux
5.2.6 Boucle foreach
5.3 Transmettre des tableaux aux méthodes
5.4 Retourner un tableau à partir d'une méthode
5.5 Listes d'arguments à longueur variable
B) Tableaux bidimensionnels
5.7 Notions de base sur les tableaux bidimensionnels
5.7.1 Déclaration et création de tableaux bidimensionnels
5.7.2 Obtention des longueurs des tableaux bidimensionnels
5.7.3 Tableaux déchiquetés (dentelés)
C) Tableaux multidimensionnels
Chapitre 6: Objets et Classes (3 semaines)
6.2 Définition des classes pour les objets
6.3 Exemple : Définition des classes et création des objets
6.4 Construction d'objets à l'aide de constructeurs
6.5 Accès aux objets via les variables de référence
6.5.1 Variables de référence et types de référence
6.5.2 Accès aux données et aux méthodes d'un objet
6.6 Variables statiques, constantes et méthodes statiques

Mode d'évaluation:

Contrôle continu: ...40 % ; Examen: 60 %.

Références bibliographiques:

[1] . Allen B. Downey Think Python: How to Think Like a Computer Scientist, O'Reilly Media, 2015;
[2] . Zed A. Shaw Learn Python 3 the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code, Addison-Wesley Professional, 2017;
[3] . Barry, P. Head first Python: A brain-friendly guide. " O'Reilly Media, Inc.", 2016;
[4] . Ramalho, L.. Fluent Python. " O'Reilly Media, Inc.", 2022;
[5] . Swinnen, G.. Apprendre à programmer avec Python 3. Editions Eyrolles, 2012;
[6] . Le Goff, V.. Apprenez à programmer en Python. Editions Eyrolles, 2019;
[7] . Matthes, E. Python crash course: A hands-on, project-based introduction to programming. no starch press, 2019;
[8] . Harvey Deitel, Java: How to Program, 9th Edition, Prentice Hall.;
[9] . Robert Sedgewick and Kevin Wayne, Introduction to Programming in Java: An Interdisciplinary Approach, Addison Wesley, 2007;
[10] . Claude Delannoy, Programmer en Java, Editions Eyrolles ;
[11] . J. Hunter, Java servlets, O'Reilly ;
[12] . P. Niemeyer, J. Knudsen, Introduction à Java, Ed. O'Reilly

Semestre: S2
Unité d'enseignement: UET 1.2.1
Matière : Eléments d'intelligence artificielle appliquée
VHS: 45h00 (Cours 1h30, TP 1h30)
Crédits: 2
Coefficient: 2

Compétences visées :

- Identifier les opportunités de l'intelligence artificielle en sciences de l'ingénieur
- Comprendre les implications éthiques de l'IA et les bonnes pratiques de son utilisation.
- Capacité à utiliser les techniques de l'IA dans la résolution de problèmes

Objectifs :

- Maitrise des algorithmes IA
- Initiation aux concepts, outils et applications fondamentales de l'intelligence artificielle moderne, en mettant l'accent sur la pratique avec Python et ses bibliothèques.
- Approfondir le langage Python,
- Comprendre les approches de l'IA dans la résolution de problèmes,

Prérequis :

Programmation avancée Python

Matériels nécessaires :

- Un ordinateur avec Python installé,
- Bibliothèques Python : NumPy, Pandas, Scikit-learn, Matplotlib, os.listdir, os.path.exists, os.mkdir, os.rmdir, Matplotlib, Seaborn, Plitly , Request, Beautiful Soup, Tkinter, PyQT, …
- Tensorflow, PyTorch, …

Contenu de la matière :

Chapitre 1 : Introduction à l'intelligence artificielle l'IA (01 semaine)

1. Définitions et champs d'application de l'IA.
2. Évolution historique de l'IA.
3. Introduction aux grands domaines :
    - Apprentissage automatqiue (Machine Learning)
    - Apprentissage profound (Deep Learning)

Chapitre 2 : Mathématiques de base pour l'IA (01 semaine)

1. **Algèbre linéaire** : vecteurs, matrices, produits, normes.
2. **Probabilités & statistiques** :
    - Variables, espérance, variance.
    - Lois usuelles : normale, binomiale, uniforme.
3. **Régression linéaire simple** :
    - Formulation, coût, optimisation.
    - Mise en œuvre avec **Scikit-learn**.
4. **Exercices :**
    - Manipulation de matrices avec la bibliothèque NumPy (Python)
    - Exercice sur la régression linéaire (utiliser une bibliothèque Python comme Scikit-learn par exemple)

- Expliquer la bibliothèque Matplotlib (Python)
- …

Chapitre 3 : Apprentissage automatqiue (Machine Learning) (03 semaines)

1. Concepts clés : Données, Modèles, features, étiquettes, généralisation.
2. Phases d'un pipeline d'apprentissage : entraînement, validation, test.
3. Types d'apprentissage :
   - Supervisé
   - Non supervisé
   - Par renforcement *(aperçu)*
4. **Exercices :**
   - Approfondir les notions vues au cours
   - ….

Chapitre 4 : Classification supervisée (3 semaines)

1. Principe d'entrainement de modèle de classification simple :
2. Les modèles et algorithmes :
   - SVM (Support Vector Machine)
   - Arbres de décisions
3. Évaluation de performance :
   - Matrice de confusion, précision, rappel, F1-score.
5. **Exercices :**
   - Expliquer comment utiliser Scikit-learn ?
   - Comparaison de plusieurs modèles sur un dataset
   - ….

Chapitre 5 : Apprentissage non supervisé

1. Notion de clustering.
2. Algorithmes :
   - **K-means**
   - DBSCAN (Density-Based Spatial Clustering of Applications with Noise)
3. Visualisation 2D et interprétation des résultats.
4. **Exercices :**
   - Expliquer comment utiliser un algorithme de clustering sur un Dataset
   - Expliquer comment visualiser les clusters.
   - ….

Chapitre 6 : Les réseaux de neurones

1. Architecture d'un réseau de neurones :
   - Perception,
   - Couches et couches caches, poids, biais.
   - Fonction d'activation : ReLU, Sigmoïde, Softmax, ….
   - Exercices d'applications
2. Introduction au **Deep Learning** :
   - Notion de couches profondes.

- Introduction au réseaux convolutifs (CNN)

3. **Exercices :**
   - Expliquer Tensorflow et PyTorch
   - Analyser un Dataset de texte et prédire des sentiments
   - ….

Chapitre 6 : Introdution Les réseaux de neurones

Chapitre 7 : Mini projet (travail personnel encadré en dehors des cours) :

Création d'un modèle complet de classification ou clustering, avec prétraitement, entraînement et visualisation ; choisir et traiter un projet du début jusque la fin parmi (à distribuer au début du semestre) :

- Reconnaissance des caractères manuscrits
- Prédiction des catastrophes naturelles
- Développer un Chatbot capable de répondre aux questions fréquentes d'une entreprise, de manière naturelle.
- Développer un système capable de distinguer les sons normaux d'une machine de ceux indiquant une anomalie (roulement défectueux, vibration excessive, etc.)
- Développer un système (mini IA) capable d'analyser les sentiments exprimés dans les publications sur réseaux sociaux à propos d'un produit, une marque ou un évènement.
- …

Travaux pratiques :

TP 01 : Initialisation

TP 02 :

- Implanter une régression simple avec Scikit-learn visualisation avec Matplotlib (par exemple)
- Visualiser les résultats avec Matplotlib
- …

TP 03 :

- **Pipeline de machine learning et séparation des données**
- Approfondir es notions vues au cours

TP 04 :

- Utilisation Scikit-learn pour entrainer un modèle de classification simple
- …….

TP 05 :

- Implanter un algorithme de clustering sur un Dataset
- Visualiser les clusters : Clustering non supervisé (K-means, DBSCAN)**.**
- ….

TP 06 :

- Construire un réseau de neurones simple avec TensorFlow ou PyTorch ou keras
- Construire un CNN simple pour classifier des images (exemple : Dataset MINIST)
- …

Mode d'évaluation :

examen 60% , CC=40%

Bibliographie :

- Ganascia, J.Gabriel (2024) : l'IA expliquée aux humains. Paris France- Edition le Seuil.
- Anglais, Lise, Dilhac, Antione, Dratwa, Jim et al. (2023) : L'éthique au coeur de l'IA. Quebec Obvia.
- J.Robert (2024) : Natural Language Processing (NLP) : définition et principes – Datasciences. Lien : https://datascientest.com/introduction-au-nlp-natural-language-processing
- Qu'est-ce que le traitement du langage naturel. Lien : https://aws.amazon.com/fr/what-is/nlp/
- M.Journe : Eléments de Mathématiques discrètes – Ellipses
- F.Challet : L'apprentisage profond avec Python – Eyrolles
- H.Bersini (2024) : L'intelligence artificielle en pratique avec Python – Eyrolles
- B.Prieur (2024) : Traitement automatique du langage naturel avec Python – Eyrolles
- V.Mathivet (2024) : Implémentation en Python avec Scikit-learn – Eyrolles
- G.Dubertret (2023) : Initiation à la cryptographie avec Python – Eyrolles
- S.Chazallet (2023) : Python 3 – Les fondamentaux du langage - Eyrolles
- H.Belhadef, I.Djemal : Méthode TALN – Cours de l'unievrsité de Msila - Algérie

Semestre: S2
Unité d'enseignement : UET 1.1
Matière : Lean Start-Up 2
VHS : 45h00 (Cours : 1h30, TD : 1h30)
Crédits : 2
Coefficient : 2

Objectifs de l'enseignement:

Ce cours prolonge le module Entrepreneuriat et Lean Start-Up 1. L'étudiant y apprend à transformer les hypothèses validées en produit minimum viable (MVP) et à mobiliser l'intelligence artificielle et la méthode TRIZ pour résoudre les contradictions techniques propres à l'innovation. Il se familiarise avec les indicateurs de performance d'une startup, en particulier le modèle AARRR, et apprend à décider, sur la base de ces indicateurs, s'il faut changer de cap (Pivot) ou poursuivre dans la même direction (Persevere). Le cours aborde enfin les sources de financement d'une startup et les stratégies de croissance, avant de se conclure par un projet intégrateur, le Startup Lab, mené en équipe tout au long du semestre et soutenu devant un jury sous la forme d'un pitch final.

