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

TRAINING OFFER

ACADEMIC MASTER

Institution	Faculty	Department
Djilali Bounaama University of Khemis Miliana	Sciences and Technologies	Electronics and Telecommunications

Domaine : Sciences et Technologies (ST)
Filière : Electronique
Spécialité : Electronique des systèmes embarqués

Année universitaire : 2022-2023

I – Master's identity card

A. Access conditions

(Indicate the bachelor's degree specializations that can provide access to the Master's degree)

Sector	Harmonized Master	Licenses providing access at the master's level	Ranking by license compatibility	Coefficient assigned to the license
Electronic	Embedded systems electronics	Electronic	1	1.00
		Telecommunications	2	0.80
		Biomedical Engineering	2	0.80
		Automatic	3	0.70
		Electrical engineering	3	0.70
		Electro mechanics	4	0.65
		Other licenses in the ST domain	5	0.60

B. Training objectives

Electronics is present in most everyday objects in a wide variety of forms. Today, it is one of the key sectors of the global economy, either at the origin of or contributing to many innovative fields such as computing, embedded systems and telecommunications. This discipline covers the analysis and design of components, circuits and systems, as well as hardware/software co-integration for control, command, calculation and interface in complex systems.

An embedded system is an electronic and computer system within a larger product. Its primary purpose is to process information received from its environment for the benefit of the host product. The embedded system must be able to provide the exact results within the defined timeframe, often in real time. IT, healthcare, the military... The devices used in certain sectors must rigorously guarantee user security and data confidentiality.

The aim of the 'Embedded Systems Electronics' Masters course is to offer advanced training for research in the fields of electronics, robotics, signal and information processing and telecommunications systems. The aim is to train scientific managers specialising in the analysis and design of electronic systems for embedded applications. Graduates of this Master's programme will become future senior managers and researchers capable of holding positions in a wide range of sectors, including industrial production, radio communication operators, services, equipment manufacturers, the automotive industry, railways, energy management, production and management of electrical energy, and electrical and electronic systems.

The knowledge acquired enables the understanding and development of devices at several levels of description, from the electronic chip to the embedded system. The course provides an understanding of the specific features of embedded systems, such as energy management, electromagnetic compatibility between the various components and data transmission aspects. Interaction with low-level software is also covered, as its study is necessary to understand the complexity of embedded systems.

The aim is therefore to acquire skills in the field of new information and communication technologies and industry. It also provides training for future researchers in industrial microcomputing, whose role is to study the still open problems of communication theory and to solve the problems encountered when implementing the proposed solutions in real applications.

This training is based on both fundamental subjects and subjects related to automatic electronics. To this end, a great deal of time has been devoted to practical work sessions: almost all the specialised subjects are supported by practical work sessions.

At the same time, students are trained to work as part of a team to encourage autonomy, a sense of responsibility and a spirit of initiative, thanks to a course that includes an end-of-study project.

C. Target profiles and skills

One of the priorities that has strongly guided our policy of training for an academic Master's degree is both the extent and diversity of the embedded systems problems facing Algeria. This Master's degree in Embedded Systems Electronics has an academic vocation. Its main aims are to

Train students capable of pursuing studies in any type of PhD,

Lead to a diploma recognised by the socio-economic environment (regional and national) and adapted to the current and future needs of our society.

In fact, this training offers numerous professional opportunities in a wide variety of industries such as the automotive, radio, television, telephony, medical, robotics, imaging, industrial computing and embedded systems sectors.

D. Regional and national employability potential of graduates

On completion of this course, the student will be able to work as a multi-skilled manager in the analysis and design of electronic systems dedicated to on-board applications, particularly in space, and to telecommunications, in response to both national and regional needs:

Sectors of activity

- Industry SME/SMI
- Design offices
- Technical sales
- Training

On a national level

- Systems automation sector
- Renewable energy sector
- Artificial intelligence sector
- Robotics sector
- Automotive industry sector
- Electricity production and distribution company;
- Telecommunications sector (telephone operators);
- Space telecommunications sector

On a regional level

- Design offices
- Education and training
- Research and development

Skills specific to the pathway:

- Understanding the operation and implementation of semiconductor devices
- Master the complex functions of electronic systems
- Take into account the reliability of an electronic system
- Managing energy in an on-board system
- Master microwave and optoelectronic circuits and characterisation techniques
- Understand sensor technology and design their processing chain

Depending on the specialisation block:

- Design digital circuits on silicon or FPGA
- Design digital systems on a chip (SoC)
- Mastering frequency synthesis systems
- Design antennas and microwave integrated circuits
- Computer-aided design of microsystems and components

- Master the design stages of microelectronic components

E. Gateways to other specialisations

There are links between this Master's degree and the 'Industrial Electrical Engineering' and 'Telecommunications' Master's degrees. This Master offers multidisciplinary and cross-disciplinary teaching programmes.

F. Training monitoring indicators

The aim of the system is to diversify assessment methods in order to evaluate student skills as widely as possible. In this context, we will assess:
(1) student empowerment ;
(2) regular monitoring of knowledge acquisition;
(3) acquisition of oral expression ;
(4) the student's ability to work as part of a team and to synthesise information;
(5) monitoring students' abilities and not just their knowledge.
The breakdown between the different forms of assessment is as follows:
- Assessment of knowledge: 40
- Oral expression: 20
- Personal work: 20
- Ability to analyse and summarise: 20

II – Half-yearly teaching organization sheets of the specialty

Semester 1

Teaching unit	Materials	Credits	Coefficient	Weekly hourly volume			Half-yearly Hourly Volume (15 weeks)	Additional Work in Consultation (15 weeks)	Assessment method	
	Titled			Course	TD	TP			Continuous Assessment	Exam
Fundamental EU Code: UEF 1.1.1 Credits: 10 Coefficients: 5	Microcontroller systems	6	3	3:00 a.m.	1h30		67h30	82h30	40%	60%
	Advanced Digital Electronics: FPGA and VHDL	4	2	1h30	1h30		45h00	55h00	40%	60%
Fundamental EU Code: UEF 1.1.2 Credits: 8 Coefficients: 4	Advanced signal processing	4	2	1h30	1h30		45h00	55h00	40%	60%
	Digital servo systems	4	2	1h30	1h30		45h00	55h00	40%	60%
Methodological EU Code: UEM 1.1 Credits: 9 Coefficients: 5	Microcontroller Systems TP	2	1			1h30	10:30 p.m.	27:30	100%	
	FPGA and VHDL practical work	2	1			1h30	10:30 p.m.	27:30	100%	
	Advanced Signal Processing / Digital Servo Systems TP	2	1			1h30	10:30 p.m.	27:30	100%	
	Embedded C++ Programming	3	2	1h30		1 hour	45h00	37h50	40%	60%
EU Discovery Code: UED 1.1 Credits: 2 Coefficients: 2	Subject of choice 1	1	1	1h30			10:30 p.m.	2:30 a.m.		100%
	Subject of choice 2	1	1	1h30			10:30 p.m.	2:30 a.m.		100%
Transversal EU Code: UET 1.1 Credits: 1 Coefficients: 1	Technical English and Terminology	1	1	1h30			10:30 p.m.	2:30 a.m.		100%
Total semester 1		30	17	1:30 p.m.	6:00 a.m.	5:30 a.m.	375 hours	375 hours		

Semester 2

Teaching unit	Materials	Credits	Coefficient	Weekly hourly volume			Half-yearly Hourly Volume (15 weeks)	Additional Work in Consultation (15 weeks)	Assessment method	
	Titled			Course	TD	TP			Continuous Assessment	Exam
Fundamental EU Code: UEF 1.2.1 Credits: 8 Coefficients: 4	Signal Processors (DSP)	4	2	1h30	1h30		45h00	67h30	40%	60%
	Embedded Artificial Intelligence	4	2	1h30	1h30		45h00	55h00	40%	60%
Fundamental EU Code: UEF 1.2.2 Credits: 10 Coefficients: 5	Embedded Processor Architecture	6	3	3:00 a.m.	1h30		67h30	55h00	40%	60%
	Industrial Programmable Controllers.	4	2	1h30	1h30		45h00	55h00	40%	60%
Methodological EU Code: UEM 1.2 Credits: 9 Coefficients: 5	Digital Signal Processors TP	2	1			1h30	10:30 p.m.	27:30	100%	
	TP Architecture of Processors for Embedded Systems	2	1			1h30	10:30 p.m.	27:30	100%	
	TP / Industrial Programmable Automation TP	2	1			1h30	10:30 p.m.	27:30	100%	
	Python/Java Programming for Embedded Systems	3	2	1h30		1 hour	37h50	37h50	40%	60%
EU Discovery Code: UED 1.2 Credits: 2 Coefficients: 2	3rd choice of subject	1	1	1h30			10:30 p.m.	2:30 a.m.		100%
	Subject of choice 4	1	1	1h30			10:30 p.m.	2:30 a.m.		100%
Transversal EU Code: UET 1.2 Credits: 1 Coefficients: 1	Compliance with standards and rules of ethics and integrity	1	1	1h30			10:30 p.m.	2:30 a.m.		100%
Total semester 2		30	17	1:30 p.m.	6:00 a.m.	5:30 a.m.	375 hours	375 hours		

Semester 3

Fundamental EU Code: UEF 2.1.1 Credits: 10 Coefficients: 5										
	Real-Time Systems	6	3	3:00 a.m.	1h30		67h30	82h00	40%	60%
Fundamental EU Code: UEF 2.1.2 Credits: 8 Coefficients: 4	Embedded systems	4	2	1h30	1h30		45h00	55h00	40%	60%
	Industrial networks and communications	4	2	1h30	1h30		45h00	55h00	40%	60%
Methodological EU Code: UEM 2.1 Credits: 9 Coefficients: 5	Embedded Systems Practical Work/ Real-Time Systems Practical Work	2	1			1h30	10:30 p.m.	27:30	100%	
	Industrial Networks TP	2	1			1h30	10:30 p.m.	27:30	100%	
	Artificial Vision System TP	2	1			1h30	10:30 p.m.	27:30	100%	
	Study and Implementation of Projects	3	2	1h30		1 hour	37h50	37h50	40%	60%
EU Discovery Code: UED 2.1 Credits: 2 Coefficients: 2	5th choice of subject	1	1	1h30			10:30 p.m.	2:30 a.m.		100%
	6th choice of subject	1	1	1h30			10:30 p.m.	2:30 a.m.		100%
Transversal EU Code: UET 2.1 Credits: 1 Coefficients: 1	Documentary research and dissertation design	1	1	1h30			10:30 p.m.	2:30 a.m.		100%
Total semester 3		30	17	1:30 p.m.	6:00 a.m.	5:30 a.m.	375 hours	375 hours		

General guidelines on the choice of cross-curricular and discovery subjects:

Six (discovery) subjects in the Master's Subject Reference "Embedded Systems Electronics" (Table above) are left to the free choice of establishments which can choose their subjects indifferently from the list presented below according to their priorities.

Subjects with detailed programs:

- RFID Radio Identification (Discovery)
- Home Automation (Discovery)
- Embedded Systems for Automotive (Discovery)
- Embedded Systems Operating Systems (Discovery)
- Smart cards (Discovery)
- Mobile Robotics (Discovery)
- Wireless Communications (Discovery)
- Robotics (Discovery)
- Renewable energies: photovoltaic solar energy (Discovery)
- Autonomous Energy Systems (Discovery)

Other subjects left to the free choice of establishments (programs open after validation of the CPND)

- Electromagnetic Compatibility (Transverse)
- Industrial actuators (Discovery)
- Zigbee Sensor Networks (Discovery)
- Information Coding and Security (Transversal)
- Technological innovations in mechatronics (Discovery)
- Processors dedicated to embedded systems (Discovery)
- Embedded Systems for Telecommunications (Discovery)
- Micro and nano embedded systems (Discovery)
- Verification and validation of embedded systems (Discovery)
- Real-Time Operating Systems (Discovery)
- Embedded systems for industry (Discovery)
- Embedded systems for avionics (Discovery)
- Study of mixed signals on embedded systems (Discovery)
- Linux system for embedded systems (Discovery)
- MOCN (Numerically Controlled Machine Tool) (Discovery)
- Web technologies: HTML, PHP, My SQL, XML
- Java 1: UML and JAVA design (basic concepts of object-oriented)
- Java 2: Real-time Java
- Java 3: Java on Android
- Linux system for embedded systems
- Connected objects: Concept, Client, Server
- Control of the movement of mobile platforms;
- GUI with Tkinter & Raspberry PI 4;
- " Power supply design for embedded systems";
- " Control of electric actuators";
- Database dedicated to embedded systems;
- Cyber-Physical Systems and Internet of Things;

- The Internet of Things;
- Implementation of finite state machines using Stateflow in the Matlab environment;
- Simulation of applications in the Proteus environment;
- Using C-Vivado language for FPGA circuits;
- The Internet of Things.
- Advanced artificial intelligence (Deep Learning, etc.).

Semester 4

An internship at a company, leading to the completion and defence of a dissertation.

	VHS	Coefficient	Credits
Personal Work	550	09	18
Internship in a company	100	04	06
Seminars	50	02	03
Other (Supervision)	50	02	03
Total Semester 4	750	17	30

This table is given for information purposes only.

