Advanced Microcontroller Systems

1. **Course number and name:** 020SAMES4/020AMSES4 Advanced microcontroller systems

2. **Credits and contact hours:** 4 ECTS credits, 2x1:15 contact hours

3. **Name of course coordinator:** Jean Sawma

4. **Instructional materials:** Professor textbook and course material

5. **Specific course information**
   a. **Catalog description:**
   Introduction to embedded systems – Introduction to STM32 family of MCUs and STM32CubeIDE –Principles of schematic interpretation for embedded applications – Overview and practical exploration of MCU Peripherals: ADC, DAC, Advanced Timers, PWM, UART, I2C, SPI, DMA, SDIO, USB – Introduction to Real Time Operating System (RTOS) – Introduction to machine learning on MCUs and TinyML.

   b. **Prerequisites:** 020SMPES3/020MPSES3 Microprocessor Systems

   c. **Selected Elective** for CCE and EE students

6. **Specific goals for the course**
   a. **Specific outcomes of instruction:**
   - Introduce STM32 family of MCUs and STM32 development environment.
   - Analyze and understand schematic for embedded applications.
   - Comprehend functionality of advanced MCU peripherals.
   - Design embedded software based on advanced MCU peripherals.
   - Introduce RTOS and its advantages.
   - Introduce machine learning on embedded devices.

   b. **PI addressed by the course:**

PI	1.1	1.2	1.3	6.1	6.2	6.3	6.4	7.1
Covered	x	x	x	x	x	x	x	x
Assessed				x	x	x	x	

7. **Topics and approximate lecture hours**
   - Introduction to embedded systems (2 lectures)
   - Introduction to STM32 family of MCUs and STM32CubeIDE (2 lectures)
   - Principles of schematic interpretation for embedded applications (1 lecture)
   - Overview of MCU Peripherals: ADC, DAC, Advanced Timers, PWM, UART, I2C, SPI, DMA, SDIO, USB (6 lectures)
   - Introduction to Real Time Operating System (5 lectures)
   - Introduction to machine learning on MCUs and TinyML. (3 lectures)
   - Lab Sessions on practical exploration of MCU Peripherals (5 labs of 2.5 hours each):
     Labs 1 and 2: Capturing and digitally filtering analog signals and outputting the processed signals in real time using ADC, DAC, timers, and DMA peripherals.
     Lab 3: Implementing master-slave communication between MCUs utilizing SPI and I2C peripherals for controlling the PWM duty cycle on the slave microcontroller.
     Lab 4: Reading and writing files on a microSD card using the SDIO peripheral.
     Lab 5: Execution of two synchronized threads, employing queues for inter-task communication within FreeRTOS.