Connaissances préalables recommandées:

Avoir suivi le module Entrepreneuriat et Lean Start-Up 1, notamment les notions de démarche Lean Startup, de Customer Discovery et de Business Model Canvas.

Contenu de la matière:

VI. Le produit minimum viable (MVP) (3 semaines)

- Définition et rôle du MVP dans la démarche Lean Startup
- Types de MVP : numérique, expérimental, service simulé
- Test des hypothèses critiques
- Prototypage rapide et outils de création de prototypes (Figma, Glide, Bubble, Canvas)
- Mesurer l'échec dans le délai le plus court possible

VII. Intelligence artificielle et méthode TRIZ au service de l'innovation (3 semaines)

- L'intelligence artificielle dans l'entrepreneuriat : génération d'idées, analyse de marché, développement du MVP, marketing
- Présentation de la méthodologie TRIZ
- Les contradictions techniques et la solution idéale finale
- Combinaison de l'intelligence artificielle et de TRIZ
- Les startups Deep Tech

VIII. Mesure de la performance et gestion de l'échec (3 semaines)

- Les indicateurs clés de performance (KPIs) et le management d'une startup
- Le modèle AARRR : Acquisition, Activation, Rétention, Revenu, Recommandation
- L'adéquation produit-marché (Product-Market Fit) et l'adéquation problème-solution (Problem-Solution Fit)
- Taux de rétention, taux d'attrition (Churn) et rythme de consommation de trésorerie (Burn Rate)
- Les indicateurs d'apprentissage (Learning Metrics)
- La gestion précoce de l'échec et les critères d'arrêt d'un projet (Kill Criteria)

IX. Pivoter ou persévérer : Pivot vs Persevere (2 semaines)

- Le changement de cap partiel ou total (Pivot) et la persévérance (Persevere)
- Les signaux qui motivent un changement de direction
- Les différents types de pivot
- Les indicateurs qui appuient la décision

X. Financement, croissance et expansion (2 semaines)

- Les sources de financement : autofinancement, investisseurs providentiels (business angels), capital-risque, financement participatif
- Les étapes successives du financement d'une startup
- Les stratégies de croissance rapide à faible coût (Growth Hacking)
- Les moteurs de croissance (Engines of Growth) : viral, payant, organique
- L'entonnoir marketing (Funnel Growth) et les boucles de croissance (Growth Loops)
- Les stratégies d'expansion internationale

XI. Projet intégrateur : Startup Lab et soutenance finale (2 semaines)

- Synthèse du projet mené en équipe, du choix du problème à la stratégie de financement et de croissance
- Préparation et présentation du pitch final devant un jury d'évaluation

Mode d'évaluation:

Examen final : 20 %. Travaux pratiques et ateliers : 20 %. Projet collectif (Startup Lab) : 60 %.

Références bibliographiques

1. Maurya, A. (2012). Running Lean.
2. Croll, A., Yoskovitz, B. (2013). Lean Analytics.
3. Doerr, J. (2017). Measure What Matters.
4. Weinberg, G., Mares, J. (2015). Traction: How Any Startup Can Achieve Explosive Customer Growth.
5. Eyal, N. (2014). Hooked: How to Build Habit-Forming Products.
6. Grove, A. S. (1983). High Output Management.
7. Horowitz, B. (2014). The Hard Thing About Hard Things.
8. Agrawal, A., Gans, J., Goldfarb, A. (2018). Prediction Machines.
9. Iansiti, M., Lakhani, K. R. (2020). Competing in the Age of AI.
10. Altshuller, G. (1999). The Innovation Algorithm: TRIZ, Systematic Innovation and Technical Creativity.
11. Ikovenko, S. (2003). TRIZ: The Theory of Inventive Problem Solving.
12. Graham, P. (2013). Do Things That Don't Scale.
13. McKinsey Global Institute (2023). The State of AI in Business.
14. Stanford University (2024). Artificial Intelligence Index Report.

Annexe - Glossaire des concepts clés

Cette liste reprend, en français, les notions avancées mobilisées dans le cours, en continuité directe avec le glossaire du module Entrepreneuriat et Lean Start-Up 1.

Terme	Définition
MVP (produit minimum viable)	Version la plus simple d'un produit, permettant de tester les hypothèses avec un minimum d'effort.
Market Fit	Concordance entre le produit proposé et les attentes du marché.
Product-Market Fit	Situation dans laquelle le produit répond réellement à un besoin du marché.
TRIZ	Méthodologie scientifique et structurée de résolution de problèmes innovants.
Technological Contradiction (contradiction technique)	Situation où l'amélioration d'une caractéristique d'un produit en dégrade une autre.
Deep Tech Startup	Entreprise fondée sur des sciences et des technologies de pointe.
Data-Driven Decision (décision fondée sur les données)	Décision appuyée sur des données mesurées plutôt que sur l'intuition.
KPIs (indicateurs clés de performance)	Indicateurs essentiels au suivi de la performance d'une startup.
AARRR	Modèle de mesure de la croissance en cinq étapes : acquisition, activation, rétention, revenu, recommandation.
Acquisition	Étape consistant à attirer des utilisateurs vers le produit ou le service.
Activation	Moment où l'utilisateur retire une première valeur réelle du produit.
Retention Rate (taux de rétention)	Proportion d'utilisateurs qui continuent d'utiliser le produit dans le temps.
Churn Rate (taux d'attrition)	Proportion d'utilisateurs qui abandonnent le produit.
Burn Rate	Rythme mensuel de consommation du capital disponible.
Runway	Durée pendant laquelle la startup peut fonctionner avant épuisement de ses fonds.
Kill Criteria (critères d'arrêt)	Conditions justifiant l'arrêt d'un projet jugé non viable.
Learning Metrics (indicateurs d'apprentissage)	Indicateurs centrés sur ce que l'équipe a appris, plutôt que sur le profit.
Pivot	Changement radical ou partiel de la stratégie du projet, décidé sur la base des indicateurs.
Persevere (persévérance)	Décision de poursuivre le projet dans la même direction stratégique.
Traction	Preuve tangible et mesurable de croissance sur le marché.
Growth Hacking	Ensemble de stratégies de croissance rapide et peu coûteuses.

Terme	Définition
Engines of Growth (moteurs de croissance)	Mécanismes de croissance d'une startup : viral, payant ou organique.
Viral Coefficient (coefficient viral)	Mesure du taux auquel les utilisateurs en attirent de nouveaux.
Funnel Growth (entonnoir de croissance)	Représentation du parcours de conversion d'un utilisateur.
Growth Loops (boucles de croissance)	Mécanismes de croissance autoalimentée générant continuellement de nouveaux utilisateurs.
LTV (valeur vie client)	Valeur totale qu'un client est susceptible de générer durant toute sa relation avec l'entreprise.
CAC (coût d'acquisition client)	Coût moyen nécessaire pour obtenir un nouveau client.
Network Effects (effets de réseau)	Phénomène par lequel la valeur d'un produit augmente avec le nombre de ses utilisateurs.
Investment Readiness (maturité pour l'investissement)	Degré de préparation d'un projet à lever des fonds auprès d'investisseurs.

V - Programme détaillé par matière du semestre S3

Semestre : 3
Unité d'enseignement : UEF 2.1.1
Matière 1 : Systèmes temps réel
VHS : 67h30 (Cours : 3h00, TD : 1h30)
Crédits : 6
Coefficient : 3

Objectifs de l'enseignement:

Présenter l'architecture d'un système d'exploitation temps réel et les techniques de programmation sous un langage temps réel.

Connaissances préalables recommandées:

- ✓ Conception des systèmes à microprocesseurs;
- ✓ Programmation informatique;

Contenu de la matière :

Chapitre 1 : Généralités, concept et terminologies sur les systèmes d'exploitation (01 semaines)
- Rappels sur l'architecture d'un ordinateur.
- Le système d'exploitation (définition, rôle, structure etc….)
- Notions d'instruction et de Macro-instruction
- Programme, Processus, Fonction …

Chapitre 2 : Introduction aux systèmes temps réel (01 semaines)
- Définition d'un système temps réel
- Contraintes temps réel. Spécifications
- Classification des systèmes temps-réel.
- Caractéristiques et Structure d'un système de commande.

Chapitre 3 : Ordonnancement dans les systèmes d'exploitation classiques (02 semaines)
- Notion de processus et états des processus
- Opération sur les processus
- Critères d'ordonnancement des processus
- Algorithme d'ordonnancement FCFS, SJF, SRTF & Round Robin

Chapitre 4 : Ordonnancement en temps réel (04 semaines)
- Notion de tâches en temps réel. Modélisation et caractérisation des tâches.
- Ordonnancement des tâches indépendantes
- Ordonnancement des tâches dépendantes
- Ordonnancement en situation de surcharge

Chapitre 5 : Ordonnancement en temps réel multiprocesseur (02 semaines)
- Position et formulation du problème
- Comparatif avec l'ordonnancement monoprocesseur
- Anomalies de l'ordonnancement multiprocesseur
- Conditions d'ordonnançabilité
- Les algorithmes Earliest Deadline et Least Laxity

Chapitre 6 : Gestion de la mémoire et de la communication (03 semaines)
- Gestion de la mémoire virtuelle et la mémoire physique (pagination, adressage, allocation …..)
- problèmes de compétition, coopération, synchronisation

- Sémaphore, les moniteurs, …
- Communication inter-tâches et messages

Chapitre 7 : Programmation **(02 semaines)**

- Introduction à la Programmation concurrente
- Gestion des aspects multitâches, Exclusion mutuelle, Synchronisation, Communication … Programmation temps réel (JAVA temps réel, ADA)
- Exemples d'applications.

Mode d'évaluation:

Contrôle continu: 40% ; Examen: 60%.