Evaluation of the End of Master's Cycle Project

- Scientific value (Jury assessment) /6
- Writing the Dissertation (Jury Assessment) /4
- Presentation and answer to questions (Jury assessment)/4
- Supervisor's assessment /3
- Presentation of the internship report (Jury assessment)/3

III - Detailed program by subject for semester S1

Semester: 1
Teaching unit: UEF 1.1.1
Subject 1: Microcontroller Systems
VHS: 67h30 (Lecture: 3h00, Tutorial: 1h30)
Credits: 6
Coefficient: 3

Objectives to be achieved:

This subject allows students to familiarize themselves with the different functionalities offered by microcontrollers in the real world. Indeed, the world is moving towards the aspect of artificial intelligence which requires designs involving microcontrollers as key parts to achieve industrial-scale applications recommended in particular in the embedded system. Thus, students will learn to handle the different input/output devices, in this case: all-or-nothing inputs, sensor modules, supervision screens, motors and actuators, as well as the establishment of an appropriate communication layer, both simple ones such as those using standardized protocols, UART, I²C, SPI, Bluetooth, and complex protocols, namely: Modbus TCP/IP, MQTT, ESP-NOW, Zigbee. The basic learning is done with the famous "ATmega328p" microcontroller given the range of applications that can be offered to the user and which goes hand in hand with the Arduino platforms. On the other hand, it has better performances than those offered by the old microcontrollers such as the "PIC-16Fxx" family, moreover, it lends itself well to accepting the different types of compilers, in particular the C-Embedded in the user-friendly environment called "IDE". That being said, it allows students to evolve more easily in a fun way towards the new generations of microcontrollers such as for example the ESP32 or the ARM32, thus requiring a very substantial background in advanced programming.

Additionally, Proteus® software allows for interactive in-circuit simulation of these "AVR" microcontrollers, and explicit testing of the code and circuit diagram before thinking about building the target hardware. When the application is working properly, as it should, in simulation mode, the PCB can be designed to validate the design in question.

The implementation of the content of this subject will be carried out within the unit:
"UEM1.1: Microcontroller Systems Practical Work" .

Recommended prerequisites:

L3 Microprocessor Systems, Digital Electronics, Finite State Machine, Basic Electronics, Assembler Language, C Language.

Content of the subject:

Chapter 1 Basic Introduction to Microcontrollers (5 weeks)

- General architecture of the AVR32/PIC32 microcontroller.
- Setting a bit, clearing a bit, checking a bit, Toggling a bit and Macros.
- AVR memory architecture (Flash, Boot section; Data memory, RAM, General Registers; EEPROM).
- C-Embedded in the IDE environment
- The general structure of an Arduino program (Header, declarative section, configuration and initialization, Main program and definition of subroutines)
- Basic instructions (Typedef, enum, If-Else, Switch-Case, Struct, Union)
- Applications (Drop-down menu with three buttons Up, Down, OK).

- Digital output management (LED flashing using delay() and millis())
- Data display (TM1637 4-digit 7-segment, 2x16 LCD)

Chapter 2. Advanced Features of Microcontrollers (5 weeks)

- Interruptions caused by external events (Rising, falling, change).
- Analog to digital conversion by programmed interrupt.
- Timer interrupts: Timer0, Timer1 and Timer2.
- PWM generator by interruption and timer management (CTC mode and PWM generator).
- Examples: incremental encoder dedicated to managing a drop-down menu
- AC power dimmer with a solid state relay (SSR)
- Example: Controlling a servo motor with OC1A/OC1B mode for different angular resolutions.
- Example: DC motor speed control with the L298N module.
- Use of internal EEPROM (Read, Write, Erase).
- Half- and full-duplex interrupt mode communication protocols: UART (manipulation of string, byte, int, double data types)
- I²C bus implementation and usage (DH22, DS18b20, ds3231 temperature sensor)
- The SPI protocol
- Full-duplex communication
- Single master – multiple slave architecture
- High-speed communication [Hz, kHz or MHz]
- Master-slave configurations.
- Flexible protocol.
- Short-distance communication.
- Initializing an SD card and storing data.

Chapter 3. High-Level Structured Programming (5 weeks)

- Multi-tasking systems (function pointers, implementation of a finite state machine "FSM", introductory example of the management of a two-way traffic light)
- Communication with Bluetooth (HC-05) using “MIT APP-Inventor” on Android and Nano platform with Atmega328P microcontroller).
- The CAN bus and its applications (design of a network, example dashboard of a vehicle).
- Modbus TCP/IP fieldbus example S7-1200 station with UNO equipped with Ethernet Shield module.

Assessment method:

Exam: 60% , Continuous Assessment: 40%

Bibliographic references:

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

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

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

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

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

[6] INTRENET OF THINGS WITH ARDUINO AND BOLD IOT: With Arduino and Bolt;By Ashwin Pajankar; Published by BPB Publications 2018.

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

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

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

Semester: 1
Teaching unit: UEF 1.1.1
Subject 2: Advanced Digital Electronics: FPGA and VHDL
VHS: 45h00 (Lecture: 1h30, Tutorial: 1h30)
Credits: 4
Coefficient: 2

Teaching objectives:

In this subject, students will have to study the different types of programmable circuits, as well as the different design methods, in particular programming using hardware description languages.
The application of the content of this subject is carried out at the level of the subject " UEM1.1: FPGA & VHDL practical work ".

Recommended prior knowledge:

Digital electronics (combinatorial and sequential).

Content of the material:

Chapter 1. Basics of Programmable Circuits (1 week)

- General architecture of programmable logic circuits: PAL, GAL, PLD, CPLD
- Examples of manufacturers and programming tools: Altera Quartus II, Xilinx ISE

Chapter 2. Programming in VHDL (5 weeks)

- History of VHDL.
- Comparison between VHDL and programming languages.
- Different descriptions of an architecture: data flow, behavioral, structural.
- Identifiers and case sensitivity.
- Comments.
- Representing numbers in VHDL
- General structure of a VHDL code: Library, Entity, Ports, Architecture.
- Data types: predefined, user-defined
- Operators: logical, relational, shift, concatenation
- Signal attributes: EVENT, ...
- Signal, variable and constant
- Process
- Component
- IF-THEN-ELSE statement
- CASE-WHEN statement
- WHEN-ELSE instruction
- WITH-SELECT-WHEN statement

Chapter 3. Applications on FPGA circuits (5 weeks)

- Multiplexer
- D-Flip
- Adder
- Universal counter with actions: activation, reset, load.
- Frequency divider.
- Frequency management with buttons: selection, division
- 7-segment decoder,
- Serial display on several 7 segments.

- 8-bit arithmetic logic unit
- 8-bit comparator

Chapter 4. Advanced Design with Finite State Machines (FSM) (4 weeks)

- Introduction: Mealy and Moore structure
- Representation of an FSM machine
- FSM Design Examples

Assessment method:

Exam: 60% , Continuous Assessment: 40%

Bibliographic references:

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

[2] . Volnei A. Pedroni, "Circuit Design and Simulation with VHDL", 2nd edition , 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, "The VHDL language: from language to circuit, from circuit to language", Dunod, 2007.

[6] . Christian Tavernier, "Programmable Logic Circuits", Dunod 1992.

Semester: 1
Teaching unit: UEF 1.1.2
Subject 3: Advanced Signal Processing
VHS: 45h00 (Lecture: 1h30, Tutorial: 1h30)
Credits: 4
Coefficient: 2

Teaching objectives:

The student receives the basic concepts that allow him to understand and apply signal processing methods concerning random signals and digital filters.

Recommended prior knowledge:

Knowledge of digital deterministic signal processing and probabilities is required to pursue this course. This knowledge is provided at the third-year Electronics degree level.

Content of the subject:

Chapter 1: Reminders on digital filters (RIF and RII) (3 weeks)

- Z-transformed
- Structures, transfer functions, stability and implementation of digital filters (RIF and RII)
- Digital minimum phase filter
- Methods of synthesizing RIF filters and RII filters
- Multi-rate digital filters

Chapter 2: Random Signals and Stochastic Processes (4 Weeks)

- Reminder on random processes
- Stationarity
- Power spectral density
- Adapted filter, Wiener filter
- Periodogram, correlogram, averaged periodogram, smoothed periodogram
- Notions of stochastic processes
- Stationarities in the broad and strict sense and Ergodicity
- Examples of stochastic processes (Poisson process, Gaussian process and Markovian process)
- Higher order statistics (Moments and cumulants, Polyspectra, non-Gaussian processes, non-linear processing)
- Introduction to particle filtering

Chapter 3: Parametric Spectral Analysis and Adaptive Digital Filtering (4 weeks)

- Parametric methods
- AR model (Lévinson, Yulewalker, Burg, Pisarenko, Music...)
- ARMA model
- LMS Stochastic Gradient Algorithm
- Recursive Least Squares Algorithm RLS

Chapter 4: Time-frequency and time-scale analysis (4 weeks)

- Time-frequency duality
- Short-term Fourier transform
- Continuous, discrete and dyadic wavelets
- Multi-resolution analysis and wavelet bases
- Wigner-Ville transform
- Time-Scale Analysis.

Assessment method:

Continuous assessment: 40%; Exam: 60%.

Bibliographic references:

[1] .Mori Yvon, "Random Signals and Stochastic Processes", Lavoisier, 2014.

[2] .N. Hermann, "Engineering Probabilities: Random Variables and Birch Simulations", 2002.

[3] .M. Kunt, "Digital Signal Processing", Dunod, Paris, 1981.

[4] .J. M Brossier, "Signal and Digital Communications, Signal Processing Collection", Hermès, Paris, 1997.

[5] .M. Bellanger, "Digital Signal Processing: Theory and Practice", 8th edition, Dunod, 2006.

Semester: 1
Teaching unit: UEF 1.1.2
Subject 4: Digital Servo Systems
VHS: 45h00 (Lecture: 1h30, Tutorial: 1h30)
Credits: 4
Coefficient: 2

Teaching objectives:

Introduce the properties and representations of discrete-time linear dynamic systems. Give the fundamental elements of the control of linear systems represented in the form of a *Z transfer function* . Present the different methods of synthesis of discrete-time correctors.

Recommended prior knowledge:

Time and frequency analysis of continuous servo systems, Graphical and state representations, Corrector synthesis.

Content of the material:

Chapter 1: Study of signal sampling (5 Weeks)

Z-transform and modified Z-transform: Shannon's theorem , zeroth-order and first-order blockers, properties of the Z-transform, Overview of the modified Z-transform and its properties,... Initial value and final value theorem of a sampled system

Sampled transfers and recurrent equations: Discretization of a continuous transfer, Representation of discrete systems by recurrence equations, Properties, etc.

Overview of the bilinear transformation of a sampled transfer: Relationship between the control of continuous systems and the control of sampled systems (study of the stability of a sampled system by the Routh criterion, etc.).

Chapter 2: Analysis of Sampled Systems in State Space (5 Weeks)

Discretization of the equation of state of a continuous system: Relationship between the equation of state of a continuous system and that of a discrete system.

Representation and resolution of the state equation of a discrete system: Different forms of the evolution matrix (diagonal, companion, observer, controller, observability and controllability).

Stability and precision of a discrete system: Roots of the characteristic equation, controllable modes, observable modes from the state representation of the sampled systems, Responses of a sampled system, Stability examination by the Jurry criterion, etc.

Notions of governability and observability for SISO and MIMO systems.

Chapter 3. Synthesis of a Controller (5 Weeks)

Pole placement by state feedback and output feedback: synthesis of simple control laws

State and output estimator: Case of inaccessible system states

Other synthesis methods: digital PID controller (1 degree of freedom structure), RST controller (2 degree of freedom structure).

Assessment method:

Continuous assessment: 40%; Exam: 60%.

Bibliographic references:

[1] .L. Maret, Automatic Regulation, 1987.

[2] .Dorf & Bishop, Modern Control Systems, Addison-Wesley, 1995

[3] .J. L Abatut, Sampled Linear Systems and Servo Control, Dunod Edition

[4] .J. Ragot, M. Roesch, Exercises and Problems in Automation, Masson Edition.

[5] .J. Mainguenaud, Automatics Course Volume 3, Masson Edition.

[6] .T. J. Katsuhiko, Modern Control Engineering, 5th Edition, Prentice Hall.

[7] .H. Buhler, Sampled Settings Volume 1, Dunod Edition.

[8] .Mr. Rivoire, Automation Course Volume 2, Chihab Edition.

[9] . *Th. Kailath, Linear Systems, Prentice-Hall, 1980.*

Semester: 1
Teaching unit: UEM 1.1
Subject 1: Practical work on microcontroller systems
VHS: 10:30 p.m. (TP: 1:30 p.m.)
Credits: 2
Coefficient: 1

Goals :

This subject is a complement intended particularly to support and properly understand the aforementioned subject "UEF1.1.1: Microcontroller Systems". Furthermore, it offers students the opportunity to manipulate in a tangible way the knowledge acquired through well-targeted examples. In addition, it is preferable that the student can take advantage of designs based on the ATmega328P microcontroller to create reliable applications. The main objective can be to allow students to create their own applications according to the requirements imposed by the design itself.

