Warsaw University
of Technology

COURSE OFFERED IN THE DOCTORAL SCHOOL

Code of the course	4606-ES-0000000-0152	Name of the course	Polish	**Podstawy programowania STM32 z użyciem biblioteki HAL**	
			English	**Introduction to STM32 programming using HAL library**	
Type of the course	specialized				
Course coordinator	Bartłomiej Ufnalski, Ph.D., D.Sc.	Course teacher	Bartłomiej Ufnalski, Ph.D., D.Sc.		
Implementing unit	Faculty of Electrical Engineering	Scientific discipline/disciplines*	all		
Level of education	Doctoral studies	Semester	Winter/spring or fall		
Language of the course	English				
Type of assessment	formative	Number of hours in a semester	30	ECTS credits	2
Minimum number of participants	10	Maximum number of participants	10	Available for students (BSc, MSc)	No

Type of classes		Lecture	Auditory classes	Project classes	Laboratory	Seminar/Workshop
Number of hours	in a week	-	-	2	-	-
	in a semester	-	-	30	-	-

* does not apply to the Researcher's Workshop

1. Prerequisites

The workshop is addressed to researchers from various disciplines wanting to take the first steps toward programming STM32 microcontrollers using HAL (hardware abstraction layer) libraries. Warning! If you already program any microcontrollers, whether using a low layer or hardware abstraction layer, to develop embedded control systems needed for e.g. your prototype devices, the classes are definitely NOT for you – they are too basic. This workshop aims to provide an experiential introduction to microcontrollers for researchers who want to start the embedded journey because they feel that this could help them e.g. automate some aspects of their experiments.

It is assumed that participants are familiar with the fundamentals of programming languages and can write code. If the very basics of C, incl. the concept of pointers, is something new to you, it is highly recommended to master the basics of the syntax already before the first meeting.

It is further assumed that the participants have a basic understanding of electrical circuits. If you don't know why a LED needs a resistor connected in series, how to use a capacitor and a resistor to build a low pass filter, or what is the purpose of pull-up and pull-down resistors, you should consider taking a relevant course, e.g. https://forbot.pl/blog/kurs-elektroniki-dla-poczatkujacych-id5151, prior to our workshop. Otherwise, you will probably feel lost at our meetings. Before enrolling in the course, please make sure that breadboarding simple circuits is considered by you to be a great fun activity. It will be helpful also to know how to use a transistor as an ON/OFF switch.

If you are not sure if this is for you, please visit Control Engineering for Hobbyists available on WUT's MS Teams. The code is k9sq4u7. We are going to experiment with concepts mentioned in e.g. Challenge 11, 18, 29, 32, 40, 64 and/or 80. Please take Challenge 85 (NPN and PNP transistors) and Challenge 89 (N-MOSFET and P-MOSFET as a switch) to warm up before the course. In case of any doubts, please contact me directly via chat on MS Teams.

2. Course objectives

The main objective is to get familiar with the basics of STM32 programming using STM32CubeIDE environment, HAL (hardware abstraction layer) libraries, and Nucleo boards.

3. Course content (separate for each type of class)

Lecture
-

Laboratory/Workshop
1. How to organize your own workshop at home. 2. Introduction to STM32CubeIDE. 3. Microcontroller (typical) peripherals. 4. Documentation, a debugger, and a logic analyzer are our best friends. 5. Blocking vs. non-blocking code (software timers, interrupts, DMA). 6. Common communication protocols and buses: a) UART (e.g. NEO-6M GPS module), b) I2C [TWI] (e.g. SH1106 OLED display or MPU6050 accelerometer and gyroscope), c) SPI (e.g. some RFID tag readers, MicroSD card readers, TFT LCD displays), d) CAN [TWAI] (e.g. we will communicate two or more STM32s). 7. “Non-standard/custom” [often one-way] communication (e.g. HX711 load cell amplifier, WS2812 LED). 8. Incremental encoders (QEP). 9. PWM (e.g. LED brightness, DC motor speed). 10. Feedback control (e.g. we will experiment with hysteresis or proportional-integral control of the temperature). 11. What next?

4. Learning outcomes			
Type of learning outcomes	Learning outcomes description	Reference to the learning outcomes of the WUT DS	Learning outcomes verification methods*
Knowledge			
K01	Knowledge of selected microcontroller peripherals employed to communicate with sensors and actuators	SD_W3	active participation during classes
K02			
K03			
Skills			
S01	Ability to use microcontrollers to automate some aspects of experiments	SD_U1	active participation during classes
S02	Ability to perceive microcontrollers as a tool not only for the control engineers, electronics engineers, power electronics engineers, or embedded software engineers	SD_U1	active participation during classes
Social competences			
SC01	Nourishing the DIY attitude	SD_K4	active participation during classes

*Allowed learning outcomes verification methods: exam; oral exam; written test; oral test; project evaluation; report evaluation; presentation evaluation; active participation during classes; homework; tests

5. Assessment criteria

The assessment process is based on active participation during workshops and active problem solving at home, which includes analysis of existing pieces of embedded code, developing your own code (at the level of tens of lines of code), debugging it, and running it on an STM32 microcontroller.

6. Literature

Exemplary challenges to be completed:

[1] https://www.youtube.com/@Dronebotworkshop

[2] http://ufnalski.edu.pl/lectures/2022_summer_school/

What next for your home lab:

[1] https://www.youtube.com/@ControllersTech

[2] https://www.youtube.com/@AndreasSpiess

[3] https://www.youtube.com/@greatscottlab

and many more similar ones.

7. Ph.D. student's workload necessary to achieve the learning outcomes**

No.	Description	Number of hours
1	Hours of scheduled instruction given by the academic teacher in the classroom	30
2	Hours of consultations with the academic teacher, exams, tests, etc.	-
3	Amount of time devoted to the preparation for classes, preparation of presentations, reports, projects, homework	30
4	Amount of time devoted to the preparation for exams, tests, assessments	-
	Total number of hours	**60**
	ECTS credits	**2**

** 1 ECTS = 25-30 hours of the Ph.D. student's work (2 ECTS = 60 hours; 4 ECTS = 110 hours, etc.)

8. Additional information

Number of ECTS credits for classes requiring the direct participation of academic teachers	
Number of ECTS credits earned by a student in a practical course	