Références bibliographiques:

[1] . T. Shanley and D. Anderson, PCI System Architecture, Adisson-Wessley.
[2] . H. Son Sang, Advances in Real-Time Systems, Prentice Hall.
[3] . J. W. S. Liu, Real-Time Systems, Prentice Hall, 2000.
[4] . D. Abbott, Linux for embedded and Real-Time systems, 2003, Architectural Press.
[5] . Nicolas Navet, Systèmes temps réel : Ordonnancement, réseaux et qualité de service, Hermès – Lavoisier, Volume 2, 2006.
[6] . Alan C. Shaw, Real-time systems and software, John Willey & Sons, Inc., 2001.
[7] . Francis Cottet et Emmanuel Grolleau, Système Temps Réel de Contrôle Commande, Dunod 2005.
[8] . Nimal Nissanke, Real-Time Systems, Prentice Hall.
[9] . G.Bollela et al., The Real-Time Specification for Java, Ed. Addison-Wesley.
[10] . Cottet Francis, Joèlle Delacroix, Ordonnancement temps réel: Cours et exercices corrigés, Hermès Science Publications, 2000.
[11] . A. Darseoil, P. Pillot, Le Temps Réel en Milieu Industriel, Dunod 1991.
[12] . Y. Trinquet, J.-P. Elloy, Exécutifs Temps Réel, Techniques d'Ingénieur.

Semestre : 03
Unité d'enseignement : UEF 2.1.2
Matière3 : Intelligence artificielle en embarqué
VHS: 45h00 (Cours: 1h30, TD: 1h30)
Crédits: 4
Coefficient: 2

Objectifs de l'enseignement:

Permettre aux étudiants de se familiariser avec les techniques de l'intelligence artificielle (IA) ainsi que ses applications dans les systèmes embarqués.

Connaissances préalables recommandées:

Cette matière exige des connaissances préalables sur les méthodes d'analyse numérique avancée, programmation C, MatLab ou Python.

Contenu de la matière:

Chapitre 1. Introduction à l'intelligence artificielle et les systèmes embarqués (02 semaines)

Intelligence Artificielle (IA) ; Domaines d'application ; Principales techniques liées à l'intelligence artificielle ; Avantages et inconvénients de l'IA ; Systèmes embarqués ; Les systèmes embarqués et les perspectives de l'IA ; Défis de mise en œuvre de l'intelligence artificielle dans la conception embarquée.

Chapitre 2. Méthodes d'apprentissage automatique et profond (05 semaines)

Apprentissage supervisé ; Apprentissage non supervisé ; Apprentissage semi-supervisé ; Apprentissage par renforcement ; Méthodes d'ensemble en apprentissage automatique (bagging, boosting et stacking), Réseaux de neurones (MLP, RBF, RNN, BNNs…) ; Réseaux de neurones convolutif (CNN), réseaux de neurone profond (DCNN, VGG-16, ResNet, LSTM, GRU, ...).

Chapitre 3. Application d'apprentissage automatique et profond (05 semaines)

Exemples simple d'application des algorithmes d'apprentissage automatique (Naive-Bayes, Decsion Tree, Random forest, k-NN, K-Means, svm, PCA, Q-Learning,…) et d'apprentissage profond dans les problèmes de régression, classification, contrôle, partitionnement de données et réduction de dimensions.
Applications : le traitement du signal (audio,..), imagerie (détection d'objet, reconnaissance de forme, segmentation ,..), langage naturel, traduction de texte, détection, diagnostic, contrôle, etc…

Chapitre 4 Implantation d'apprentissage automatique et profond dans les systèmes embarqués (03 semaines)

Concepts de base sur l'apprentissage automatique embarqué ; Elagage ; Quantification et précision mixte ; Architectures embarquées destinées au apprentissage automatique et profond ; Approches reconfigurables (FPGAs), approches CPUs, et GPUs ; Approches microcontrôleurs ; Approches orientées accélérateurs ; Implantation efficace d'unités MAC (multiply–accumulate) ; Optimisations au niveau logiciel et au niveau matériel ; Principaux niveaux d'abstractions ; TinyML ; Cartes embarqué Coral, Jetson Nano, Raspberry Pi, …

Mode d'évaluation:

Contrôle continu: 40% ; Examen: 60%.

Références biblio-webo graphiques:

[1] . Warden, P. and Situnayake, D., 2019. Tinyml: Machine learning with tensorflow lite on arduino and ultra-low-power microcontrollers. O'Reilly Media.

[2] . Paluszek, Michael, and Stephanie Thomas. MATLAB machine learning. Apress, 2016.

[3] . Raschka, S., 2015. Python machine learning. Packt publishing ltd.

[4] . Liu, Y.H., 2017. Python Machine Learning By Example. Packt Publishing Ltd.

[5] . Ketkar, N. and Santana, E., 2017. Deep learning with Python (Vol. 1). Berkeley: Apress.

[6] . Kim, P., 2017. Matlab deep learning. With machine learning, neural networks and artificial intelligence, 130(21).

[7] . Warwick, K., 2013. Artificial intelligence: the basics. Routledge.

[8] . Gajski, D.D., Abdi, S., Gerstlauer, A. and Schirner, G., 2009. Embedded system design: modeling, synthesis and verification. Springer Science & Business Media.

[9] . Arora, Mohit. Embedded system design: Introduction to SoC system architecture. Learning Bytes Publishing, 2016.

[10] . Parab, J., Shinde, S.A., Shelake, V.G., Kamat, R.K. and Naik, G.M., 2008. Practical aspects of embedded system design using microcontrollers. Springer Science & Business Media.

[11] . Alippi, C., 2014. Intelligence for embedded systems (pp. 1-283). Berlin: Springer.

Semestre: 3
Unité d'enseignement: UEF 2.1.2
Matière 1 : Systèmes Embarqués
VHS: 45h00. (Cours: 1h30, TD: 1h30)
Crédits: 4
Coefficient: 2

Objectifs de l'enseignement:

Dans cette matière, les étudiants seront en mesure de spécifier et de produire sur une base Windows une application ou un système embarqué.

Connaissances préalables recommandées:

Le contenu de cette matière suppose la maîtrise des techniques de programmation ainsi qu'une connaissance moyenne d'un système d'exploitation de type Windows ou Unix et de leurs couches réseau.

- ✓ Microcontrôleurs et Microprocesseurs
- ✓ Algorithmique ;
- ✓ Programmation ASM,C/C++,…etc.
- ✓ Notion de base des systèmes d'exploitation
- ✓ Logiciels de développements intégré(Proteus Tool suite,Mplab,CCS)

Contenu de la matière:

Chapitre 1. Introduction aux systèmes embarqués **(2 semaines)**

1.1. Historique des systèmes embarqués
1.2. Définition des systèmes embarqués
1.3. Développement de systèmes embarqués
1.4. Caractéristiques des systèmes embarqués
1.5. Quelques propriétés des systèmes embarqués
1.6. Contraintes de conception d'un système embarqué
1.7. Quelques exemples de systèmes embarqués
1.8. Domaines d'application des systèmes embarqués :
1.9. Les défis de l'embarqué :

Chapitre 2. Architecture des Systèmes Embarqués **(4 semaines)**

2.Aspect matériel d'un système embarqué
2.2 Architecture de Von Neumann
2.3 Architecture de Harvard
2.4 Le microcontrôleur et sa structure interne
2.5 La structure d'un programme en assembleur pour PIC
2.6 La gestion des interruptions PIC16F84
2.7 La gestion des minuteries internes du PIC16F84
(Temporisation et comptages d'impulsions)
2.8 Exemple pratique d'un système embarqué à base de microcontrôleur

Chapitre 3. Langages deProgrammation pour systèmes embarqués **(3 semaines)**

4.1 Les Règles de bases du compilateur C-PCW de CCS
4.2 Les variables et les constantes
4.3 Les opérateurs du langage C-CCS
4.4 Les structures répétitives.
4.5 Les fonctions C-CCS adaptées aux microcontrôleurs

4.6 La gestion de E/S
4.7 La gestion des temporisations
4.8 La gestion des interruptions
4.9 La gestion de la liaison série
4.10 Exemples

Chapitre 4. Système d'exploitation pour l'embarqué et multitâches **(4semaines)**

5.Introduction
5.2 Machine d'état
5.3 Notions de systèmes d'exploitation temps réel pour l'embarqué(RTOS)
5.3.1 L'ordonnanceur(The scheduler)
5.3.2 Services RTOS
5.3.3 Outils de synchronisation et de transmission de messages
5.3.4 exemple d'application

Chapitre 5. Etude de cas :développement d'une application embarquée telle que l' implémentation d'un correcteur PID pour la régulation de la température **(2semaines)**

Mode d'évaluation:

Contrôle continu: 40% ; Examen : 60%.

Références bibliographiques:

[1] . http://beru.univ-brest.fr/~singhoff/supports.html
[2] . Tim Wilmshurst., Designing Embedded Systems with PIC Microcontrollers : Principles and applications.
[3] . Christian Tavernier , Microcontrôleurs AVR : des ATtiny aux ATmega - Description et mise en œuvre
[4] . Christian Tavernier , Microcontrôleurs Pic 18 - Description Et Mise En Oeuvre
[5] . Alexandre Nketsa ; Circuits logiques programmables : Mémoires PLD, CPLD et FPGA, informatique industrielle
[6] . A. Dorseuil and P. Pillot. Le temps réel en millieu industriel. Edition DUNOD, Collection Informatique Industrielle, 1991.
[7] . Ch. Bonnet, I. Demeure , Introduction aux systèmes temps réel,, Edition HERMES
[8] . Ivan Cibrario Bertolotti_ Gabriele Manduchi-Real-Time Embedded Systems_ Open-Source Operating Systems Perspective-CRC Press (2012).
[9] . Eugenio Villar, Maite Veiga (auth.), Juan Carlos LÃ³pez, RomÃ¡n Hermida, Walter Geisselhardt (eds.)-Advanced Techniques for Embedded Systems Design and Test-Springer US (1998)
[10] . Distributed Systems—Concepts and Design, 2nd Ed. Addison-Wesley Publishers Ltd., 1994.
[11] . I. Demeure and C. Bonnet. Introduction aux systèmes temps réel. Collection pédagogique de télécommunications, Hermès, septembre 1999.
[12] . PIC Microcontroller and Embedded Systems By Muhammad Ali Mazidi
[13] . C Programming Language by Kernighan & Ritchie PDF
[14] . http://www.craslab.org
[15] . http://beru.univ-brest.fr/~singhoff/supports.html
[16] . https://www.ukonline.be/cours/embeddedsystems
[17] . https://www.techno-science.net/glossaire-definition/Systeme-embarque-page-3.html
[18] . http://www.embedds.com/
[19] . http://www.keil.com/rtos/
[20] . http://embedded-lab.com/

Semestre: 3
Unité d'enseignement: UEF 2.1.2
Matière 2 :Réseaux et Communications Industriels
VHS: 45h00. (Cours: 1h30, TD: 1h30)
Crédits: 4
Coefficient: 2

Objectifs de l'enseignement:

Permettre à l'étudiant de se familiariser avec les notions de transmission de données numériques, plus particulièrement les différents types de réseaux existants dans le monde industriel. L'accent sera mis sur la compréhension des différentes topologies avec leurs avantages et inconvénients vis-à-vis d'une installation industrielle donnée.