Recommended prerequisites:

Microprocessor systems, digital electronics, FPGA, basic electronics, embedded C, Grafcet.

Main content:

TP 1: Tutorial and familiarization with the IDE and Proteus environment (Basic instructions: If-Else, Switch-case, delay(), millis(), typedef, enum, struct, union).

TP 2: External interruptions with two push buttons (controlling four LEDs with shift registers in both directions).

TP 3: Implementation of a finite state machine using a simple push button to control a DC motor in PWM with an H-bridge from an Analog module.

TP 4: Design and construction of a power dimmer (SSR static relay) via the Timer1 interrupt and the external interrupt to detect the zero crossing signal.

TP 5: Management of a drop-down menu with three push buttons and 2x16 LCD display.

TP 6: Control of a stepper motor via Timer1 in CTC mode " *Clear* Timer on *Compare*" .

TP 7: ADC interfacing with operational amplifiers and analog sensors (Manipulation of internal registers).

TP 8: Multitasking management using a function pointer and the timer1 interrupt to manage the scrolling time of a group of LEDs.

TP 9: Reading/writing data from an internal EEPROM and an external EEPROM of type 24lc256 using the I^2C bus.

TP 10: Control of a DC type motor via the Bluetooth module " HC-05" using the MIT App Inventor " Android" application .

TP 11: Implementation of a PID controller dedicated to temperature regulation using the DS 18b20 sensor and a previously mentioned SSR static relay.

TP 12: Data exchange between Arduino (Slave) UNO and ESP32 (Master) type platforms using the UART communication protocol.

TP 13: Data exchange between Arduino (Clients) UNO and Raspberry pi 4 (Broker) type platforms using the MQTT protocol.

TP 14: Data exchange between Arduino and an S7-1200 type PLC via the Modbus TCP/IP protocol to control a Micro master 420 type speed controller.

Assessment method:

Continuous Assessment: 100%

Bibliographic references:

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

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

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

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

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

[6] .Published by 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] .By Ashwin Pajankar; Published by BPB Publications 2018.

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

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

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

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

Semester: 1
Teaching unit: UEM 1.1
Subject 2: FPGA and VHDL practical work
VHS: 10:30 p.m. (TP: 1:30 p.m.)
Credits: 2
Coefficient: 1

Teaching objectives:

This subject is built around the subject " UEF1.1.1: Advanced digital electronics: FPGA and VHDL "; it allows students to apply the knowledge acquired in the form of specific examples.

This subject allows the student to design an electronic system using the VHDL description language and to test each design on an FPGA board.

Recommended prior knowledge:

Combinatorial and sequential logic.

Content of the material:

TP1: Presentation of the development and simulation tool: Altera Quartus II or Xilinx ISE .

TP2: Use of the development board through a given example of an adder.

TP3: First circuit examples: multiplexer, D flip-flop.

TP4: Simple 48-bit decimal counter.

TP5: 48-bit decimal counter with actions: activation, reset, load.

TP6: Traffic light.

TP7: Multiplier/divider with shift registers.

TP8: Serial display on several 7 segments.

TP9: Digital clock.

TP10: Frequency division.

TP11: Frequency division controllable with buttons.

TP12: VGA display.

Assessment method:

Continuous Assessment: 100%

Bibliographic references:

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

[2] .Volnei A. Pedroni, "Circuit Design and Simulation with VHDL", 2nd edition, 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, "The VHDL language: from language to circuit, from circuit to language", Dunod, 2007.

[6] . *Christian Tavernier, "Programmable Logic Circuits", Dunod 1992.*

Semester: 1
Teaching unit: UEM1.1
Subject 3: Advanced Signal Processing/ Digital Servo Systems
VHS: 10:30 p.m. (TP: 1:30 p.m.)
Credits: 2
Coefficient: 1

Teaching objectives:

Practical work carried out using MATLAB to give a practical aspect to complex theoretical concepts.

Recommended prior knowledge:

Mathematics (Theory and calculus of probabilities, Complex analysis) - Deterministic signal theory, Probabilities and statistics.

Content of the subject :

Advanced Signal Processing TP
TP1 : Synthesis and application of a low-pass RIF filter using the window method (Hanning, Hamming, Bessel and/or Blackman)
TP2 : Synthesis and application of a low-pass IIR filter by bilinear transformation
TP3 : Parametric spectral analysis AR and/or ARMA of sound signals (example of non-signals stationary)
TP4 : Elimination of 50Hz interference by the LMS gradient algorithm
TP5 : Denoising a signal using the discrete wavelet transform DWT.

Digital servo systems TP

Assessment method:

Continuous assessment: 100%.

Bibliographic references:

[1] .Mori Yvon, "Random Signals and Stochastic Processes", Lavoisier , 2014.
[2] . 2. N. Hermann, "Engineering Probabilities: Random Variables and Birch Simulations", 2002.
[3] . 3. M. Kunt, "Digital Signal Processing", Dunod, Paris, 1981.
[4] . 4. M. Bellanger, "Digital Signal Processing: Theory and Practice", 8th edition, Dunod, 2006

Semester: 1
Teaching unit: UEM 1.1
Subject 4: Embedded C++ Programming
VHS: 37h30 (Lecture: 1h30, Practical work: 1h00)
Credits: 3
Coefficient: 2

Teaching objectives:

C++ is one of the most widely used languages in the industry, thanks to the power of object-oriented language , its versatility, and its execution performance.
This subject aims to introduce the language with a focus on its specific use in embedded systems. Students will be able to assimilate object concepts, discover and implement the C++ language in projects, and acquire a beginning of autonomy in this language.

Recommended prior knowledge:

Digital systems, C programming.

Content of the material:

Chapter 1. The C++ language - introduction (1 week)
History, comparison with other languages, use of this language in industry, standard and version of C++ (C++98 • C++03 • C++11 • C++14 • C++17 • C++20), new features of the language
Chapter 2. Compiler (1 week)
- Open source GCC/G++ compiler
- x86/ARM cross compiler
- Dynamic and static linkage
- Debugging
- Makefile

Chapter 3. Types, Constants, Variables (1 week)
References and Pointers, declaration, scope, initialization, array: declaration, initialization, namespace, dynamic allocation
Chapter 4. Concept of object (2 weeks)
- From C to C++, Classes and Objects
- Protection, Access
- Instance Variable, Constructor, Destructor
- Overload
- Operator “This”
- UML/SysML Object and Modeling
- Automatic code generation

Chapter 5. Derived Classes (1 week)
- Inheritance and Instantiation
- Friends
- Virtual classroom
- Multiple inheritance

Chapter 6. Operator Overloading (1 week)
- Functions Operator
- Overload
- Friend function, Friend class

Chapter 7. Flow Control (1 week)
Inputs, Outputs, Status, Overload, File Management
Chapter 8. Exceptions (1 week)
Definition of an exception, Interception

Chapter 9. Data Structure and STL (1 week)
Vector, Map, List, Stack, Standard Algo
Chapter 10. Particularities of embedded C++ (1 week)

- Std Bookstore
- Compilation and linking
- Specific coding rules
- Virtual classroom

Chapter 11. Multithreading (2 weeks)

- thread_local attribute
- Thread class, Mutex class
- Semaphore vs Mutex
- Conditions, Locks, Future and Promise
- Atomic Operator and Operation

Chapter 12. Smart Pointers (1 week)

- Deprecation of auto_ptr
- unique_ptr,
- shared_ptr
- weak_p

Chapter 13. Principles of templates (1 week)

- Concept of generic programming
- Model class
- Model functions

Assessment method:

Exam: 60% , Continuous Assessment: 40%

Bibliographic references:

[1] . Michael Barr, Anthony Massa, Programming embedded systems: in C and C++, 2nd 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

Semester: 1
Teaching unit: UED 1.1
Subject 1: optional
VHS: 10:30 p.m. (lesson: 1.5 hours)
Credits: 1
Coefficient: 1

Semester: 1
Teaching unit: UED 1.1
Subject 2: optional
VHS: 10:30 p.m. (lesson: 1.5 hours)
Credits: 1
Coefficient: 1

Semester: 1
Teaching unit: UET 1.1
Subject: Technical English and Terminology
VHS: 10:30 p.m. (lesson: 1.5 hours)
Credits: 1
Coefficient: 1

Teaching objectives:

Introduce students to technical vocabulary. Strengthen their language skills. Help them understand and summarize a technical document. Enable them to understand a conversation in English held in a scientific setting.

Recommended prior knowledge:

Basic English Vocabulary and Grammar

Content of the material:

- Written comprehension: Reading and analysis of texts relating to the specialty.

- Oral comprehension: Based on authentic popular science video documents, note-taking, summary and presentation of the document.

- Oral expression: Presentation of a scientific or technical subject, development and exchange of oral messages (ideas and data), Telephone communication, Gestural expression.

- Written expression: Extracting ideas from a scientific document, Writing a scientific message, Exchanging information in writing, writing CVs, letters of application for internships or jobs.

Recommendation: It is strongly recommended that the subject manager present and explain at the end of each session (at most) around ten technical words of the specialty in the three languages (if possible) English, French and Arabic.

Assessment method:

Review: 100%.

Bibliographic references:

[1] . PT Danison, Practical guide to writing in English: usages and rules, practical advice, Editions d'Organisation 2007
[2] . A. Chamberlain, R. Steele, Practical Guide to Communication: English, Didier 1992
[3] . R. Ernst, Dictionary of applied techniques and sciences: French-English, Dunod 2002.
[4] . J. Comfort, S. Hick, and A. Savage, Basic Technical English, Oxford University Press, 1980
[5] . EH Glendinning and N. Glendinning, Oxford English for Electrical and Mechanical Engineering, Oxford University Press 1995
[6] . TN Huckin, and AL Olsen, Technical writing and professional communication for nonnative speakers of English, McGraw-Hill 1991
[7] . J. Orasanu, Reading Comprehension from Research to Practice, Erlbaum Associates 1986

IV - Detailed program by subject for semester S2

Semester: 2
Teaching unit: UEF 1.2.1
Subject 1: Digital Signal Processors (DSP)
VHS: 45h00 (Lecture: 1h30, Tutorial: 1h30)
Credits: 4
Coefficient: 2

Teaching objectives:

Understand the internal architecture of a DSP and the hardware platform integrating this DSP as well as the development environment on a DSP-based platform. At the end of this subject, the student must master the design flow and must also be able to make an algorithm-architecture match for the implementation of algorithms on a DSP-based platform.

Recommended prior knowledge:

Microprocessor systems. Advanced digital signal processing. Assembly language and C programming.

Content of the material:

Chapter 1: General information on DSP processors (1 week)
Definitions, presentation of the different DSP families, classification of DSPs, areas of application of DSPs, main algorithms processed, DSP processors and other approaches, history and recent developments

Chapter 2: Fixed-Point and Floating-Point Arithmetic (2 weeks)
Digitization of signals, sampling of an analog signal, uniform quantization (characteristic, error characteristic, dynamics), non-uniform quantization, logarithmic quantization (compression expansion law, segment approximations of the compression laws A and μ), number representation formats, coding of integers (positive or unsigned integers, 1's complement, 2's complement), representation of real numbers in a calculator (fixed point, floating point)

Chapter 3: TMS320C6x DSP Architecture (4 weeks)
Internal architecture of the C6000, the processor, memory mapping, functional units, execution and *fetch packets* , pipeline architecture, registers, control registers, peripherals (timers, PLLs, interrupts, HPI, GPIO), serial link (*multichannel buffered serial port*), presentation of the instruction set

Chapter 4: Memory Management (2 weeks)
Presentation and interest of the Harvard architecture. Internal memories (levels L1 and L2). External memories (SRAM, Flash, DDRAM, ...) Memory addressing plan. *.cmd file (organization of sections). Management of external memory by EMIF (*External Memory InterFace*). Addressing modes (indirect, circular). Block transfer technique. Data organization for EDMA. Parameters and options for EDMA. Example of data transfer.

Chapter 5: Development Environment: 'Code Composer Studio' (CCS) (2 weeks)
Introduction, Basic Setup, creating a new project under CCS, running the program (*Break Point, Watch Window, Plots* , Images, data logging), CCS GEL (*General Extension Language) scripts, using DIP switches* and LEDs

Chapter 6: Signal Processing Algorithms on DSP (4 weeks)
Algorithm-architecture matching. FIR and IIR filtering. Shift and circular buffers, quantization problems, real-time constraints, input/output management. FFT implementation on DSP (inverted addressing).
Assessment method:

Continuous assessment: 40%; Exam: 60%.

Bibliographic references :

[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] . TB Welch, CHG Wright and MG 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] . SW Smith, The Scientist and Engineer's Guide to Digital Signal Processing.

[9] . G. Baudoin and F. Virolleau, DSPs: TMS320C54x family. Application development.

[10] . L. Correvon, DSP and Real Time: Industrial Application, University of Engineering of the Canton of Neuchâtel.

[11] . P. Laspsley, J. Bier, A. Shoham, EA 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, DSPs, ADSP 218X family: Principles and applications, Dunod, 2000.

[15] . B. Bouchez, Digital audio applications of DSP: theory and practice of digital sound processing, 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. July 1991.

