SYLLABUS

1. Information regarding the programme

1.1 Higher education institution	**Babeş-Bolyai University**
1.2 Faculty	**Faculty of Mathematics and Computer Science**
1.3 Department	**Department of Computer Science**
1.4 Field of study	**Computers and Information Technology**
1.5 Study cycle	**Bachelor**
1.6 Study programme / Qualification	**Information Engineering**

2. Information regarding the discipline

2.1 Name of the discipline (en) (ro)	Design with microprocessors, Proiectare cu microprocesoare						
2.2 Course coordinator	Arthur Robert Tunyagi						
2.3 Seminar coordinator	Arthur Robert Tunyagi						
2.4. Year of study	**3**	2.5 Semester	**5**	2.6. Type of evaluation	**E**	2.7 Type of discipline	**Compulsory DD**
2.8 Code of the discipline		MLE5191					

3. Total estimated time (hours/semester of didactic activities)

3.1 Hours per week	4	Of which: 3.2 course	2	3.3 seminar/laboratory	2 LP
3.4 Total hours in the curriculum	56	Of which: 3.5 course	28	3.6 seminar/laboratory	28
Time allotment:					hours
Learning using manual, course support, bibliography, course notes					24
Additional documentation (in libraries, on electronic platforms, field documentation)					10
Preparation for seminars/labs, homework, papers, portfolios and essays					4
Tutorship					4
Evaluations					2
Other activities:					

3.7 Total individual study hours	44
3.8 Total hours per semester	100
3.9 Number of ECTS credits	4

4. Prerequisites (if necessary)

4.1. curriculum	• course of general electronics
4.2. competencies	• skills of C and C++ programming, basic analogue and digital

	electronics knowledge.

5. Conditions (if necessary)

5.1. for the course	• During the development of the timetable for the semester the laboratory activities will always be placed after the course activity of the same week. • blackboard, multimedia projector, internet access, 220V CA sockets
5.2. for the seminar /lab activities	• blackboard, multimedia projector, internet access, workplaces for students equipped with computers running on Windows and having installed the STM32CubeIDE, CubeMX and the CubeProgrammer software packages. Laboratory equipment set like: power supplies, oscilloscopes, mustimeters. Development boards like: NUCLEO-F446RE, STM32F407G-DISC1, breadboards with jumper wires, basic electronic lab toolsets (screwdrivers, pliers, tweezers, etc) basic electronic components.

6. Specific competencies acquired

Professional competencies	• Describing the structure and operation of hardware, software and communication components. • Explaining the role, interaction and operation of hardware, software and communication components. • Construction of hardware and software components of computing systems using design methods, languages, algorithms, data structures, protocols and technologies. • Metric based evaluation of functional and non-functional characteristics of computing systems. • Implementation of hardware, software and communication components. • Appropriate use of the operating principles of electronic devices and circuits, as well as methods of measuring electrical quantities. • Analysing, designing, executing and measuring of electronic circuits of low/ medium complexity. • Diagnosis / troubleshooting of electronic circuits and instruments. • Use of electronic tools to characterize and evaluate the performance of electronic circuits. • Designing electronic circuits of low / medium complexity and implementing them using CAD techniques.

Transversal competencies	• Honorable, responsible, ethical behavior, in the spirit of the law, to ensure the professional reputation. • Demonstrating initiative and pro-active behavior for updating professional, economical and organizational culture knowledge

7. Objectives of the discipline (outcome of the acquired competencies)

7.1 General objective of the discipline	• Familiarizing students with embedded systems. • Familiarizing students with hardware and firmware design specific to microcontroller systems. • Familiarizing students with the communication standards used in the field.
7.2 Specific objective of the discipline	• Building the way of thinking of the students used in the development of embedded systems so that starting from a practical problem to reach the representation of the corresponding hardware and firmware architecture. • Developing students' ability to understand an electronic schematic developed around a microcontroller.

8. Content

8.1 Course	Teaching methods	Remarks
1. Overview of embedded systems. The Corex ARM microcontrollers. STM32CubeIDE, CubeMX, CubeProgrammer environment. - Documents used: (datasheet, reference manual, programming manual)	oral presentation and exemplification	2h
2. Internal structure of the microcontroller. AHB / APB buses, address map, peripherals, RCC module	oral presentation and exemplification	2h
3. Clock system operation, general considerations about oscillators and PLL circuits, the connection between the consumption of CMOS circuits and the frequency of the oscillator.	oral presentation and exemplification	2h
4. GPIO Peripheral, Related Registers, Configuration Types (In / Out, Pull Up / Down, PushPull / OpenDrain, Alternative Functionalities) - GPIO Driver Development	oral presentation and exemplification	2h
5. The interrupt system, the NVIC controller and the EXTI controllers used at the registry level	oral presentation and exemplification	2h
6. Introduction to ST-specific HAL programming. comparison between GPIO usage and interrupts using the respective HAL-API and registry level programming	oral presentation and exemplification	2h
7. The DMA unit, GPIO operation and application using synchronous, asynchronous and DMA models.	oral presentation and exemplification	2h
8. Types of timers present in the STM32	oral presentation and	2h