Connaissances préalables recommandées:

- ✓ Réseaux informatiques locaux ;
- ✓ Capteurs ;
- ✓ Automates programmables industriels ;

Contenu de la matière:

Chapitre 1. Généralités sur les bus de terrain (04 semaines)
1.1-Définitions (Bus, Terrain, Réseau, Réseau local, Réseau local industriel (RLI), Réseau local d'entreprise, Quelques noms des réseaux locaux industriels). 1.2-Présentation de l'environnement industriel. 1.3-Architecture d'un RLI (réseaux de terrain, réseaux d'atelier, réseaux d'usine). 1.4-Caractéristiques d'un RLI (le nombre de nœuds, la quantité d'information, le temps de transmission). 1.5-Caractéristiques des données échangées dans un RLI (Nature des messages échangés, Taille des messages). 1.6-Rôle d'un RLI dans une installation industrielle. 1.7-Architecture OSI et RLI (Adaptation du modèle OSI aux RLI, Caractéristiques de la couche physique pour les RLI, Caractéristiques de la sous-couche MAC pour les RLI).

Chapitre 2. : Le bus 485 Modbus (02 semaines)
Rappel sur la norme RS232. La liaison RS485. Le protocole Modbus. Adressage et trame Modbus.

Chapitre 3. Le bus CAN (Controller Area Network) (03 semaines)
Vue globale sur CAN. Modèles CAN OSI. Trames de données CAN et caractéristiques. Méthodes d'accès et principe d'arbitrage. Débits. Hardware du CAN. Services de la couche application. CANopen.

Chapitre 4. Profibus (03 semaines)
Vue globale sur Profibus et caractéristiques. Les trois types de Profibus (DP, FMS et PA). Mode d'accès. Ethernet Industriel et Profinet. Débits.

Chapitre 5. Aperçu sur les réseaux industriels sans fils (03 semaines)
Technologies, protocoles et architectures des réseaux industriels sans fils (WLAN 802.11, Bluetooth, Protocoles HART, Wireless Profibus, Bluetooth, ZigBee, ...). Sécurité des réseaux de communication industriels sans fil.

Mode d'évaluation:

Contrôle continu: 60 % ; Examen: 40 %.

Références bibliographiques:

[1] . Belgacem Jarray, Réseaux industriels : Bus, Interfaces, Ethernet Industriel, Hart. Cours et exercices corrigés. Ellipses, 2017.
[2] . J.F. Hérold, O.Guilloton et P. ANAYA, Informatiques industrielle et réseaux en 20 fiches. Dunod, 2010.
[3] . Jean-Pierre Thomesse, Les réseaux Locaux industriels. Eyrolles, 1994.
[4] . Pascal Vrignat, Réseaux locaux industriels - Cours et travaux pratiques. Gaëtan Morin, 1999.
[5] . Ciame, Réseaux de terrain : critères de sûreté de fonctionnement. Lavoisier, 2009.
[6] . Ciame, Réseaux de terrain : Description et critères de choix. Hermes, 2001.

Semestre: 02
Unité d'enseignement: UEM 2.1
Matière :TP Intelligence artificielle
VHS: 22h30 (TP: 1h30)– Une fois par quinzaine
Crédits: 2
Coefficient: 1

Objectifs de l'enseignement:

Permettre aux étudiants de se familiariser avec l'application des techniques de l'intelligence artificielle (IA) dans les systèmes embarqués. Se familiariser avec les bibliothèques ML et DL à savoir Keras, Scikit-learn, Tensorflow, etc. sous python ainsi que leur implantation sur des cartes de développement (MC, Raspberry Pi, ou FPGA).

Connaissances préalables recommandées:
Cette matière exige des connaissances préalables sur les langages de programmation tel que le C, VHDL, MatLab ou Python.

Contenu de la matière:

TP1. Présentation de kit de développement, le langage de programmation (bibliothèques, toolboxes,…), et les bases de données (open sources).

TP2. Application de l'apprentissage automatique (k-NN, RF, SVM,…) dans les problèmes de régression, classification, contrôle, partitionnement de données et réduction de dimensions.

TP3. Application de l'apprentissage profond (DCNN, VGG-16,..) dans l'imagerie (détection d'objet, classification ,..), langage naturel, traduction de texte, détection des anomalies, et diagnostic.

TP4. Exemples sur l'Implantation d'apprentissage automatique et profond dans les systèmes embarqués (carte embarqué Coral, Jetson Nano, Arduino, FPGA, Raspberry Pi, …)

Mode d'évaluation:
Contrôle continu: 100%

Références biblio-webo graphiques:

[1] . Warden, P. and Situnayake, D., 2019. Tinyml: Machine learning with tensorflow lite on arduino and ultra-low-power microcontrollers. O'Reilly Media.
[2] . Paluszek, Michael, and Stephanie Thomas. MATLAB machine learning. Apress, 2016.
[3] . Liu, Y.H., 2017. Python Machine Learning By Example. Packt Publishing Ltd.
[4] . Ketkar, N. and Santana, E., 2017. Deep learning with Python (Vol. 1). Berkeley: Apress.
[5] . Kim, P., 2017. Matlab deep learning. With machine learning, neural networks and artificial intelligence, 130(21).
[6] . Gajski, D.D., Abdi, S., Gerstlauer, A. and Schirner, G., 2009. Embedded system design: modeling, synthesis and verification. Springer Science & Business Media.
[7] . Arora, Mohit. Embedded system design: Introduction to SoC system architecture. Learning Bytes Publishing, 2016.
[8] . Parab, J., Shinde, S.A., Shelake, V.G., Kamat, R.K. and Naik, G.M., 2008. Practical aspects of embedded system design using microcontrollers. Springer Science & Business Media.
[9] . Alippi, C., 2014. Intelligence for embedded systems (pp. 1-283). Berlin: Springer.

Semestre: 3
Unité d'enseignement: UEM 2.1
Matière 2: TP Systèmes embarqués / TP Systèmes temps réel
VHS: 22h30 (TP: 1h30)
Crédits: 2
Coefficient: 1

Objectifs de l'enseignement:

Faire comprendre aux étudiants les aspects pratiques des systèmes embarqués et temps réel. Conception d'un système embarqué dédié à l'automobile où il sera fait appel Entrées/Sorties analogiques et numériques, Timer, Interruption, Conversion ADC-DAC, Affichage LCD, 7 segments, Communication sérielle (RS232, …), Communication WIFI, Bluetooth, Conception d'une interface IHM.

Connaissances préalables recommandées:

Programmation en C, Commandes DOS.

Contenu de la matière:

A.TP systèmes embarqués - Une fois par quinzaine

Simulation/Implémentation d'applications en utilisant les couples matériel/logiciel suivants: FPGA/VHDL, Arduino/C, Microprocesseur/Assembleur, PIC/C ou Assembleur, DSP/C, …

Exemple d'application : **Conception d'un système de condamnation centrale d'une automobile**

Ci-dessous est exposée, à titre d'exemple, la méthodologie appliquée à la conception d'un système de condamnation centrale d'une automobile. Libre à l'enseignant de choisir une autre application.

Etape initiale : Elaborer un programme qui permet de faire ouvrir et fermer, au moyen d'une commande infrarouge ou radio fréquence, les quatre portes d'une voiture en utilisant la commande TOR avec l'allumage (ou clignotement) de LEDs rouges et l'émission de bips sonores de durées variables.
Variante 1 : Prendre en considération la fermeture réelle des portes (en utilisant des butées (bouton poussoir).
Variante 2 : Prévoir le cas de l'ouverture des portes par erreur (action par accident de la commande infrarouge) sans l'ouverture effective de la porte. Condamnation automatique après un certain délai.
Variante 3 : Prévoir le cas d'une porte mal fermée tandis que la voiture est en mouvement. Alerter le chauffeur : bip sonore, affichage sur le tableau de bord, …
Variante 4 : Cryptage du signal de commande

Amener l'étudiant à imaginer une architecture globale et à proposer différentes techniques de programmation (interruption, polling, fonctions, communication (synchrone asynchrone), gestion des E/S, Affichage, Alarme, …) et estimer à chaque fois le coût de l'application.

B.TP systèmes temps réel - Une fois par quinzaine

TP 01. Savoir utiliser les commandes de base de MS-DOS, Ecriture et exécution d'un script batch DOS. Modification de fichier et lancement de commandes.

TP 02. Initiation aux commandes Linux : Gestion des processus : Créer (lancer), Visualisation (listing) et Arrêter (tuer) des processus externes. Gestion de la mémoire sous Linux (Comprendre l'allocation dynamique de la mémoire, Diagnostiquer quelques problèmes liés à l'allocation dynamique)

TP 03. Traitement d'un exemple simple (cas sémaphore) en utilisant un des langages temps réel.