Semester: 02
Teaching unit: UEF 1.2.1
Subject 2: Embedded artificial intelligence
VHS: 45h00 (Lecture: 1h30, Tutorial: 1h30)
Credits: 4
Coefficient: 2

Teaching objectives:

To enable students to become familiar with artificial intelligence (AI) techniques and its applications in embedded systems.

Recommended prior knowledge:

This subject requires prior knowledge of advanced numerical analysis methods, C, MatLab or Python programming.

Content of the material:

Chapter 1. Introduction to Artificial Intelligence and Embedded Systems (2 weeks)

Artificial Intelligence (AI); Application Areas; Main Techniques Related to Artificial Intelligence; Advantages and Disadvantages of AI; Embedded Systems; Embedded Systems and the Prospects of AI; Challenges of Implementing Artificial Intelligence in Embedded Design.

Chapter 2. Machine and Deep Learning Methods (4 weeks)

Supervised learning; Unsupervised learning; Semi-supervised learning; Reinforcement learning; Ensemble methods in machine learning (bagging, boosting and stacking), Neural networks (MLP, RBF, RNN, BNNs, etc.); Convolutional neural networks (CNN), deep neural networks (DCNN, VGG-16, ResNet, LSTM, GRU, etc.).

Chapter 3. Machine and Deep Learning Application (03 weeks)

Simple examples of application of machine learning algorithms (Naive-Bayes, Decsion Tree, Random forest, k-NN, K-Means, svm, PCA, Q-Learning, etc.) and deep learning in problems of regression, classification, control, data partitioning and dimension reduction.

Applications : signal processing (audio, etc.), imaging (object detection, shape recognition, segmentation, etc.), natural language, text translation, detection, diagnosis, control, etc.

Chapter 4 Implementation of machine and deep learning in embedded systems (03 weeks)

Basic concepts of embedded machine learning; Pruning; Quantization and mixed precision; Embedded architectures for machine and deep learning; Reconfigurable approaches (FPGAs), CPU approaches, and GPUs; Microcontroller approaches; Accelerator-oriented approaches; Efficient implementation of MAC (multiply–accumulate) units; Software and hardware optimizations; Main levels of abstractions; TinyML; Embedded boards Coral, Jetson Nano, Raspberry Pi, ...

Assessment method:

Continuous assessment: 40%; Exam: 60%.

Graphic biblio-webo references :

[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, YH, 2017. Python Machine Learning By Example. Packt Publishing Ltd.

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

[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, DD, 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, SA, Shelake, VG, Kamat, RK and Naik, GM, 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.*

Semester: 2
Teaching unit: UEF 1.2.2
Subject 1: Architecture of Embedded Processors
VHS: 67h30 (Lecture: 3h00, Tutorial: 1h30)
Credits: 6
Coefficient: 3

Teaching objectives:

This course allows students to learn in less time how to exploit the ARM processor architecture widely used in embedded systems. The first part shows how to configure and work with the STM32 platform development environment. The second part presents the basics of programming and the main aspects of the official HAL (Hardware Abstraction Layer). Towards the end, a more advanced part covers aspects such as the use of a real-time operating system, advanced applications like IoT and USB.The application of the content of this subject is carried out at the level of the subject " **UEM1.2: TP Architecture of microprocessors for Embedded** ".

Recommended prior knowledge:

Microprocessor systems, Microcontroller systems, Embedded C/C++ programming, Digital electronics: FPGA & VHDL programming.

Content of the material:

Chapter 1. Introduction to ARM-Cortex Processors (4 weeks)

- The different types of ARM-Cortex processors
- ARM Cortex processor architecture
- The registers
- Memory
- The pipeline
- Interrupts and exceptions
- The instruction set
- The performances
- Introduction to STM32 families.
- SM32CubeIDE development environment .
- Overview of the Nucléo development board.

Chapter 3. Exploiting ARM-Cortex Processors (8 weeks)

- Configuring input/output ports
- Interrupt Management
- UART communication
- DMA Management
- Clock management
- Using Timers
- Analog to digital conversion
- Digital to analog conversion
- I²C communication
- SPI Communication
- Calculation of CRC (Cyclic Redundancy Check)
- Using the Watchdog Timer
- The real-time clock

Chapter 4. Advanced Applications of ARM-Cortex Processors (2 weeks)

- FreeRTOS
- IoT Development
- USB Development

Assessment method:

Exam: **60%** , Continuous Assessment: **40%**

Bibliographic references:

[1] . Carmine Noviello, "Mastering STM32: A step-by-step guide to the most complete ARM Cortex-M platform, using the official STM32Cube development environment", 2nd edition, 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", 3rd edition, 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", 2nd edition, Elsevier, 2015.

[6] . Joseph Yiu, "The definitive guide to ARM Cortex-M3 and Cortex-M4 Processors", 3rd edition, Elsevier, 2014.

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

[8] . Joseph Yiu, "The definitive guide to ARM Cortex-M3," 2nd edition, Elsevier, 2010.

Semester: 2
Teaching unit: UEF1.2.2
Subject 2: Industrial Programmable Automation
VHS: 45 hours (Class: 1h30, tutorial: 1h30)
Credits: 4
Coefficient: 2

Teaching objectives:

This course allows the student to understand the hardware and software organization of APIs, to choose an API and the associated components according to the desired application and to use a programming language adapted to the API.

Recommended prior knowledge:

Combinational and sequential logic, Microprocessors, Microcontrollers, Sensors, Industrial networks and communications.

Content of the subject:

Chapter 1: API: Industrial Programmable Logic Controllers (2 weeks)
Definition of an API, Internal and external architecture of an API and characteristics. Choice of API. Types of Inputs/Outputs of an API and its characteristics.

Chapter 2: Materialization of industrial processes by APIs (3 weeks)
Definition of an automated system. The essential parts of an automated system (PO, PC, HMI, Interfacing). Operating principle of a PLC and an automated order-information system. Wiring. Sensor-actuator concepts, industrial networks, etc.

Chapter 3: API Programming (5 weeks)
Introduction to Grafcet. Introduction to languages: LD, IL, FBD, SFC, SCL. Application: definition of PO-PC parts, development of Grafcet, Ladder programming. Application exercises.

Chapter 4: Process Visualization (3 weeks)
Introduction to HMI (Human Machine Interface) and SCADA systems , process representation and control, alarm display, recipes, archiving, user management, etc. Application exercises.

Chapter 5: Programmable logic controller dedicated to security (2 weeks)
Architecture, process and machine control, management of safety functions.

Assessment method:

Continuous assessment: 40%; Exam: 60%.

Bibliographic references :

[1] . Frank D. Petruzella, Programmable Logic Controllers, 4th edition, Ed. McGraw Hill 2004.

[2] . William Bolton, Industrial Programmable Automation, Dunod Editions, I'Usine Nouvelle, 2010.

[3] . Ian G. Warnock, Programmable Controllers: Operation and Application, Prentice Hall.

[4] . Gilles Michel, Architecture and applications of industrial programmable automatons, Dunod.

[5] . G. Michel, Industrial Programmable Automation, Dunod, 1979.

[6] . S. Thelliez and J.M.Toullote, Grafcet and programmed industrial logic, Eyrolles, 1980.

[7] . JC Bossy, P. Brard, P. Faugère, C. Merlaud, Grafce: its practice and its applications, Educalivre Ed. Casteilla, 1995.

[8] . Henri Ney, Elements of automation, Electrotechnical and standardization collection, Edition Nathan, 1996.

[9] . M. Diaz, Petri Nets - Fundamental Models. IC2 Treatise - Computer Science and Information Systems Series, Hermès Science 2001

[10] . A. Choquet-Geniet, Petri Nets – A Modeling Tool, Dunod, 2006. Page | 50

[11] . P. Ladet, Modeling tools for sequential automation, Petri nets, Engineering techniques, 1990.

[12] . Siemens learning/training medium, Module 041-101 TIA Portal WinCC Basic with KTP700 Basic and SIMATIC S7-1200.

[13] . *IEC 61508-2 Functional safety of electrical/electronic/programmable electronic safety-related systems.*

Semester: 2
Teaching unit: UEM 1.2
Subject 1: Digital Signal Processors (DSP) Practical Work
VHS: 10:30 p.m. (TP: 1:30 p.m.)
Credits: 2
Coefficient: 1

Teaching objectives:

Put into practice the theoretical knowledge acquired during lectures and tutorials. Familiarize the student with application programs on a DSP platform through algorithm architecture matching. The programs can also be executed using the simulator provided with CCS.

Recommended prior knowledge:

Microprocessor systems. Advanced signal processing. Assembly language and C programming.

Content of the material:

Training teams are requested to complete at least 4 practical exercises (or more, if possible) depending on the type of DSP platform available. Below is a list of practical exercises that meet the objectives of the subject. In addition, it is permitted to add or replace some practical exercises from the attached list with other practical exercises related to the subject. Note : Any changes made to this list must be reported to the CPND so that other establishments can benefit from them.

TMS320C6x evaluation board
Integrated development environment "Code Composer Studio (CCS)", compiling, loading, running and debugging simple programs.
TP2: Acquisition, processing and restitution of audio signals with the DSK TMS320C6x
Sampling, spectrum folding, quantization, data transfer from/to Codec and use in polling mode or interrupt mode.
TP3: Signal generation with the DSK TMS320C6x
Wave generation, sine wave, AM modulation and FM modulation.
TP4: Implementation of digital filters by the DSK TMS320C6x
IIR and RIF filters.
TP: 5 Implementation of the Fourier Transform described and fast
TFD and TFR

TP5: Using Matlab with the DSK TMS320C6x
Simulations: Matlab or Simulink, automatic code generation for DSK using Simulink, Real Time Workshop and Code Composer Studio.

Assessment method:

Continuous assessment: 100%

Bibliographic references :

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

[2] . TB Welch, CHG Wright and MG 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.*

Semester: 2
Teaching unit: UEF 1.2
Subject 2: Practical work on Embedded Processor Architecture
VHS: 10:30 p.m. (TP: 1:30 p.m.)
Credits: 2
Coefficient: 1

Teaching objectives:

This subject is built around the subject " UEF1.2.2: Embedded Microprocessor Architecture "; it allows students to apply the knowledge acquired in the form of specific examples.

Recommended prior knowledge:

Microprocessor systems, combinational and sequential logic, fundamental electronics, power electronics.

Content of the material:

TP 1: Introduction to the STM Nucleo development board

TP 2: Getting started with the SM32CubeIDE environment.

TP 3: Implementation of a simple project (flashing of an LED) on the STM Nucleo card

TP 4: Development of a project with several inputs/outputs: push buttons, LEDs, etc.

TP 5: Developing a project with interruptions

TP 6: Configuration and use of Timer devices.

TP 7: Transmission and reception with USART serial communication

TP 8: Analog to digital conversion

TP 9: Variation of the PWM pulse width

TP 10: Demonstration of DMA operation

TP 11: Digital to analog conversion

Assessment method:

Continuous Assessment: 100%

Bibliographic references:

[1] . Carmine Noviello, "Mastering STM32: A step-by-step guide to the most complete ARM Cortex-M platform, using the official STM32Cube development environment", 2nd edition, 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", 3rd edition, 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", 2nd edition, Elsevier, 2015.

[6] . Joseph Yiu, "The definitive guide to ARM Cortex-M3 and Cortex-M4 Processors", 3rd edition, Elsevier, 2014.

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

[8] .*Joseph Yiu, "The definitive guide to ARM Cortex-M3," 2nd edition, Elsevier, 2010.*

Semester: 0 2

Teaching unit: UEM 1.2
Subject: Artificial Intelligence Practical Work
VHS: 10:30 p.m. (TP: 1:30 p.m.) – Once a fortnight
Credits: 2
Coefficient: 1

Teaching objectives:

To enable students to become familiar with the application of artificial intelligence (AI) techniques in embedded systems. To become familiar with ML and DL libraries, namely Keras, Scikit-learn, Tensorflow, etc., under Python, as well as their implementation on development boards (MC, Raspberry Pi, or FPGA).

Recommended prior knowledge:

This subject requires prior knowledge of programming languages such as C, VHDL, MatLab or Python.

Content of the material:

TP1. Presentation of the development kit, the programming language (libraries, toolboxes, etc.), and databases (open sources).

TP2. Application of machine learning (k-NN, RF, SVM, etc.) in problems of regression, classification, control, data partitioning and dimension reduction.

TP3. Application of deep learning (DCNN, VGG-16, etc.) in imaging (object detection, classification, etc.), natural language, text translation, anomaly detection, and diagnosis.

TP4. Examples of implementing machine and deep learning in embedded systems (Coral embedded board, Jetson Nano, Arduino, FPGA, Raspberry Pi, etc.)

Assessment method:
Continuous assessment: 100%

Graphic biblio-webo references :

[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, YH, 2017. Python Machine Learning By Example. Packt Publishing Ltd.
[4] . Ketkar, N. and Santana, E., 2017. Deep learning with Python (Vol. 1). Berkeley: A press.
[5] . Kim, P., 2017. Matlab deep learning. With machine learning, neural networks and artificial intelligence, 130(21).
[6] . Gajski, DD, 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, SA, Shelake, VG, Kamat, RK and Naik, GM, 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.