microcontrollers. (basic timer, general purpose timer, advanced control timer)	exemplification	
9. Timers opperation modes (Output compare mode, PWM mode, Input capture mode, Encoder interface)	oral presentation and exemplification	2h
10. The ADC and DAC peripherals, the principle of operation and their usage in the 3 modes (synchronous, interrupt, DMA)	oral presentation and exemplification	2h
11. The USART peripheral, operating principle, its useage in all 3 modes, using the HAL-API programming	oral presentation and exemplification	2h
12. The I2C peripheral, operating principle, its useage in all 3 modes, using the HAL-API programming	oral presentation and exemplification	2h
13. The SPI and the I2S peripherals, operating principle, usage in the 3 modes of interraction.	oral presentation and exemplification	2h
14. The CAN peripheral, operating principle, its use in all 3 modes. applications using the HAL-API	oral presentation and exemplification	2h

Bibliography:

- Mastering STM32, Carmin Noviello
- ARM Microcontrollers Programming for Embedded Systems, Sever Spânulescu.
- The definitive guide to the ARM Cortex-M3, ARM Cortex-M4 processors, Joseph Yiu.
- www.st.com
- www.arm.com

8.2 Seminar / laboratory	Teaching methods	Remarks
1. Presentation of the equipment used in the laboratory. (power supply, oscilloscope, measuring device, breadboard, making simple circuits using breadboard)	Laboratory work	2h
2. Presentation of the electrical schematics of the development boards used in the lab. (the STM32F407 Disc1 and the Nucleus F446RE) - pin identification and understanding of components integrated on the boards, The on board ST Link debugger	Laboratory work	2h
3. Debugging methods using STM32CubeIDE, lighting external LEDs and reading the status of switches connected to the development board.	Laboratory work	2h
4. Carrying out a project using GPIO ports (digital clock with 7-segment display by multiplexing), using the system timer as the time base.	Laboratory work	2h
5. Applications with external interrupts. measuring the period of an external signal using the 1KHZ system timer.	Laboratory work	2h
6. Applications with GPIO ports and external interrupts via the HAL-API. (applications	Laboratory work	2h

using capacitive touch sensor, application using a PIR sensor)		
7. Using the DMA controller examples for GPIO prototypes. DMA transfer times	Laboratory work	2h
8. Applications using the timers, interrupts associated with timers.	Laboratory work	2h
9. Applications using timers. frequency generation, PWM signal generation, WS2812 LED control using PWM mode timers and the DMA controller.	Laboratory work	2h
10. Applications with the ADC converter (monitoring of multi-channel resistive sensors). applications with the DAC converter (waveform generation)	Laboratory work	2h
11. Development of a communication protocol between the embedded board and the computer.	Laboratory work	2h
12. Example of I2C communication protocol using the TSL25911 sensor.	Laboratory work	2h
13. Example of using the SPI communication. The LIS3DSH MEMS sensor.	Laboratory work	2h
14. Example of communication using the CAN protocol and the TJA1050 driver.	Laboratory work	2h

Bibliography:

- Mastering STM32, Carmin Noviello
- ARM Microcontrollers Programming for Embedded Systems, Sever Spânulescu.
- The definitive guide to the ARM Cortex-M3, ARM Cortex-M4 processors, Joseph Yiu.
- www.st.com
- www.arm.com

9. Corroborating the content of the discipline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

Nowadays the ARM Cortex series microcontrollers are widely used in various household and industry-oriented applications. The course was structured to provide the students with a solid foundation in the field of the embedded system design. While preparing the content of the course there were taken into account the needs of the labour market in Cluj Napoca and other similar cities from Romania. On the same time these topics are present in several Universities in Romania and abroad.

10. Evaluation

Type of activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Share in the grade (%)
10.4 Course	evaluation of the theoretical skills accumulated during the lectures.	multiple choice test	50%

10.5 Seminar/lab activities	evaluation of the practical skills accumulated during the laboratories works.	multiple choice test	50%

10.6 Minimum performance standards

- Understanding of the working principles of an embedded system developed around an ARM Cortex microcontroller.
- Understand the steps required to develop a firmware, as well as the error-finding part using specific software packages.
- Ability to design a minimal embedded system from both a hardware and firmware perspective.

Date	Signature of course coordinator	Signature of seminar coordinator
...11.05.2022......	Lect.dr Arthur Tunyagi.	Lect. dr. Arthur Tunyagi

Date of approval

24.05.2022

Signature of the head of department

Prof. dr. Laura Dioşan