Mode d'évaluation:

Contrôle continu: 100%

Références bibliographiques:

Semestre: S3
Unité d'enseignement: UEM 2.1
Matière 3: TP Réseaux industriels
VHS: 22h30 **(TP :** 1h30**)**
Crédits: 2
Coefficient: 1

Objectifs de l'enseignement:

A la fin de cette matière avec la matière Cours/TD associée, l'étudiant sera capable d'utiliser les logiciels industriels (Step-7, Unity Pro, Studio 5000, etc.) pour configurer et programmer un réseau industriel (Profibus, Profinet, CAN, Modbus, etc.) contenant les éléments principaux de communication y compris : le PC industriel, l'HMI, la PLC et les modules d'entrée/sortie. Il aura aussi un bon aperçu sur les différents protocoles inclus dans les couches OSI des bus de terrain (méthodes d'accès, structures de trames, codage, etc.).

Connaissances préalables recommandées:

- ✓ Automates programmable (PLC) ;
- ✓ Langages de programmation pour PLC : Contact, Log, List, Stl, Graph (Grafcet) ;
- ✓ Réseau informatique locaux (LAN) ;
- ✓ Capteurs et actionneurs ;

Contenu de la matière:

TP1 : PLC (Entrées/Sorties) avec Step 7, TIA Portal ou Unity Pro. Programmation avec Graph et/ou Ladder et/ou STL et/ou Log et/ou List.
TP2 : Profibus avec STEP 7 ou TIA Portal. Maître avec esclave passive et/ou esclave intelligent. Simulation et/ou pratique. Utilisation d'un analyseur de protocole pour décoder les messages Profibus.
TP3 : Profinet avec STEP 7 ou TIA Portal. PLC1 + PLC2 + HMI ou PC industriel. Simulation et/ou pratique. Utilisation d'un analyseur de protocole pour décoder les messages Profinet.
TP4 : Modbus série avec Unity Pro. Maître + esclave + Ecran d'exploitation. Utilisation d'un analyseur de protocole pour décoder les messages Modbus.
TP5 : Modbus TCP/IP avec Unity Pro. PLC1 + PLC2. Simulation et/ou pratique. Utilisation d'un analyseur de protocole pour décoder les messages Modbus.

TP1 : Implémentation et mise en œuvre sur RS232, RS485.
TP2 : Mise en œuvre du Bus CAN entre modules Arduino.
TP3 : Dialogue Modbus entre un PC maître et dispositif esclave. Communication Modbus maître depuis un automate programmable industriel.
TP4 : Etude de réseau industriel Profibus (à base d'Arduino ou visite d'une usine).
TP5 : Echange données via réseau Ethernet industriel.
TP6 : Elaboration d'un réseau local industriel sans fils.

Mode d'évaluation:

Contrôle continu: 100 %.

Références bibliographiques:

[1] . https://www.se.com/;
[2] . https://support.industry.siemens.com/.
[3] . Belgacem Jarray, Réseaux industriels : Bus, Interfaces, Ethernet Industriel, Hart. Cours et exercices corrigés. Ellipses, 2017.
[4] . J.F. Hérold, O.Guilloton et P. ANAYA, Informatiques industrielle et réseaux en 20 fiches. Dunod, 2010.
[5] . Jean-Pierre Thomesse, Les réseaux Locaux industriels. Eyrolles, 1994.
[6] . Pascal Vrignat, Réseaux locaux industriels - Cours et travaux pratiques. Gaëtan Morin, 1999.
[7] . Ciame, Réseaux de terrain : critères de sûreté de fonctionnement. Lavoisier, 2009.
[8] . Ciame, Réseaux de terrain : Description et critères de choix. Hermes, 2001.

Semestre: 3
Unité d'enseignement: UEM 1.3
Matière 4 : Etude et Réalisation des projets
VHS: 45h00 (Cours: 1h30, TP: 1h30)
Crédits: 3
Coefficient: 2

Objectifs de l'enseignement:
Réaliser une carte électronique qui constitue un système embarqué pour une application donnée. Développement du code et conception matérielle conjointe.

Connaissances préalables recommandées:
Programmation, systèmes à microprocesseurs et à microcontrôleurs.

Contenu de la matière:

Chapitre 1 : Etude d'un logiciel de simulation (3 Semaines)
Prise en main d'un environnement de conception (exemple « *Proteus Design Suite* »), simulation, analyse de circuits électroniques et développement de circuits imprimés

Chapitre 2 : Définition et conduite d'un projet (2 Semaines)
Etude du projet (un système embarqué simple répondant aux besoins d'un secteur particulier tel que le médical, la voiture ou la domotique) établissement du cahier de charge, choix techniques, coût, échéancier et planification de l'exécution des travaux, documentation, choix de composants.

Chapitre 3 : Réalisation de la partie électronique (3 Semaines)

Chapitre 4 : Réalisation de la partie logicielle (3 Semaines)

Chapitre 5 : Simulation et tests (2 Semaines)

Chapitre 6 : Rapport technique (2 Semaines)
Rédaction du dossier technique et soutenance.

Mode d'évaluation:
Contrôle continu: 100% ; Examen: 00%.

Références bibliographiques:

[1] . *https://labcenter.s3.amazonaws.com/downloads/Tutorials.pdf*
[2] . https://en.wikipedia.org/wiki/Proteus_Design_Suite
[3] . http://www.coursexercices.com/PDF_Cours_Exercices_Telecharger.php?q=proteus+ares+tutorial
[4] . https://www.ele.uva.es/~jesman/BigSeti/ftp/Cajon_Desastre/Software-Manuales/EBook%20-%20Proteus%20Manual.pdf
[5] . https://www.arduino.cc/
[6] . https://www.manager-go.com/gestion-de-projet/
[7] . https://www.techno-science.net/definition/729.html
[8] . https://formation.aapq.org/etape.php
[9] . https://www.nutcache.com/fr/blog/demarche-de-projet/

Semestre: 3
Unité d'enseignement : UET 2.1
Matière 1 : Reverse Engineering
VHS : 45h00 (Cours : 1h30 et Atelier : 1h30)
Crédits: 2
Coefficient: 2

Objectifs de l'enseignement :

- Comprendre les principes et les objectifs du Reverse Engineering (RE) dans le domaine des sciences et de technologie (ST),
- S'initier aux outils et aux méthodes du RE dans la spécialité concernée.
- Appréhender la valeur et l'éthique des principes du RE dans le design, la fabrication et l'assurance qualité de produits,
- Encourager la pensée critique, la curiosité technique, l'ingénierie inverse raisonnée et l'innovation,
- Apprendre à analyser, documenter et modéliser un système existant sans documentation initiale.

Compétences visées

- Décomposer et analyser un système existant,
- Reproduire fidèlement un schéma technique ou un modèle 3D à partir d'un produit existant,
- Appliquer des outils de diagnostic et de simulation,
- Travailler en groupe sur un projet exploratoire,
- Identifier les limites juridiques de la rétroconception

A**daptabilité aux spécialités du domaine Sciences et Technologie : é**

- Toutes les spécialités du domaine ST sont concernées suivant
- Exemples de taches : Documentation technique numérique, résultats de veille technologique, Gestion de projets techniques, Collaboration autour de plans, Analyses de rapports, Compréhension de procédés industriels, Suivi de données de production, Techniques de reporting, Prototypage, Essais)

Prérequis :

- Connaissances fondamentales dans la spécialité.

Contenu de la matière :

1. Introduction à la Réverse Engineering

- Historique, enjeux légaux et éthiques du RE,
- Définitions et champs d'application : Approches (matériels, logiciels, procédés…)
- Domaines : maintenance, re-fabrication, cybersécurité, veille concurrentielle

2. Méthodologie générale

- Analyse d'un système "boîte noire" (black box)
- Décomposition fonctionnelle
- Diagrammes de blocs, entrées/sorties, flux d'énergie ou d'information

3. Reverse engineering matériel

- Cartes électroniques : inspection visuelle, repérage de composants
- Utilisation d'outils : multimètre, oscilloscope, analyseur logique
- Reconnaissance de schémas électroniques
- Reconstitution de schémas sous KiCad / Proteus

4. Reverse engineering logiciel

- Analyse statique de binaires (ex : .exe, .hex)
- Décompilation, désassemblage (introduction à Ghidra, IDA Free, ou Hopper)
- Observation de comportements : sniffing, monitoring (ex : Wireshark)
- Cas des microcontrôleurs : lecture mémoire flash, extraction firmware

5. Reverse engineering mécanique

- Numérisation 3D : scanner, mesures manuelles
- Reproduction de modèles CAO à partir de pièces existantes
- Logiciels utilisés : SolidWorks, Fusion360

6. Sécurité et détection d'intrusion

- Reverse engineering dans la cybersécurité : détection de malware, vulnérabilités
- Signature de logiciels, protections contre le RE (obfuscation, chiffrement)

7. Cas d'études réels

- Analyse d'un produit obsolète ou inconnu (souris, alimentation, module Bluetooth, etc.)
- Exemple de rétroconception de pièce mécanique ou système simple (ventilateur, boîtier)

Exemples de TP (base les 4 Génies)

- **Génie Electrique :**

- Rétro-ingénierie d'un module électronique sans schéma
- Exemple : module Bluetooth, relais temporisé
- Objectifs : identifier le fonctionnement, dessiner le schéma, proposer une variante améliorée.
- Identification de composants (IC, transistors, résistances, etc.).
- Utilisation d'outils : multimètre, oscilloscope, analyseur logique.
- Lecture et extraction de firmware depuis un microcontrôleur.
- Introduction à la détection de contrefaçons électroniques.

- **Génie Mécanique :**

- Rétro-ingénierie d'un mécanisme simple
- Exemples : pompe manuelle, clé dynamométrique, mini-presse..
- Démontage mécanique d'un système (pompe, engrenage, vérin…).
- Mesures et reconstruction de plans ou modèles 3D avec logiciel CAO (SolidWorks, Fusion360).
- Identification de matériaux et modes de fabrication.
- Simulation fonctionnelle à partir du modèle recréé.