Semester: 02
Teaching unit: UEM 1.2

Subject: Industrial Programmable Automation Practical Work
VHS: 10:30 p.m. (TP: 1:30 p.m.) – Once a fortnight
Credits: 2
Coefficient: 1

Teaching objectives:

Teach students how to install, program, and use a PLC. Show them how to analyze and comply with the technological and security constraints associated with interfacing various industrial components with a PLC. Introduce them to mastering program editing and debugging tasks, as well as correcting detected errors.

Recommended prior knowledge:

- ✓ Microcontrollers;
- ✓ API;
- ✓ Grafcet;
- ✓ Ladder.

Subject content : Choose at least 5 practical work.

TP01: Getting to grips with the API environment: Simulation of an automated system, Review of different software. Introduction to Siemens Step7 software

TP02: Implementing an API: Hardware Configuration. Introduction to Ladder programming (Start-Stop an actuator with hold). Use of digital inputs/outputs: Use of relays, contactors, etc. (do the necessary wiring if necessary).

TP03: Controlling the level of a tank. Using special blocks (interrupts)

Examples of applications: Carry out at least 2 practical exercises from the following list of practical exercises

TP04: Control of a pneumatic cylinder

TP05: Control of traffic lights for a simple intersection

TP06: Control of a bottle filling and transfer unit

TP07: Control of an automated drill

TP08: Transfer and sorting of parts of different dimensions

TP09: Control of a matrix unit

TP10: Control of a Drug Manufacturing Unit

TP11: Control of a Tunnel Kiln

Assessment method:

Continuous assessment: 100%.

Bibliographic references :

[1] . Frank D. Petruzella, Programmable Logic Controllers, 4th edition, Ed. McGraw Hill 2004.
[2] . William Bolton, Industrial Programmable Automation, Dunod Editions, l'Usine Nouvelle, 2010.
[3] . Ian G. Warnock, Programmable Controllers: Operation and Application, Prentice Hall.
[4] . Gilles Michel, Architecture and applications of industrial programmable automatons, Dunod.
[5] . G. Michel, Industrial Programmable Automation, Dunod, 1979.
[6] . S. Thelliez and J.M.Toullote, Grafcet and programmed industrial logic, Eyrolles, 1980.
[7] . JC Bossy, P. Brard, P. Faugère, C. Merlaud, Grafce: its practice and its applications, Educalivre Ed. Casteilla, 1995.

Semester: 3
Teaching unit: UEM 1.2

Subject: Python/Java Programming for Embedded Systems
VHS: 37h50 (Course: 1h30 , Practical work: 1 h) – Choose between Python or Java
Credits: 3
Coefficient: 2

Teaching objectives:

According to the IEEE, Python and Java are the Best Programming Languages in 2021. Python is a very popular and easy-to-learn programming language. It is suitable for both beginners and experts due to its simplicity, readable syntax, and variety of uses. It is essential in all fields: embedded, a web developer, an analyst, a data scientist, or a marketing expert, etc. Java is a pure object-oriented language; it is the most appropriate language for real-world programming . It is present in the software kernels of almost all current technological devices. This subject allows the student to access an appreciable level in Java programming that allows him to deal with this aspect of modern technology.

Recommended prior knowledge:

- ✓ Programming (Pasacal/Matlab language,/C language);
- ✓ Computer Science 1, Computer Science 2, Computer Science 3;

Content of the material:

I) Python (Choose between Python or Java):

Chapter 1. **Installing and Using Python** (1 week)
Chapter 2. **Basics** (2 weeks)
2-A. Interactive mode and script mode ,
2- A-1. Python Calculator ,
*2-A-2. Use of operators: +, -, *, /, //, %, and **,*
2-A-3.c Priority
2-B. Variable and data type :
2-B-1. Variable initialization , Variable modification , Compound assignment
2-B-2. Data type : (. Number , Character , String)
2-B-3. Conversion (str function)
2-C. Predefined function
2-C-1. Use the functions of the math module (abs, max, min, pow, round, sin, sqrt, log, exp, acos, etc.)
2-C-2. Print function
2-C-3. Formatted output (use the format function)
2-C-4. Input function
2-C-5. Function import
2-D. Source code
2-D-1. Variable naming rule
2-D-2. Commentary
Chapter 3. **Conditional Structures** (1 week)
(Minimal if form, if-else form, full if-elif-else form)
The limits of the simple if condition
Comparison operators
Predicates and Booleans
The keywords and, or and not
Chapter 4. **Loops** (1 week)
The while loop
The for loop
Nested loops
The keywords break and continue

Chapter 5. **Functions** (1 week)
Creating functions
Default values of parameters
Signature of a function
The return instruction
The modules,
The import method
The import method: from … import …
The packages
Import packages
Create your own packages
Chapter 6: Lists and Tuples (2 weeks)
Creation and editing of lists
Defining a List, Creating Lists
Insert objects into a list
Add an item to the end of the list
Insert an item into the list
List concatenation
Removing items from a list
The keyword del
The remove method
The list path
The enumerate function
Creating tuples
Chapter 7: Dictionaries (2 weeks)
Creation and editing of dictionaries
Create a dictionary
Deleting keys from a dictionary
The path methods
Key Route
Course of values
Traversing keys and values simultaneously
Dictionaries and function parameters
Chapter 8: Objects and Classes (2 weeks)
Describe objects and classes, and use classes to model objects
Define classes with data fields and methods.
Construct an object using a constructor that invokes the initializer to create and initialize data fields.
Chapter 9: Files (2 weeks)
Relative and absolute paths
Reading and writing to a file
Opening the file
Close the file
Read the entire file
Writing to a file
Writing other data types
The keyword with
Saving objects to files
Save an object to a file

II) Java (Choose between Python or Java):

Chapter 1: Introduction to Java (1 week)
Chapter 2: Control instructions (3 weeks)
A) Choice statements: The *if statement* (The simple *if* , *The* if *with* else part , nested *ifs* and *if-else if*), 2.4 Logical operators, The *switch statement*, Conditional operators, 2.7 Precedence and associativity of operators
B) The loops

2.9 The *while loop*
2.10 The *do...while loop*
2.11 The *for loop*
2.12 Nested Loops
2.13 The keywords *break* and *continue*
The *break statement*
The *break statement* with label
The instruction *continues*
The instruction *continues* with label

Chapter 3: Mathematical Functions, Characters and Strings (3 weeks)

3.2 Commonly Used Mathematical Functions
3.2.1 Trigonometric methods
3.2.2 Exponent Methods (Exponentials and Powers)
3.2.3 Rounding methods
3.2.4 The min, max and abs methods
3.2.5 The random method
3.3 Character Data Type and Operations
3.4 The String type
3.4.1 Getting the Length of a String
3.4.2 Getting Characters from a String
3.4.3 Concatenation of character strings
3.4.4 Converting character strings
3.4.5 Reading a string from the keyboard
3.4.6 Reading a character from the keyboard
3.5 Formatted output to screen (System.out.printf instruction)

Chapter 4: Methods (2 weeks)

4.2 Definition of a method
4.3 Calling a method
Void Methods and Value Return Methods
4.5 Passing parameters by values
4.6 Scope of variables
4.7 Method Overloading

Chapter 5: Tables (3 weeks)

A) One-dimensional arrays
5.2 Basics of Tables
5.2.1 Declaration of tables
5.2.2 Creating tables
5.2.3 Table size and default values
5.2.4 Accessing array elements
5.2.5 Array Initializers
5.2.6 Foreach Loop
5.3 Passing Arrays to Methods
5.4 Returning an array from a method
5.5 Variable-length argument lists
B) Two-dimensional arrays
5.7 Basics of Two-Dimensional Arrays
5.7.1 Declaring and Creating Two-Dimensional Arrays
5.7.2 Obtaining the lengths of two-dimensional arrays
5.7.3 Jagged (perforated) tables
C) Multidimensional arrays

Chapter 6: Objects and Classes (2 weeks)

6.2 Defining Classes for Objects
6.3 Example: Defining classes and creating objects
6.4 Constructing Objects Using Constructors
6.5 Accessing Objects via Reference Variables
6.5.1 Reference variables and reference types
6.5.2 Accessing an object's data and methods

6.6 Static variables, constants and static methods

Assessment method

Continuous assessment: ...40%; Exam: 60%.

Bibliographic references :

[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.. Learning to program with Python 3. Editions Eyrolles, 2012;

[6] . Le Goff, V.. Learn to program in 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, Programming in Java, Editions Eyrolles;

[11] . J. Hunter, Java servlets, O'Reilly;

[12] . P. Niemeyer, J. Knudsen, Introduction to Java, Ed. O'Reilly

Semester: 2
Teaching unit: UED 1.2

Subject 1: optional
VHS: 10:30 p.m. (lesson: 1.5 hours)
Credits: 1
Coefficient: 1

Semester: 2
Teaching unit: UED 1.2
Subject 2: optional
VHS: 10:30 p.m. (lesson: 1.5 hours)
Credits: 1
Coefficient: 1

Semester: 2

Teaching unit: UET 1.2
Subject: Compliance with standards and rules of ethics and integrity.
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives:

To raise student awareness of the ethical principles and rules that govern life at university and in the workplace. To raise awareness of the need to respect and value intellectual property. To explain the risks of moral evils such as corruption and how to combat them, and to alert them to the ethical issues raised by new technologies and sustainable development.

Recommended prior knowledge:

Ethics and professional conduct (the foundations)

Content of the subject:

A.Respect for the rules of ethics and integrity,

1. **Reminder of the MESRS Ethics and Professional Conduct Charter:** Integrity and honesty. Academic freedom. Mutual respect. Demand for scientific truth, Objectivity and critical thinking. Fairness. Rights and obligations of the student, the teacher, the administrative and technical staff,

2. Integrity and responsible research

- Respect for the principles of ethics in teaching and research
- Responsibilities in Teamwork: Professional equality of treatment. Conduct against discrimination. Pursuit of the public interest. Inappropriate conduct in teamwork .
- Adopting responsible conduct and combating abuses: Adopting responsible conduct in research. Scientific fraud. Conduct against fraud. Plagiarism (definition of plagiarism, different forms of plagiarism, procedures to avoid involuntary plagiarism, detection of plagiarism, sanctions against plagiarists, etc.). Falsification and fabrication of data.

3. Ethics and professional conduct in the world of work:

Legal confidentiality in business. Corporate loyalty. Corporate responsibility. Conflicts of interest. Integrity (corruption in the workplace, its forms, consequences, methods of combating and sanctions against corruption).

B- Intellectual property

I- Fundamentals of intellectual property

1- Industrial property . Literary and artistic property.
2- Rules for citing references (books, scientific articles, communications in a congress, theses, dissertations, etc.)

II- Copyright

1. Copyright in the digital environment

Database copyright , software copyright . Specific case of free software.

2. Copyright in the Internet and E-Commerce

Domain name law. Intellectual property on the internet. E-commerce website law. Intellectual property and social networks.

3. Patent

Definition. Rights in a patent. Usefulness of a patent. Patentability. Patent application in Algeria and around the world .

III- Protection and promotion of intellectual property

How to protect intellectual property. Rights violations and legal tools. Valuing intellectual property. Protecting intellectual property in Algeria.

C. Ethics, sustainable development and new technologies

Link between ethics and sustainable development, energy saving, bioethics and new technologies (artificial intelligence, scientific progress, Humanoids, Robots, Drones,

Assessment method:

Exam: 100%

Bibliographic references :

[1] . The World Intellectual Property Organization website www.wipo.int

[2] . Charter of University Ethics and Professional Conduct, https://www.mesrs.dz/documents/12221/26200/Charte+fran__ais+d__f.pdf/50d6de61-aabd-4829-84b3-8302b790bdce

[3] . Order No. 933 of July 28, 2016 establishing the rules relating to the prevention and fight against plagiarism

[4] . The ABCs of Copyright, United Nations Educational, Scientific and Cultural Organization (UNESCO)

[5] . E. Prairat, On teaching ethics. Paris, PUF, 2009.

[6] . Racine L., Legault GA, Bégin, L., Ethics and engineering, Montreal, McGraw Hill, 1991.

[7] . Siroux, D., Deontology: Dictionary of Ethics and Moral Philosophy, Paris, Quadrige, 2004, pp. 474-477.

[8] . Medina Y., Ethics, what will change in the company, Editions d'Organisation, 2003.

[9] . Didier Ch., Thinking about the ethics of engineers, Presses Universitaires de France, 2008.

[10] . Gavarini L. and Ottavi D., Editorial. of professional ethics in training and research, Research and training, 52 | 2006, 5-11.

[11] . Caré C., Morality, Ethics, Deontology. Administration and Education, 2nd quarter 2002, no. 94.

[12] . Jacquet-Francillon, François. Concept: professional ethics. Le télémaque, May 2000, no. 17

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

[14] . Galloux, JC, Industrial Property Law. Dalloz 2003.

[15] . Wagret F. and JM., Patents, trademarks and industrial property. PUF 2001

[16] . Dekermadec, Y., Innovating through patents: a revolution with the internet. Insep 1999

[17] . AEUTBM. The engineer at the heart of innovation. Belfort-Montbéliard University of Technology

[18] . Fanny Rinck and Léda Mansour, Literacy in the Digital Age: Copy-Paste Among Students, Grenoble 3 University and Paris-Ouest Nanterre La Défense University, Nanterre, France

[19] . Didier DUGUEST IEMN, Citing your sources, IAE Nantes 2008

[20] . Similarity Detection Software: A Solution to Electronic Plagiarism? Report of the Working Group on Electronic Plagiarism Presented to the CREPUQ Subcommittee on Pedagogy and ICT

[21] . Emanuela Chiriac, Monique Filiatrault and André Régimbald, Student Guide: Intellectual Integrity, Plagiarism, Cheating and Fraud... Avoiding Them and, Above All, How to Properly Cite Your Sources, 2014.

[22] . Publication of the University of Montreal, Strategies for preventing plagiarism, Integrity, fraud and plagiarism, 2010.

[23] . Pierrick Malissard, Intellectual Property: Origin and Evolution, 2010.

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

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

V - Detailed program by subject for semester S3

Semester: 3
Teaching unit: UEF1
Subject 1: Artificial vision system
VHS: 45 hours (Class: 1:30, TD: 1:30)
Credits: 4
Coefficient: 2

Teaching objectives:

Understand the main components of an artificial vision system and the role of each component in its design. Study the tools that enable the automatic reproduction of tasks performed by the human visual system and interpreted by the brain.

Recommended prior knowledge:

- ✓ Signal processing;
- ✓ Image processing.

Content of the subject:

Chapter 1. Image acquisition and digitization (2 weeks)
Functional composition of an artificial vision system, Image definition, Image sensors, image digitization , Human vision , 3D modeling and camera calibration.

Chapter 2. Reminders on image processing (2 weeks)
Point operations (Logarithmic transformation, Contrast inversion, Histogram modification) , Local operations (Spatial filtering, Frequency filtering)

Chapter 3. Contours and segmentation (3 weeks)
Contour detection (Contour definition, gradient approach, Lapalucian approach, Canny filtering, LOG filtering, Active contours) Segmentation (Histogram thresholding, Region approaches)

Chapter 4. Motion (2 weeks) Motion estimation, optical flow, Horn & Schunk method, Block matching Gunnar Farneback approach

Chapter 5. Feature Detection (2 weeks)
Definition of a point of interest, geometric transformations, Moravec detector, Harris detector, SIFT
Chapter 6. Machine Learning (3 weeks)
Definition of machine learning, supervised classification (K-PPV, Naive Bayes, SVM), unsupervised classification (K-Means, Fuzzy C-Means)

Evaluation method :
Continuous assessment: 40%; Exam: 60%.

Bibliographic references :

[1] .R. Gonzalez, Digital Image Processing *, Upper Saddle River, NJ, Prentice Hall, 2002 (ISBN 978-0-201-18075-6).*
[2] . M. Bergounioux , Introduction to mathematical image processing: deterministic methods *, vol. 76, Berlin, Heidelberg, Springer Berlin Heidelberg, coll. "Mathematics and Applications", March 19, 2015*
[3] . A. Herbulot, Non-parametric statistical measures for image and video segmentation and minimization by active contours, Doctoral thesis defended at the University of Nice - Sophia Antipolis, October 10, 2007

[4] . C.Grava, Motion compensation by cellular neural networks: Application in medical imaging, Doctoral thesis, defended at INSA Lyon 2003.

[5] . JP. Cocquerez and S. Phillip, Preface H.Maitre Image Analysis: Filtering and Segmentation, Edition Masson, 1995

[6] . HP 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 recognition 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, Lecture notes, Unsupervised classification, AgroParisTech, 2017

Semester: 3
Teaching unit: UEF 2.1.1
Subject 2: Real-time systems
VHS: 67h30 (Lecture: 3h00, Tutorial: 1h30)
Credits: 6
Coefficient: 3

Teaching objectives:

Present the architecture of a real-time operating system and programming techniques in a real-time language.

Recommended prior knowledge:

- ✓ Design of microprocessor systems;
- ✓ Computer programming;

Content of the subject:

Chapter 1: Generalities, concepts and terminologies on operating systems (1 week)
- Reminders on the architecture of a computer.
- The operating system (definition, role, structure etc.)
- Concepts of instruction and macro-instruction
- Program, Process, Function...

Chapter 2: Introduction to real-time systems (1 week)
- Definition of a real-time system
- Real-time constraints. Specifications
- Classification of real-time systems.
- Characteristics and Structure of a Control System.

Chapter 3: Scheduling in Classic Operating Systems (2 weeks)
- Concept of process and process states
- Operation on processes
- Process scheduling criteria
- FCFS, SJF, SRTF & Round Robin Scheduling Algorithm

Chapter 4: Real-time scheduling (4 weeks)
- Concept of real-time tasks. Modeling and characterization of tasks.
- Scheduling independent tasks
- Scheduling dependent tasks
- Scheduling in overload situations

Chapter 5: Multiprocessor Real-Time Scheduling (2 weeks)
- Position and formulation of the problem
- Comparison with single-processor scheduling
- Multiprocessor scheduling anomalies
- Conditions of ordinability
- The Earliest Deadline and Least Laxity algorithms

Chapter 6: Memory and Communication Management (3 weeks)
- Management of virtual memory and physical memory (paging, addressing, allocation, etc.)
- problems of competition, cooperation, synchronization

- Semaphore, monitors, ...
- Inter-task communication and messages

Chapter 7: Programming (2 weeks)

- Introduction to Concurrent Programming
- Management of multitasking aspects, Mutual exclusion, Synchronization, Communication... Real-time programming (JAVA real-time, ADA)
- Examples of applications.

Assessment method:

Continuous assessment: 40%; Exam: 60%.

Bibliographic references :

[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, Real-time systems: Scheduling, networks and quality of service, Hermès – Lavoisier, Volume 2, 2006.
[6] . Alan C. Shaw, Real-time systems and software, John Willey & Sons, Inc., 2001.
[7] . Francis Cottet and Emmanuel Grolleau, Real-Time Control and Command System, 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, Real-time scheduling: Course and corrected exercises, Hermès Science Publications, 2000.
[11] . A. Darseoil, P. Pillot, Real Time in an Industrial Environment, Dunod 1991.
[12] . Y. Trinquet, J.-P. Elloy, Real-Time Executives, Engineering Techniques.

Semester: 3
Teaching unit: UE F 2.1.2
Subject 1: Embedded Systems
VHS: 45h00. (Lecture: 1h30, Tutorial: 1h30)
Credits: 4
Coefficient: 2

Teaching objectives:

In this subject, students will be able to specify and produce an application or an embedded system on a Windows basis.

Recommended prior knowledge:

The content of this subject assumes mastery of programming techniques as well as an average knowledge of a Windows or Unix type operating system and their network layers .

- ✓ Microcontrollers and Microprocessors
- ✓ Algorithmic;
- ✓ ASM, C/C++, etc. programming.
- ✓ Basics of Operating Systems
- ✓ Integrated development software (Proteus Tool suite, Mplab, CCS)

Content of the material:

Chapter 1. Introduction to Embedded Systems (2 weeks)

1.1. History of embedded systems
1.2. Definition of embedded systems
1.3. Development of embedded systems
1.4. Characteristics of embedded systems
1.5. Some properties of embedded systems
1.6. Design constraints of an embedded system
1.7. Some examples of embedded systems
1.8. Areas of application of embedded systems:
1.9. The challenges of the embedded:

Chapter 2. Embedded Systems Architecture (3 weeks)

2. Hardware aspect of an embedded system
2.2 Von Neumann Architecture
2.3 Harvard Architecture
2.4 The microcontroller and its internal structure
2.5 The structure of an assembler program for PIC
2.6 PIC16F84 Interrupt Management
2.7 Managing the PIC16F84's internal timers
(Timing and pulse counts)
2.8 Practical example of an embedded system based on a microcontroller

Chapter 3. Real-time systems: General concepts (3 weeks)

3.1 Interactive System
3.2 Reactive or Real-Time System
3.3 Real-Time Systems
3.3.1 Real-time systems with hard time constraints
3.3.2 Real-Time Systems with Soft Time Constraints
3.4 General architecture and operating modes

3.4.1 General operation: infinite loop
3.4..2 Cyclic Operation
3.4..3 Event Operation
3.5 Characteristics of a real-time system.
3.6 Real-time system examples
Chapter 4. Programming Languages for Embedded Systems (2 weeks)
CCS C-PCW Compiler Basic Rules
4.2 Variables and constants
4.3 C-CCS language operators
4.4 Repetitive structures.
4.5 C-CCS functions adapted to microcontrollers
4.6 I/O Management
4.7 Managing timeouts
4.8 Interrupt Management
4.9 Serial Link Management
4.10 Examples
Chapter 5. Embedded and Multitasking Operating System (3 weeks)
5. Introduction
5.2 State Machine
5.3 Concepts of real-time operating systems for embedded systems (RTOS)
5.3.1 The scheduler
5.3.2 RTOS Services
5.3.3 Synchronization and Message Transmission Tools
5.3.4 Application example

Chapter 6. Case study: development of an embedded application such as the implementation of a PID corrector for temperature regulation (2 weeks)

Assessment method:

Continuous assessment: 40%; Exam: 60%.

Bibliographic references :

[1] . http://beru.univ-brest.fr/~singhoff/supports.html
[2] . Tim Wilmshurst., Designing Embedded Systems with PIC Microcontrollers: Principles and applications.
[3] . Christian Tavernier, AVR Microcontrollers: from ATtiny to ATmega - Description and implementation
[4] . Christian Tavernier, Pic 18 Microcontrollers - Description and Implementation
[5] . Alexandre Nketsa; Programmable logic circuits: PLD, CPLD and FPGA memories, industrial computing
[6] . A. Dorseuil and P. Pillot. Real time in an industrial environment. DUNOD Edition, Industrial Computing Collection, 1991.
[7] . Ch. Bonnet, I. Demeure, Introduction to real-time systems, HERMES Edition
[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 to Real-Time Systems. Educational Telecommunications Collection, Hermès, September 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/

Semester: 3
Teaching unit: UE F 2.1.2
Subject 2: Industrial Networks and Communications
VHS: 45h00. (Lecture: 1h30, Tutorial: 1h30)
Credits: 4
Coefficient: 2

Teaching objectives:

To enable students to become familiar with the concepts of digital data transmission, particularly the different types of networks that exist in the industrial world. Emphasis will be placed on understanding the different topologies and their advantages and disadvantages for a given industrial installation.

Recommended prior knowledge:

- ✓ Local computer networks ;
- ✓ Sensors;
- ✓ Industrial programmable controllers ;

Content of the material:

Chapter 1. General information on field buses (4 weeks)

1.1-Definitions (Bus, Field, Network, Local Area Network, Industrial Local Area Network (ILN), Enterprise Local Area Network, Some names of industrial local area networks). 1.2-Presentation of the industrial environment. 1.3-Architecture of an IDN (field networks, workshop networks, factory networks). 1.4-Characteristics of an IDN (number of nodes, amount of information, transmission time). 1.5-Characteristics of the data exchanged in an IDN (Nature of the messages exchanged, Size of the messages). 1.6-Role of an IDN in an industrial installation. 1.7-OSI and IDN architecture (Adaptation of the OSI model to IDNs, Characteristics of the physical layer for IDNs, Characteristics of the MAC sublayer for IDNs).

Chapter 2 : The 485 Modbus bus (2 weeks)

Reminder of the RS232 standard. The RS485 link. The Modbus protocol. Modbus addressing and framing.

Chapter 3. The CAN (Controller Area Network) bus (3 weeks)

Overview of CAN. CAN OSI models. CAN data frames and characteristics. Access methods and arbitration principle. Data rates. CAN hardware. Application layer services. CANopen.

Chapter 4. Profibus (3 weeks)

Overview of Profibus and its characteristics. The three types of Profibus (DP, FMS, and PA). Access mode. Industrial Ethernet and Profinet. Data rates.

Chapter 5. Overview of Wireless Industrial Networks (3 weeks)

Technologies, protocols and architectures of industrial wireless networks (WLAN 802.11, Bluetooth, HART protocols, Wireless Profibus, Bluetooth, ZigBee, etc.). Security of industrial wireless communication networks.

Assessment method:

Continuous assessment: 60%; Exam: 40%.

Bibliographic references :

[1] . Belgacem Jarray, Industrial Networks: Buses, Interfaces, Industrial Ethernet, Hart. Course and corrected exercises. Ellipses, 2017.
[2] . JF Hérold, O. Guilloton and P. ANAYA, Industrial computing and networks in 20 files. Dunod, 2010.
[3] . Jean-Pierre Thomesse, Local Industrial Networks. Eyrolles, 1994.
[4] . Pascal Vrignat, Local Industrial Networks - Courses and Practical Work. Gaëtan Morin, 1999.
[5] . Ciame, Field networks: operational safety criteria. Lavoisier, 2009.
[6] . Ciame, Field networks: Description and selection criteria. Hermes, 2001.