- **Génie Civil :**

- Analyse d'ouvrages existants sans plans (murs, dalles, structures…).
- Exemples : escalier métallique, appui de fenêtre, coffrage)
- Étude et rétroconception d'un élément de structure existant
- Identification des matériaux, des assemblages et des contraintes.

- Modélisation de l'ouvrage via Revit, AutoCAD ou SketchUp.
- Étude de réhabilitation ou reproduction d'éléments structurels anciens.

- **Génie des Procédés :**
- Rétroconception d'un module de laboratoire
- Exemples : instruments, distillation, filtration, échangeur, réacteur simples…
- Analyse de systèmes industriels existants (colonne de distillation, échangeur, réacteur…).
- Reconstitution des schémas PFD et PID à partir de l'observation d'une installation.
- Identification des capteurs, actionneurs, organes de commande.
- Étude de flux de matière/énergie dans un procédé.

Mode d'évaluation :

- TP techniques
- Mini-projet de rétro-ingénierie (rapport + soutenance)
- Examen final (QCM + étude de cas)

- Examen : 60% et CC TP : 40%

Références bibliographiques :

- Reverse Engineering for Beginners – Dennis Yurichev (gratuit en ligne)
- The IDA Pro Book – Chris Eagle (logiciels)
- Practical Reverse Engineering – Bruce Dang
- Documentation :
  - https://ghidra-sre.org
  - https://www.kicad.org
  - https://www.autodesk.com/products/fusion-360

Semestre : 3
Unité d'enseignement : UET2.1
Matière :Recherche documentaire et conception de mémoire
VHS : 15h00 (Cours : 1h00)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement :
Donner à l'étudiant les outils nécessaires afin de rechercher l'information utile pour mieux l'exploiter dans son projet de fin d'études. L'aider à franchir les différentes étapes menant à la rédaction d'un document scientifique. Lui signifier l'importance de la communication et lui apprendre à présenter de manière rigoureuse et pédagogique le travail effectué.

Connaissances préalables recommandées :
Méthodologie de la rédaction, Méthodologie de la présentation.

Contenu de la matière:

Partie I- : Recherche documentaire :

Chapitre I-1 : Définition du sujet (02 Semaines)
- Intitulé du sujet
- Liste des mots clés concernant le sujet
- Rassembler l'information de base (acquisition du vocabulaire spécialisé, signification des termes, définition linguistique)
- Les informations recherchées
- Faire le point sur ses connaissances dans le domaine

Chapitre I-2 : Sélectionner les sources d'information (02 Semaines)
- Type de documents (Livres, Thèses, Mémoires, Articles de périodiques, Actes de colloques, Documents audiovisuels…)
- Type de ressources (Bibliothèques, Internet…)
- Evaluer la qualité et la pertinence des sources d'information

Chapitre I-3 : Localiser les documents (01 Semaine)
- Les techniques de recherche
- Les opérateurs de recherche

Chapitre I-4 : Traiter l'information (02 Semaines)
- Organisation du travail
- Les questions de départ
- Synthèse des documents retenus
- Liens entre différentes parties
- Plan final de la recherche documentaire

Chapitre I-5 : Présentation de la bibliographie (01 Semaine)
- Les systèmes de présentation d'une bibliographie (Le système Harvard, Le système Vancouver, Le système mixte…)
- Présentation des documents.
- Citation des sources

Partie II : Conception de mémoire

Chapitre II-1 : Plan et étapes du mémoire (02 Semaines)

- Cerner et délimiter le sujet (Résumé)
- Problématique et objectifs du mémoire
- Les autres sections utiles (Les remerciements, La table des abréviations…)
- L'introduction (*La rédaction de l'introduction en dernier lieu)*
- État de la littérature spécialisée
- Formulation des hypothèses
- Méthodologie
- Résultats
- Discussion
- Recommandations
- Conclusion et perspectives
- La table des matières
- La bibliographie
- Les annexes

Chapitre II- 2 : Techniques et normes de rédaction (02 Semaines)

- La mise en forme. Numérotation des chapitres, des figures et des tableaux.
- La page de garde
- La typographie et la ponctuation
- La rédaction. La langue scientifique : style, grammaire, syntaxe.
- L'orthographe. Amélioration de la compétence linguistique générale sur le plan de la compréhension et de l'expression.
- Sauvegarder, sécuriser, archiver ses données.

Chapitre II-3 : Atelier : Etude critique d'un manuscrit **(01 Semaine)**

Chapitre II-4 : Exposés oraux et soutenances (01 Semaine)

- Comment présenter un Poster
- Comment présenter une communication orale.
- Soutenance d'un mémoire

Chapitre II-5 : Comment éviter le plagiat ? **(01 Semaine)**

(Formules, phrases, illustrations, graphiques, données, statistiques,...)

- La citation
- La paraphrase
- Indiquer la référence bibliographique complète

Mode d'évaluation :

Examen : 100%

Références bibliographiques :

1. *M. Griselin et al., Guide de la communication écrite, 2e édition, Dunod, 1999.*
2. *J.L. Lebrun, Guide pratique de rédaction scientifique : comment écrire pour le lecteur scientifique international, Les Ulis, EDP Sciences, 2007.*
3. *A. Mallender Tanner, ABC de la rédaction technique : modes d'emploi, notices d'utilisation, aides en ligne, Dunod, 2002.*
4. *M. Greuter, Bien rédiger son mémoire ou son rapport de stage, L'Etudiant, 2007.*
5. *M. Boeglin, lire et rédiger à la fac. Du chaos des idées au texte structuré. L'Etudiant, 2005.*
6. *M. Beaud, l'art de la thèse, Editions Casbah, 1999.*
7. *M. Beaud, l'art de la thèse, La découverte, 2003.*
8. *M. Kalika, Le mémoire de Master, Dunod, 2005.*

Proposition de quelques matières de découverte

Semestre:
Unité d'enseignement: UED
Matière 1:Radio-identification RFID
VHS: 22h30 (Cours: 1h30)
Crédits: 1
Coefficient: 1

Objectifs de l'enseignement:

Acquérir ses notions techniques et pratiques suffisantes sur la technologie RFID en vue de son implémentation dans des projets sur les systèmes embarqués.

Connaissances préalables recommandées:

Architecture des systèmes à microprocesseurs

Contenu de la matière:

- Présentation, définition et historique
- Ethique, vie privée,
- Obstacles à l'utilisation de la rfid : environnement métallique, collisions,
- Classifications des tags RFID
- Principe de fonctionnement
- lecture seule ou lecture/écriture ?
- protocoles TTF et ITF
- Les applications du RFID
- la communication en champ: NFC
- le fonctionnement de NFC/RFID
- les caractéristiques principales
- Les applications du NFC

Mode d'évaluation:

Examen: 100 %.

Références bibliographiques :

1. *D. Henrici, RFID Security and Privacy: Concepts, Protocols, and Architectures, Springer-Verlag 2008*
2. *K. Finkenzeller, RFID Handbook: Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication, 3d edition, Wiley 2010*
3. *Syed Ahson and Mohammad Ilyas, RFID Handbook : Applications, Technology, Security, and Privacy, CRC Press 2008*
4. *http://www.centrenational-rfid.com/fonctionnement-dun-systeme-rfid-article-17-fr-ruid-17.html*
5. *https://fr.wikipedia.org/wiki/Protocoles_de_communication_RFID*
6. *https://fr.wikipedia.org/wiki/Radio-identification*
7. *https://fr.wikipedia.org/wiki/Communication_en_champ_proche*

Semestre:
Unité d'enseignement: UED
Matière 2:Domotique
VHS: 22h30 (Cours: 1h30)
Crédits: 1
Coefficient: 1

Objectifs de l'enseignement:

Donner à l'étudiant l'ensemble des équipements d'une maison intelligente, leur fonctionnement et leurs 'utilisations afin qu'il soit capable de dimensionner et de concevoir une installation domotique.

Connaissances préalables recommandées:

Systèmes à microprocesseurs, Capteurs,…

Contenu de la matière:

Chapitre 1. Confort dans les bâtiments (1 Semaine)
- Thermique, acoustique et visuel,

Chapitre 2. Notions sur la sécurité des biens et des personnes (7 Semaines)
- Sécurité incendie, Contrôle d'accès, Anti-intrusion, Vidéosurveillance, Télésurveillance, ...

Chapitre 3.Gestion technique des bâtiments et communication (7 Semaines)
- Eclairage, Climatisation, Chauffage, Régulation, Réseaux, Télégestion, Supervision, GTB (gestion technique du bâtiment), GTC (gestion technique centralisée), ...

Mode d'évaluation:

Examen: 100%.

Références bibliographiques :

1. *C. Locqueneux, Le Guide de la Maison et des Objets Connectés, Eyrolles 2016*
2. *F-X. Jeuland, La Maison communicante, Eyrolles, 2008 (2e édition)*
3. *PROMOTELEC, Habitat communicant, Éditions Promotelec, 2006*
4. *E. A. Decamps, La Domotique, Presses universitaires de France, Collection « Que sais-je ? », 1988.*
5. *M. Al-Qutayri, Smart Home Systems, In-Teh, Croatia 2010*
6. *C. Nugent, Smart Homes and Beyond, IOS Press, Netherlands 2006*

Semestre :
Unité d'enseignement : UED
Matière 3 : Systèmes embarqués pour l'automobile
VHS : 22h30 (Cours : 1h30)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement:

Cette matière a pour objectif de donner aux étudiants les fondements nécessaires pour savoir développer et concevoir des applications de l'électronique embarquée au service de l'automobile qui est une discipline à part entière visant à maitriser de façon optimale la circulation et la sécurité d'un véhicule.

Connaissances préalables recommandées :

Capteurs et Instrumentation.

Contenu de la matière :

Chapitre 1 :Introduction aux systèmes embarqués

Chapitre 2 :Capteurs embarqués
Capteurs de vitesse et de débit, Capteurs d'accélération, Capteurs de température, Capteurs de pression, Capteurs de proximité, Capteurs gyroscopique.

Chapitre 3 : Actionneurs embarqués
Actionneur hydraulique, Actionneur pour Air Bag, Système de climatisation, Système de freinage.

Chapitre 4 :Architecture du système véhicule
Calculateur électronique, Bus de communication CAN, Réseaux capteurs/Actionneurs.

Chapitre 5 : Systèmes embarqués dans l'automobile
Systèmes des capteurs embarqués, Système anti-blocage des roues (ABS), Système anti-patinage des roues (ASR),Régulation électronique du comportement dynamique (ESP), Mesure de la vitesse de rotations des roues (codeur) et de la vitesse du véhicule (effet Doppler).

Chapitre 6 :Architecture type d'un modèle de véhicule fabriqué en Algérie

Mode d'évaluation :

Examen : 100 %

Références bibliographiques:

1. *S. Daly, Automotive Air Conditioning and Climate Control Systems, Elsevier, 2006.*
2. *J. Fenton Advances in Vehicle Design, Mechanical Engineering Publications Ltd, 1999.*
3. *B. Hollembeak, Today's Technician: Automotive Electricity and Electronics Classroom and Shop Manual Pack, 5th edition, Delmar, 2010.*
4. *N. Zaman, Automotive Electronics Design Fundamentals, Springer, 2015.*
5. *G. Asch, Acquisition de données : du capteur à l'ordinateur, Dunod, 2003.*
6. *G. Asch et al. Acquisition de données, 3e édition, Dunod, 2011.*
7. *M. Bayart, B. Conrard, A. Chovin, M. Robert, Capteurs et actionneurs intelligents, 2005.*
8. *P. Dassonvalle, Les Capteurs : Exercices et problèmes corrigés, Dunod, 2005.*
9. *R. Frank, Understanding Smart Sensors (Artech House sensors library), 2nd edition.*
10. *F. Boudoin, M. Lavabre, Capteurs : principales utilisations, Edition Casteilla, 2007.*
11. *J. G. Webster, Measurement, Instrumentation and Sensors Handbook, Taylor & Francis Ltd*

Semestre :
Unité d'enseignement : UED
Matière 4 : Systèmes d'exploitation des systèmes embarqués
VHS : 22h30 (Cours : 1h30)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement:

Permettreaux étudiants de connaître les principes fondamentaux des systèmes d'exploitation tout en étudiant leur mise en pratique dans un système embarqué tel que l'Android.

Connaissances préalables recommandées :

Notions de base en mathématiques, algorithmique et programmation.

Contenu de la matière :

Chapitre 1. **(4 semaines)**
Présentation générale des systèmes d'exploitation et éléments techniques (ex : OS, Android, windows et linux)

Chapitre 2. **(3 semaines)**
Gestion des processus

Chapitre 3. **(3 semaines)**
Gestion de la mémoire

Chapitre 4. **(3 semaines)**
Gestion des fichiers

Chapitre 5. **(2 semaines)**
Exécutable

Mode d'évaluation :

Examen : 100 %

Références bibliographiques:

1. *Andrew Tanenbaum, Systèmes d'exploitation, édition Pearson.*
2. *Michael Griffiths, Michel Vayssade, Architecture des systèmes d'exploitation, Hermès - Lavoisier.*
3. *P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer TinyOS: An Operating System for Sensor Networks in Ambient Intelligence, p. 115-148, Springer, 2005.*
4. *P. Levis, D. Gay, TinyOS Programming, Cambridge University Press, 2009.*
5. *TinyOS Open Technology Alliance : http://www.cs.berkeley.edu/~culler/tinyos/ alliance/overview.pdf*
6. *www.contiki-os.org/support.html*

Semestre :
Unité d'enseignement : UED
Matière 5 : Cartes à puces
VHS : 22h30 (Cours : 1h30)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement:

Acquérir des notions techniques suffisantes sur la technologie, le fonctionnement et l'utilisation des cartes à puces en vue de son implémentation dans des projets sur les systèmes électroniques embarqués.

Connaissances préalables recommandées :

Architecture des systèmes à microcontrôleurs et/ou à microprocesseurs.

Contenu de la matière :

- Généralités, Historique,Applications et marchés de la carte à puce.
- Semi-conducteurspour cartes à puces, Technologies, Composants en logique câblée, Microcalculateurs.
- Cryptologie et sécurité, Principes de la cryptographie, Crypto systèmes symétriques, Crypto systèmes asymétriques, Crypto systèmes à apport nul de connaissance,Sécurité physique et logique des cartes à puces.
- Principes de construction, Interconnexion des composants, Encartage, Connectique.
- Systèmes d'exploitation des cartes à puce, Généralités et mécanismes de base, Systèmes d'exploitation fermés, Systèmes d'exploitation ouverts.
- Communication par contact, Communication par radiofréquences.

Mode d'évaluation :

Examen : 100 %

Références bibliographiques:

1. *W.Rankl and W. Effing, Smart Card Handbook, Wiley, 2010.*
2. *C. Tavernier, Les cartes à puce, Dunod, 2011.*

Semestre :
Unité d'enseignement : UED
Matière 6 : Robotique mobile
VHS : 22h30 (Cours : 1h30)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement:

Cette matière a pour objectif de donner aux étudiants les fondements nécessaires pour savoir développer et concevoir des applications de l'électronique embarquée au service de la robotique mobile qui est une discipline à part entière visant à maitriser le mouvement.

Connaissances préalables recommandées :

Mathématiques, Programmation.

Contenu de la matière :

Chapitre 1 : Classification et Modélisation des robots mobiles (R-M) (2 semaines)

Chapitre 2 : Les capteurs utilisés en R-M (3 semaines)

Chapitre 3 : La localisation des R-M (2 semaines)

Chapitre 4 : La représentation de l'environnement d'un R-M (2 semaines)

Chapitre 5 : Les techniques de planifications de trajectoires (2 semaines)

Chapitre 6 : Les techniques de navigation et de SLAM (2 semaines)

Chapitre 7 : Les robots humanoïdes (2 semaine)

Mode d'évaluation :

Examen : 100 %

Références bibliographiques:

1. *R. Siegwart, I.R. Nourbakhch, D. Scaramuzza, Introduction to Autonomous Mobile Robots, 2nd Edition, MIT Press, 2011.*
2. *L. Jaulin, La robotique mobile, Editions ISTE, 2015.*
3. *V. Maille, C. Accard, B. Breton, Les robots : apprendre la robotique par l'exemple, Editions Ellipse, 2016.*

Semestre :
Unité d'enseignement : UED
Matière 7 : Communication sans fils
VHS : 22h30 (Cours : 1h30)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement:

Permettre aux étudiants de se familiariser avec les systèmes de communications sans fils afin des les utiliser dans des applications en électronique des systèmes embarqués. Parmi les divers réseaux sans fils, nous nous intéresserons plus spécifiquement au réseau WIFI de la norme IEEE 802.11.

Connaissances préalables recommandées :

Notions de bases en mathématiques, statistiques et traitement de signal.

Contenu de la matière :

Chapitre 1. Les réseaux sans fils **(4 semaines)**

Chapitre 2. Présentation du WIFI (802.11) : **(3 semaines)**
Différents normes et équipements WIFI

Chapitre 3. Mise en œuvre du WIFI : **(3 semaines)**
Mode infrastructure, mode ad hoc et mise en place d'un réseau

Chapitre 4. Le chiffrement : **(3 semaines)**
WEP, WAP …

Chapitre 5. Les piratages et les solutions : **(2 semaines)**
Filtrage des adresses MAC, définitions des adresses IP et installation des pare-feu

Mode d'évaluation :

Examen : 100 %

Références bibliographiques:

1. *Tanenbaum, Réseaux, 4ème édition, Prentice Hall, 2003.*
2. *R. Parfait, Les réseaux de télécommunications, Hermès science publications, 2002.*
3. *E. Hollocou, Techniques et réseaux de télécommunications, Armand Colin, 1991.*
4. *C. Servin, Réseaux et télécoms, Dunod, Paris, 2006.*
5. *D. Dromard et D. Seret, Architectures des réseaux, Editions Pearsont, 2009.*
6. *P. Polin, Les réseaux : Principes fondamentaux, Edition Hermès.*
7. *D. Comer, TCP/IP, architectures, protocoles et applications, Editions Interéditions.*
8. *D. Présent, S. Lohier, Transmissions et Réseaux, cours et exercices corrigés, Dunod.*
9. *P. Clerc, P. Xavier, Principes fondamentaux des Télécommunications, Ellipses, Paris, 1998.*
10. *D. Battu, Initiation aux Télécoms : Technologies et Applications, Dunod, Paris, 2002.*
11. *P. Rolin, G. Martineau, L. Toutain, A. Leroy, Les réseaux, principes fondamentaux, édition Hermès, 1997.*

Semestre :
Unité d'enseignement : UED
Matière 8 : Robotique
VHS : 22h30 (Cours : 1h30)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement:

Introduire l'étudiant aux aspects fondamentaux dela robotique et aux récents développements dans le domaine de la robotique industrielle.

Connaissances préalables recommandées :

Aucune.

Contenu de la matière :

Chapitre 1 : Généralités
Définitions, Constituants d'un robot, Classification des robots, Caractéristiques d'un robot, Les générations de robot, Programmation des robots.

Chapitre 2 : Degré de libertés - Architecture
Positionnement d'un solide dans l'espace, Liaison, Mécanismes, Morphologie des robots, manipulateurs

Chapitre 3 : Modèle géométrique d'un robot en chaîne simple
Nécessité d'un modèle, Coordonnées opérationnelles, Translation et rotation, Matrices de transformation homogène, Obtention du modèle géométrique, Paramètres de Denavit-Hartenberg modifié, Inversion du modèle géométrique - Méthode de Paul, Solutions multiples – Espace de travail – Aspects

Chapitre 4 : Technique de simplification
Vitesse et accélération des robots, Matrice Jacobéenne et son utilité, Définition des équations en direct et en inverse, Signification des singularités.