Semester: 3
Teaching unit: UEM 2.1
Subject 1: Practical work on embedded systems / Practical work on real-time systems
VHS: 10:30 p.m. (TP: 1:30 p.m.)
Credits: 2
Coefficient: 1

Teaching objectives:

To help students understand the practical aspects of embedded and real-time systems. Design of an embedded system dedicated to the automobile where analog and digital inputs/outputs, timer, interruption, ADC-DAC conversion, LCD display, 7 segments, serial communication (RS232, etc.), WIFI communication, Bluetooth, design of an HMI interface will be used.

Recommended prior knowledge:

C Programming, DOS Commands.

Content of the material:

A.Embedded systems practical work - Once a fortnight

B. Simulation/Implementation of applications using the following hardware/software pairs: FPGA/VHDL, Arduino/C, Microprocessor/Assembler, PIC/C or Assembler, DSP/C, etc.

Application example: Design of a central locking system for an automobile

Below is an example of the methodology applied to the design of a central locking system for an automobile. The teacher is free to choose another application.

Initial step: Develop a program that allows the four doors of a car to be opened and closed using infrared or radio frequency control, using the TOR control with the lighting (or flashing) of red LEDs and the emission of beeps of varying duration.
Variant 1: Take into account the actual closing of the doors (using stops (push button).
Option 2: Provide for the case of doors being opened by mistake (accidental action of the infrared control) without the door actually being opened. Automatic locking after a certain time.
Option 3: Plan for the case of a door not properly closed while the car is moving. Alert the driver: beep, display on the dashboard, etc.
Variant 4: Encryption of the control signal

Lead the student to imagine a global architecture and to propose different programming techniques (interruption, polling, functions, communication (synchronous asynchronous), I/O management, Display, Alarm, etc.) and estimate the cost of the application each time.

C.Real-time systems practical work - Once a fortnight

TP 01. Know how to use basic MS-DOS commands, write and execute a DOS batch script. Modify files and launch commands.
TP 02. Introduction to Linux Commands: Process Management: Creating (Launching), Viewing (Listing) and Stopping (Killing) External Processes. Memory Management in Linux (Understanding Dynamic Memory Allocation, Diagnosing Some Problems Related to Dynamic Allocation)
TP 03. Processing a simple example (semaphore case) using one of the real-time languages.

Assessment method:

Continuous assessment: 100%

Bibliographic references :

Semester: S3

Teaching unit: UEM 2.1
Subject: Industrial Networks Practical Work
VHS: 10:30 p.m. (TP: 1h30)
Credits: 2
Coefficient: 1

Teaching objectives:

At the end of this subject with the associated Lecture/Tutorial subject, the student will be able to use industrial software (Step-7, Unity Pro, Studio 5000, etc.) to configure and program an industrial network (Profibus, Profinet, CAN, Modbus, etc.) containing the main communication elements including: the industrial PC, the HMI, the PLC and the input/output modules. He will also have a good overview of the different protocols included in the OSI layers of field buses (access methods, frame structures, coding, etc.).

Recommended prior knowledge:

- ✓ Programmable logic controllers (PLC);
- ✓ Programming languages for PLC: Contact, Log, List, StI, Graph (Grafcet);
- ✓ Local area network (LAN);
- ✓ Sensors and actuators;

Content of the material:

TP1: PLC (Inputs/Outputs) with Step 7, TIA Portal or Unity Pro. Programming with Graph and/or Ladder and/or STL and/or Log and/or List.
TP2: Profibus with STEP 7 or TIA Portal. Master with passive slave and/or intelligent slave. Simulation and/or practice. Use of a protocol analyzer to decode Profibus messages.
TP3: Profinet with STEP 7 or TIA Portal. PLC1 + PLC2 + HMI or industrial PC. Simulation and/or practice. Use of a protocol analyzer to decode Profinet messages.
TP4: Serial Modbus with Unity Pro. Master + Slave + Operating Screen. Using a protocol analyzer to decode Modbus messages.
TP5: Modbus TCP/IP with Unity Pro. PLC1 + PLC2. Simulation and/or practice. Use of a protocol analyzer to decode Modbus messages.
TP1: Implementation and operation on RS232, RS485.
TP2: Implementation of the CAN Bus between Arduino modules.
TP3: Modbus dialogue between a master PC and a slave device. Modbus master communication
from an industrial programmable logic controller.
TP4: Study of Profibus industrial network (based on Arduino or visit to a factory).
TP5: Data exchange via industrial Ethernet network.
TP6: Development of a wireless industrial local network .

Assessment method:

Continuous assessment: 100%.

Bibliographic references :

[1] . https://www.se.com/;
[2] . https://support.industry.siemens.com/ .
[3] . Belgacem Jarray, Industrial Networks: Buses, Interfaces, Industrial Ethernet, Hart. Course and corrected exercises. Ellipses, 2017.
[4] . JF Hérold, O. Guilloton and P. ANAYA, Industrial computing and networks in 20 files. Dunod, 2010.
[5] . Jean-Pierre Thomesse, Local Industrial Networks. Eyrolles, 1994.
[6] . Pascal Vrignat, Local Industrial Networks - Courses and Practical Work. Gaëtan Morin, 1999.
[7] . Ciame, Field networks: operational safety criteria. Lavoisier, 2009.
[8] . Ciame, Field networks: Description and selection criteria. Hermes, 2001.

Semester: 3
Teaching unit: UEM2.1

Subject 3: Practical work on artificial vision
VHS: 10:30 p.m. (TP: 1:30 p.m.)
Credits: 2
Coefficient: 1

Teaching objectives:

To present and discuss the methodologies applied to computer vision. The concepts covered and their applications should, on the one hand, prepare students for modern computer vision tools and, on the other hand, lead them to master the ideas and techniques that allow them to integrate a digital imaging and vision system into an industrial application.

Recommended prior knowledge:

Signal processing, Image processing.

Content of the material:

TP1 : introduction to the use of OpenCV

- Representing images and videos in OpenCV
- Color and palette treatments

TP2 : Camera calibration and 3D reconstruction

TP3 : Contour detection and segmentation

TP4 : Motion detection and estimation

TP5 : Feature detection

TP6 : Object tracking

TP7 : Machine Learning

Assessment method:

Continuous assessment: 100%

Bibliographic references

Semester: 3
Teaching unit: UEM 1.3
Subject 4: Study and Implementation of Projects
VHS: 37h30 (Lecture: 1h00, Practical work: 1h30)
Credits: 3
Coefficient: 2

Teaching objectives:

Produce an electronic card that constitutes an embedded system for a given application.
Joint code development and hardware design.

Recommended prior knowledge:

Programming, microprocessor and microcontroller systems.

Content of the material:

Chapter 1: Study of simulation software (3 weeks)
Getting started with a design environment (e.g. " *Proteus Design Suite* "), simulation, analysis of electronic circuits and development of printed circuits

Chapter 2: Definition and management of a project (2 weeks)
Study of the project (a simple embedded system meeting the needs of a particular sector such as medical, automotive or home automation) establishment of the specifications, technical choices, cost, schedule and planning of the execution of the works, documentation, choice of components .

Chapter 3: Creation of the electronic part (3 weeks)

Chapter 4: Implementation of the software part (3 weeks)

Chapter 5: Simulation and Testing (2 Weeks)

Chapter 6: Technical Report (2 Weeks)
Drafting of the technical file and defense.

Assessment method:

Continuous assessment: 100%; Exam: 00%.

Bibliographic references :

[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/project-management/
[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/

Semester: 3
Teaching unit: UED 2.1
Subject 1: optional
VHS: 10:30 p.m. (Class: 1.5 hours)
Credits: 1
Coefficient: 1

Semester: 3
Teaching unit: UED2.1
Subject 2: optional
VHS: 10:30 p.m. (Class: 1.5 hours)
Credits: 1
Coefficient: 1

Semester: 3
Teaching unit: UET2.1
Subject: Documentary research and dissertation design
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives :
To give students the tools they need to research useful information and use it more effectively in their final year project. To help them navigate the various stages of writing a scientific document. To demonstrate the importance of communication and to teach them how to present their work in a rigorous and educational manner.

Recommended prior knowledge:
Writing methodology, Presentation methodology.

Content of the subject :

Part I-: Documentary research:

Chapter I-1: Definition of the subject (02 Weeks)
- Subject title
- List of keywords related to the subject
- Gather basic information (acquisition of specialized vocabulary, meaning of terms, linguistic definition)
- The information sought
- Take stock of your knowledge in the field

Chapter I-2: Selecting information sources (2 weeks)
- Type of documents (Books , Theses, Dissertations, Periodical articles, Conference proceedings, Audiovisual documents, etc.)
- Type of resources (Libraries, Internet, etc.)
- Evaluate the quality and relevance of information sources

Chapter I-3: Locate documents (01 Week)
- Research techniques
- Search operators

Chapter I-4 : Processing information (2 weeks)
- Work organization
- The starting questions
- Summary of the documents selected
- Links between different parties
- Final plan of the documentary research

Chapter I-5: Presentation of the bibliography (01 Week)
- Bibliography presentation systems (The Harvard system, The Vancouver system, The mixed system, etc.)
- Presentation of documents.
- Citation of sources

Part II: Memory Design

Chapter II-1 : Plan and stages of the dissertation (2 weeks)

- Identify and delimit the subject (Summary)
- Problems and objectives of the thesis
- Other useful sections (Acknowledgments, Table of abbreviations, etc.)
- The introduction (*Writing the introduction last)*
- State of the specialized literature
- Formulation of hypotheses
- Methodology
- Results
- Discussion
- Recommendations
- Conclusion and perspectives
- Table of Contents
- The bibliography
- The annexes

Chapter II- 2 : Writing techniques and standards (2 weeks)

- Formatting. Numbering of chapters, figures and tables.
- The cover page
- Typography and punctuation
- Writing. Scientific language: style, grammar, syntax.
- Spelling. Improvement of general language skills in terms of comprehension and expression.
- Save, secure, archive your data.

Chapter II-3 : Workshop: Critical study of a manuscript (1 Week)

Chapter II-4 : Oral presentations and defenses (1 Week)

- How to present a poster
- How to present an oral communication.
- Defense of a dissertation

Chapter II-5: How to avoid plagiarism ? (1 Week)

(Formulas, sentences, illustrations, graphs, data, statistics, etc.)

- The quote
- The paraphrase
- Indicate the full bibliographic reference

Assessment method:

Exam: 100%

Bibliographic references:

1. *M. Griselin et al., Guide to Written Communication, 2nd edition, Dunod, 1999.*
2. *JL Lebrun, Practical guide to scientific writing: how to write for the international scientific reader, Les Ulis, EDP Sciences, 2007.*
3. *A. Mallender Tanner, ABC of technical writing: user guides, instructions, online help, Dunod, 2002.*
4. *M. Greuter, How to write your dissertation or internship report well, L'Etudiant, 2007.*
5. *Mr. Boeglin, Reading and Writing at University. From the Chaos of Ideas to Structured Text. L'Etudiant, 2005.*
6. *Mr. Beaud, the art of the thesis, Editions Casbah, 1999.*
7. *Mr. Beaud, the art of the thesis, La découverte, 2003.*
8. *Mr. Kalika, Master's thesis, Dunod, 2005.*

Suggestion of some discovery subjects

Half:
Teaching unit: UED
Subject 1: RFID radio-identification
VHS: 10:30 p.m. (Class: 1.5 hours)
Credits: 1
Coefficient: 1

Teaching objectives:

Acquire sufficient technical and practical knowledge of RFID technology with a view to its implementation in projects on embedded systems.

Recommended prior knowledge:

Architecture of microprocessor systems

Content of the material:

- Presentation, definition and history
- Ethics, privacy,
- Obstacles to the use of RFID: metallic environment, collisions,
- RFID Tag Classifications
- Operating principle
- read-only or read/write?
- TTF and ITF protocols
- Applications of RFID
- field communication: NFC
- how NFC/RFID works
- the main features
- NFC applications

Assessment method:

Exam: 100%.

Bibliographic references:

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/RFID_communication_protocols*
6. *https://fr.wikipedia.org/wiki/Radio-identification*
7. *https://fr.wikipedia.org/wiki/Communication_en_champ_proche*

Half:
Teaching unit: UED
Subject 2: Home automation
VHS: 10:30 p.m. (Class: 1.5 hours)
Credits: 1
Coefficient: 1

Teaching objectives:

Provide the student with all the equipment of a smart home, their operation and their uses so that they are able to size and design a home automation installation.

Recommended prior knowledge:

Microprocessor systems, sensors, etc.

Content of the material:

Chapter 1. Comfort in buildings (1 week)
- Thermal, acoustic and visual,

Chapter 2. Concepts on the safety of property and people (7 Weeks)
- Fire safety, Access control, Anti-intrusion, Video surveillance, Remote surveillance, ...

Chapter 3. Technical building management and communication (7 weeks)
- Lighting, Air conditioning, Heating, Regulation, Networks, Remote management, Supervision, BMS (technical building management), GTC (centralized technical management), . ..

Assessment method:

Review: 100%.

Bibliographic references:

1. *C. Locqueneux, The Guide to the Home and Connected Objects, Eyrolles 2016*
2. *FX. Jeuland, The Communicating House, Eyrolles, 2008 (2nd edition)*
3. *PROMOTELEC, Communicating Habitat, Promotelec Editions, 2006*
4. *EA Decamps, Home Automation, University Press of France, "What do I know?" Collection, 1988.*
5. *M. Al-Qutayri, Smart Home Systems, In-Teh, Croatia 2010*
6. *C. Nugent, Smart Homes and Beyond, IOS Press, Netherlands 2006*

Semester:
Teaching unit: UED
Subject 3: Embedded systems for automobiles
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives:

The aim of this subject is to provide students with the necessary foundations to know how to develop and design on-board electronics applications for the automotive industry, which is a discipline in its own right aimed at optimally controlling the circulation and safety of a vehicle.

Recommended prior knowledge:

Sensors and Instrumentation.

Content of the subject:

Chapter 1: Introduction to Embedded Systems

Chapter 2: On-board sensors
Speed and flow sensors, Acceleration sensors, Temperature sensors, Pressure sensors, Proximity sensors, Gyroscopic sensors.

Chapter 3: On-board actuators
Hydraulic Actuator, Air Bag Actuator, Air Conditioning System, Braking System.

Chapter 4: Vehicle System Architecture
Electronic calculator, CAN communication bus, Sensor / Actuator networks.

Chapter 5: Embedded Systems in Automotive
On-board sensor systems, Anti-lock braking system (ABS), Anti-skid system (ASR), Electronic stability control (ESP), Measurement of wheel rotation speed (encoder) and vehicle speed (Doppler effect).

Chapter 6: Typical architecture of a vehicle model manufactured in Algeria

Assessment method:

Exam: 100%

Bibliographic references :

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, Data Acquisition: From Sensor to Computer, Dunod, 2003.*
6. *G. Asch et al. Data Acquisition, 3rd edition, Dunod, 2011.*
7. *M. Bayart, B. Conrard, A. Chovin, M. Robert, Intelligent sensors and actuators, 2005.*
8. *P. Dassonvalle, The Sensors: Exercises and Corrected Problems, Dunod, 2005.*
9. *R. Frank, Understanding Smart Sensors (Artech House sensors library), 2nd edition.*
10. *F. Boudoin, M. Lavabre, Sensors: main uses, Edition Casteilla, 2007.*
11. *JG Webster, Measurement, Instrumentation and Sensors Handbook, Taylor & Francis Ltd*

Semester:
Teaching unit: UED
Subject 4: Embedded Systems Operating Systems
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives:

Allow students to learn the fundamental principles of operating systems while studying their practical application in an embedded system such as Android.

Recommended prior knowledge:

Basic concepts in mathematics, algorithms and programming.

Content of the subject:

Chapter 1. (4 weeks)
General presentation of operating systems and technical elements (e.g.: OS, Android, Windows and Linux)

Chapter 2. (3 weeks)
Process management

Chapter 3. (3 weeks)
Memory management

Chapter 4. (3 weeks)
File management

Chapter 5. (2 weeks)
Executable

Assessment method:

Exam: 100%

Bibliographic references :

1. *Andrew Tanenbaum , Operating Systems, Pearson Publishing.*
2. *Michael Griffiths , Michel Vayssade , Operating Systems Architecture , Hermes - 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*

Semester:
Teaching unit: UED
Subject 5: Smart cards
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives:

Acquire sufficient technical knowledge on the technology, operation and use of smart cards with a view to their implementation in projects on embedded electronic systems.

Recommended prior knowledge:

Architecture of microcontroller and/or microprocessor systems.

Content of the subject:

- General information, History, Applications and markets of the smart card.
- Semiconductors for smart cards, Technologies, Hardwired logic components, Microcomputers.
- Cryptology and security, Principles of cryptography, Symmetric cryptosystems, Asymmetric cryptosystems, Zero-knowledge cryptosystems, Physical and logical security of smart cards.
- Construction principles, Interconnection of components, Insertion, Connections.
- Smart card operating systems, Generalities and basic mechanisms, Closed operating systems, Open operating systems.
- Contact communication, Radio frequency communication.

Assessment method:

Exam: 100%

Bibliographic references :

1. *W. Rankl and W. Effing, Smart Card Handbook, Wiley, 2010.*
2. *C. Tavernier, Smart cards, Dunod, 2011.*

Semester:
Teaching unit: UED
Subject 6: Mobile Robotics
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives:

The aim of this subject is to provide students with the necessary foundations to know how to develop and design applications of embedded electronics for mobile robotics, which is a discipline in its own right aimed at controlling movement.

Recommended prior knowledge:

Mathematics, Programming.

Content of the subject:

Chapter 1: Classification and Modeling of Mobile Robots (MR)	(2 weeks)
Chapter 2: Sensors used in MR	(3 weeks)
Chapter 3: RM Localization	(2 weeks)
Chapter 4: Representing the environment of an RM	(2 weeks)
Chapter 5: Path planning techniques	(2 weeks)
Chapter 6: Navigation and SLAM techniques	(2 weeks)
Chapter 7: Humanoid Robots	(2 weeks)

Assessment method:

Exam: 100%

Bibliographic references :

1. *R. Siegwart, IR Nourbakhch, D. Scaramuzza, Introduction to Autonomous Mobile Robots, 2nd Edition , MIT Press, 2011.*
2. *L. Jaulin, Mobile robotics, Editions ISTE, 2015.*
3. *V. Maille, C. Accard, B. Breton, Robots: learning robotics by example, Editions Ellipse, 2016.*

Semester:
Teaching unit: UED
Subject 7: Wireless Communication
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives:

To enable students to become familiar with wireless communications systems in order to use them in electronic applications of embedded systems. Among the various wireless networks, we will focus more specifically on the WIFI network of the IEEE 802.11 standard.

Recommended prior knowledge:

Basic concepts in mathematics, statistics and signal processing.

Content of the subject:

Chapter 1. Wireless Networks (4 weeks)

Chapter 2. Introduction to WIFI (802.11): (3 weeks)
Different WIFI standards and equipment

Chapter 3. WIFI Implementation: (3 weeks)
Infrastructure mode, ad hoc mode and network setup

Chapter 4. Encryption: (3 weeks)
WEP, WAP ...

Chapter 5. Hacks and Solutions: (2 weeks)
MAC address filtering, IP address definitions and firewall installation

Assessment method:

Exam: 100%

Bibliographic references :

1. *Tanenbaum, Networks, 4th edition, Prentice Hall, 2003.*
2. *R. Parfait, Telecommunications networks, Hermès science publications, 2002.*
3. *E. Hollocou, Telecommunications techniques and networks, Armand Colin, 1991.*
4. *C. Servin, Networks and Telecoms, Dunod, Paris, 2006.*
5. *D. Dromard and D. Seret, Network Architectures, Pearsont Editions , 2009.*
6. *P. Polin, Networks: Fundamental Principles, Edition Hermès.*
7. *D. Comer, TCP/IP, architectures, protocols and applications, Editions Interéditions.*
8. *D. Present, S. Lohier, Transmissions and Networks, course and corrected exercises, Dunod.*
9. *P. Clerc, P. Xavier, Fundamental Principles of Telecommunications, Ellipses, Paris, 1998.*
10. *D. Battu, Introduction to Telecoms: Technologies and Applications, Dunod, Paris, 2002.*
11. *P. Rolin, G. Martineau, L. Toutain, A. Leroy, Networks, fundamental principles, Hermès edition, 1997.*

Semester:
Teaching unit: UED
Subject 8: Robotics
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives:

To introduce the student to the fundamental aspects of robotics and recent developments in the field of industrial robotics.

Recommended prior knowledge:

None.

Content of the subject:

Chapter 1: General Information
Definitions, Components of a robot, Classification of robots, Characteristics of a robot, Generations of robots, Programming robots.

Chapter 2: Degrees of Freedom - Architecture
Positioning of a solid in space, Connection, Mechanisms, Morphology of robots, manipulators

Chapter 3: Geometric model of a simple chain robot
Need for a model, Operational coordinates, Translation and rotation, Homogeneous transformation matrices, Obtaining the geometric model, Modified Denavit-Hartenberg parameters, Inversion of the geometric model - Paul's method, Multiple solutions - Workspace - Aspects

Chapter 4: Simplification Technique
Speed and acceleration of robots, Jacobean matrix and its usefulness, Definition of direct and inverse equations, Meaning of singularities.

Assessment method:

Exam: 100%

Bibliographic references :

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

Semester:
Teaching unit: UED
Subject 9: Renewable energies: photovoltaic solar energy
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives:

This subject covers concepts relating to non-polluting renewable energies, photovoltaic (PV) devices, PV conversion, solar cell manufacturing processes, PV module assemblies, their degradation, etc. It also covers auxiliary systems: the battery, the fuel cell (with hydrogen as the energy vector), converters, etc. The subject will also focus on the different loads to be supplied, whether continuous or alternative, by looking for all the possibilities of coupling with a PV generator, the description of a global PV system, its characteristics and the optimization of the system's operation.

Recommended prior knowledge:

Notions on semiconductors, radiation physics, mathematics, electronics...

Content of the subject:

Chapter 1: Renewable Energies

Forms of energy, What is renewable energy, Main renewable energies, The global energy situation, ...

Chapter 2: The Solar Source

Solar radiation, Solar deposit, Solar energy (thermal, photovoltaic, thermodynamic)

Chapter 3: The photovoltaic source

Photovoltaic conversion, Solar cell technology, Properties of solar cells, Modeling of a photovoltaic cell (module) (electrical, thermal modeling, etc.), Conversion efficiency, form factor, etc., Different connections (series, parallel, mixed), Impact of various factors on electrical characteristics, Degradation, Protection of photovoltaic modules, Applications of photovoltaic energy (pumping, connection to the network, etc.).

Chapter 4: Photovoltaic Systems

Direct connection between photovoltaic generator and load, Storage (Battery), Fuel cell, Chopper, Inverter, Study of an example of a global system (hybridization), Problem of sizing a photovoltaic installation, *Maximum Power Point Tracker* (MPPT).

Assessment method:

Exam: 100%

Bibliographic references :

1. *A. Vapaille, Semiconductor Devices and Integrated Circuits, Dunod , 1987.*
2. *M. Orgeret, solar batteries, Masson, 1985.*
3. *A. Ricaud, Solar photocells, Polytechnic and University Presses of Romandie, 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 of electrical energy from renewable sources, Engineering Techniques, Electrical Engineering Treatises, D4005/6, May 2003.*
7. *J. Nelson, The physics of solar cells, Imperial College Press.*
8. *A. Labouret, P. Cumune, Solar Cells, 5th edition - The basics of photovoltaic energy, Dunod, 2010*
9. *A. Labouret, Photovoltaic solar energy, 3rd edition, 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*

Semester:
Teaching unit: UED
Subject 10: Autonomous Energy Systems
VHS: 10:30 p.m. (Class: 1.5 hours)
Credit: 1
Coefficient: 1

Teaching objectives:

To stimulate the student's interest in renewable energy in general and in energy systems using solar or wind energy in particular. To give the student some skills in the sizing of a wind or photovoltaic installation .

Recommended prior knowledge:

semiconductors, radiation physics, mathematics, electronics...

Content of the subject:

Chapter 1: Electrical energy production devices
Notions on energy transformations (mechanical; thermal; hydraulic, etc.), History (Volta, Oersted, Faraday, etc.), the alternator, the dynamo, methods of producing electrical energy (hydraulic power station, thermal power stations). Non-renewable energy sources (fossil and nuclear). Renewable energy sources.

Chapter 2: Wind Energy
History, principle and structure, Characteristics and sizing, Map of wind resources in Algeria, Wind farms and power, Standards, Advantages and disadvantages. Example of a wind turbine installation.

Chapter 3: Hybrid Systems
Hybrid Systems (Hydrokinetic Systems, Operating Principle of Hydrokinetic Systems, Different Types of Hydrokinetic Systems and Operators, etc.)

Chapter 4: Photovoltaic solar energy
Principle of a photovoltaic installation, solar deposits in Algeria, photovoltaic cell technologies, photovoltaic modules, MPPT, photovoltaic characteristics and connections, standards. The inverter (role, principle, characteristics and efficiency). Example of a photovoltaic installation.

Chapter 5: Other renewable energy sources
Renewable energy sources (solar energy, wind energy, hydroelectric power, biomass, geothermal energy). The different renewable energy sources around the world. Profitability.

Assessment method:

Exam: 100%

Bibliographic references :

1. *J. Vernier, Renewable Energies , PUF edition, 2012*
2. *E. Riolet , The mini-wind turbine, Eyrolles edition , 2010*
3. *A. Labouret and M. Villoz, Photovoltaic solar energy, Editions du Moniteur 2009*
4. *B. Fox, Wind Electric Power: Production, Forecasting and Grid Integration, Technical and Engineering Collection, Dunod/L'Usine Nouvelle 2015 (2nd edition)*
5. *A. Damien, Biomass energy: Definitions, resources and transformation methods, Technical and Engineering Collection, Dunod/L'Usine Nouvelle 2013 (2nd edition)*
6. *A. Labouret, M. Villoz, Photovoltaic installations: Design and sizing of installations connected to the network, Technical and Engineering Collection, Dunod/Le Moniteur 2012 (5th edition)*
7. *http://www.cder.dz/spip.php?article1442*