Mode d'évaluation :

Examen : 100 %

Références bibliographiques:

1. *H. Asada, J.J.E. Slotine, Robot Analysis and Control, a Wiley Interscience Publication, 1986.*
2. *J.J. Craig, Introduction to Robotics, Mechanics and Control, Addison-Wesley, 1989.*

Semestre :
Unité d'enseignement : UED
Matière 9 : Energies renouvelables : le solaire photovoltaïque
VHS : 22h30 (Cours : 1h30)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement:

Cette matière aborde des notions relatives aux énergies renouvelables non polluantes, aux dispositifs photovoltaïques (PV), à la conversion PV, aux procédés de fabrication d'une cellule solaire, aux assemblages des modules PV, à leur dégradation ...Elle aborde,en outre, les systèmes auxiliaires : la batterie, la pile à combustible(avec l'hydrogène comme vecteur d'énergie), les convertisseurs, … La matière s'intéressera,par ailleurs, aux différentes charges à alimenter continues ou alternatives en recherchant toutes les possibilités de couplage avec un générateur PV, à la description d'un système PV global, ses caractéristiques et à l'optimisation du fonctionnement du système.

Connaissances préalables recommandées :

Notions sur les semi-conducteurs, la physique du rayonnement, les mathématiques, l'électronique...

Contenu de la matière :

Chapitre 1 :Energies renouvelables
Les formes d'énergie, C'est quoi une énergie renouvelable, Principales énergies renouvelables,La situation énergétique mondiale, …

Chapitre 2 :La source solaire
Rayonnement solaire, Le gisement solaire, L'énergie solaire (thermique, photovoltaïque, thermo-dynamique)

Chapitre 3 :La source photovoltaïque
La conversion photovoltaïque, Technologie des cellules solaires, Propriétés des cellules solaires, Modélisation d'une cellule (d'un module) photovoltaïque (modélisation électrique, thermique …), Rendement de conversion, facteur de forme …, Différentes connexions (série, parallèle, mixte), Impact de divers facteurs sur les caractéristiques électriques, Dégradation, Protections des modules photovoltaïques, Les applications de l'énergie photovoltaïque(pompage, connexion au réseau, …).

Chapitre 4 : Systèmes photovoltaïques
Connexion directe générateur photovoltaïque –charge, Stockage(Batterie), Pile à combustible, Hacheur, Onduleur, Etude d'un exemple de système global(l'hybridation), Problème de dimensionnementd'une installation photovoltaïque, *Maximum Power Point Tracker* (MPPT).

Mode d'évaluation :

Examen : 100 %

Références bibliographiques:

1. *A. Vapaille, Dispositifs et circuits intégrés semiconducteurs, Dunod, 1987.*
2. *M. Orgeret, les piles solaires, Masson, 1985.*
3. *A. Ricaud, Photopiles solaires, Presses polytechniques et universitaires romandes, 1997.*
4. *E.Lorenzo, G. Araflio, Solar Electricity - Engineering of Photovoltaic Systems.*
5. *Minano, R. Zilles, Stand alone photovoltaic Applications, JAMES & JAMES 1994.*
6. *B. Multon, Production d'énergie électrique par sources renouvelables, Techniques de l'Ingénieur, Traités de Génie Electrique, D4005/6, mai 2003.*
7. *J. Nelson, The physics of solar cells, Imperial College Press.*

8. *A. Labouret, P. Cumune, Cellules solaires, 5e édition - Les bases de l'énergie photovoltaïque, Dunod, 2010*
9. *A. Labouret , Energie solaire photovoltaïque, 3ème édition, Dunod, 2006.*
10. *Deambi, Suneel, Photovoltaic System Design: Procedures, Tools and Applications, CRC Press, 2016.*
11. *O. Isabella, K. Jäger, A. Smets, R. Van Swaaij, MiroZeman, Solar Energy: The Physics and Engineering of Photovoltaic Conversion, Technologies and Systems, UIT Cambridge Ltd, 2016.*
12. *Gottfried H. Bauer, Lecture Notes in Physics 901, Photovoltaic Solar Energy Conversion, Springer-Verlag Berlin Heidelberg, 2015.*
13. *www.pveducation.org*
14. *http://www.cythelia.fr/nos-documents/*
15. *http://www.solems.com/depots-de-couches-minces*

Semestre :
Unité d'enseignement : UED
Matière 10 : Systèmes énergétiques autonomes
VHS : 22h30 (Cours : 1h30)
Crédit : 1
Coefficient : 1

Objectifs de l'enseignement:

Susciter l'intérêt de l'étudiant aux énergies renouvelables en général et aux systèmes énergétiques exploitant l'énergie solaire ou éolienne en particulier. Faire acquérir à l'étudiant une certaine compétence dans le dimensionnement d'une installation éolienne ou photovoltaïque.

Connaissances préalables recommandées :

les semi-conducteurs , la physique du rayonnement , les mathématiques, l'électronique...

Contenu de la matière :

Chapitre 1 : Dispositifs de production d'énergie électrique
Notions sur les transformations d'énergie (mécanique ; thermique ; hydraulique, …), Historique (Volta, Oersted, Faraday, etc.), l'alternateur, la dynamo, les modes de production de l'énergie électrique (centrale électrique hydraulique, les centrales thermiques). Les sources d'énergies non renouvelables (fossiles et nucléaires). Les sources d'énergies renouvelables.

Chapitre 2 : Energie éolienne
Historique, principe et structure, Caractéristiques et dimensionnement, Carte du gisement éolien en Algérie, Parcs éoliens et puissance, Normes, Avantages et inconvénients. Exemple d'une installation éolienne.

Chapitre 3 : Systèmes hybrides
Systèmes Hybrides (Hydrolienne, Principe de fonctionnement de l'hydrolienne, Les différents types d'hydroliennes et les exploitants,…)

Chapitre 4 : Energie solaire photovoltaïque
Principe d'une installation photovoltaïque, le gisement solaire en Algérie, Technologies des cellules photovoltaïques, Les modules photovoltaïques, MPPT, Caractéristiques et connectique photovoltaïque, Normes. L'onduleur (rôle, principe, caractéristiques et rendement). Exemple d'une installation photovoltaïque.

Chapitre 5 : Autres sources d'énergies renouvelables
Les familles d'énergie renouvelables (énergie solaire, énergie éolienne, énergie hydraulique, Biomasse, Géothermie). Les différentes énergies renouvelables dans le monde. Rentabilité.

Mode d'évaluation :

Examen : 100 %

Références bibliographiques:

1. *J. Vernier, Les énergies renouvelables, édition PUF, 2012*
2. *E. Riolet, Le mini-éolien, édition Eyrolles, 2010*
3. *A. Labouret et M. Villoz, Energie solaire photovoltaïque, Editions du Moniteur 2009*
4. *B. Fox, Energie électrique éolienne : Production, prévision et intégration au réseau, Collection Technique et Ingénierie, Dunod/L'Usine Nouvelle 2015 (2e édition)*
5. *A. Damien, La biomasse énergie: Définitions, ressources et modes de transformation, Collection Technique et Ingénierie, Dunod/L'Usine Nouvelle 2013 (2 e édition)*

6. *A. Labouret, M. Villoz, Installations photovoltaïques: Conception et dimensionnement d'installations raccordées au réseau, Collection Technique et Ingénierie, Dunod/Le Moniteur 2012 (5e édition)*
7. *http://www.cder.dz/spip.php?article1442*

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

Université.....
Faculté.....
Département.....

Filière :.....
Spécialité :.....
Année universitaire 20.... /20....

Le...../...../**20**....

PROCES VERBAL CONCERNANT LE CHOIX DES MATIERES DE DECOUVERTE DE LA PREMIERE ANNEE MASTER

Les enseignants, soussignés, après délibération ont arrêté le choix des matières de découverte du master ….. proposées dans le canevas de ce master. A ce propos, les enseignants* dont les noms suivent s'engagent à assurer l'enseignement de ces matières. Dans le cas où l'équipe de formation choisit une matière dont le programme n'est pas disponible dans le canevas, le responsable de la filière s'engage à adresser ce programme au CPND-ST pour enrichissement et validation.

* A chaque matière, il est possible d'indiquer le nom de l'enseignant principal et éventuellement le nom d'un enseignant suppléant.

Semestres	**Matières de découverte**	**Enseignants**
S1	Matière 1 : Matière 2 :	
S2	Matière 1 : Matière 2 :	

Observations :.....
.....

Noms et Prénoms des enseignants		**Matières enseignées**	**Semestre**	**Emargements**
1				
2				
3				
4				
5				
6				
7				
8				
9				

Le responsable de la filière Le chef du département

***Rappels** : La nature des matières de découverte doivent apporter un complément à la formation et doivent être choisies en fonction des besoins du tissu socio-économique local ou régional et de la disponibilité des enseignants spécialistes en la matière.*

Copies aux VRP/VDP

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

Université.....
Faculté.....
Département.....

Filière :.....
Spécialité :.....
Année universitaire 20.... /20....

Le...../...../**20**....

PROCES VERBAL CONCERNANT LE CHOIX DES MATIERES DE DECOUVERTE DE LA DEUXIEME ANNEE MASTER

Les enseignants, soussignés, après délibération ont arrêté le choix des matières de découverte du master ….. proposées dans le canevas de ce master. A ce propos, les enseignants* dont les noms suivent s'engagent à assurer l'enseignement de ces matières. Dans le cas où l'équipe de formation choisit une matière dont le programme n'est pas disponible dans le canevas, le responsable de la filière s'engage à adresser ce programme au CPND-ST pour enrichissement et validation.

* A chaque matière, il est possible d'indiquer le nom de l'enseignant principal et éventuellement le nom d'un enseignant suppléant.

Semestres	**Matières de découverte**	**Enseignants**
S3	Matière 1 : Matière 2 :	

Observations :.....
.....

Noms et Prénoms des enseignants		**Matières enseignées**	**Semestre**	**Emargements**
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

Le responsable de la filière Le chef du département

***Rappels** : La nature des matières de découverte doivent apporter un complément à la formation et doivent être choisies en fonction des besoins du tissu socio-économique local ou régional et de la disponibilité des enseignants spécialistes en la matière.*

Copies aux VRP/VDP