Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Department of Computer Engineering

Intelligent (Real Time) embedded systems
Working program of the academic discipline (Syllabus)

Details of the academic discipline

Level of higher education	*First (bachelor's degree)*
Branch of knowledge	*12 Information technologies*
Specialty	*123 Computer Engineering*
Educational program	*Computer Engineering*
Discipline status	*Selective*
Form of education	*daytime*
Year of education, semester	*4 year (2 semester)*
Scope of the discipline	*4 credits, 120 hours* *Lectures 18 (36 hours), Laboratory 9 (18 hours)*
Semester control/ control measures	*Test*
Lessons schedule	*According to the schedule for the spring semester of the current academic year at http://rozklad.kpi.ua*
Language of teaching	*Ukrainian*
Information about head of the course / teachers	*Lecturer: Kaplunov Artem Volodymyrovych, art.kaplunov@gmail.com* *Laboratory: Kaplunov Artem Volodymyrovych, art.kaplunov@gmail.com*
Placement of the course	*Course code: qbqro7w*

Program of the Academic Discipline

1. Description of the educational discipline, its purpose, subject of study and learning outcomes

The purpose of the discipline "Intelligent Systems Development Technologies" is to study the theoretical foundations, functional capabilities, and principles of interaction of the main components of Real Time systems. The subject of study of the credit module "Intelligent Embedded (Real Time) Systems" is the architecture and functionality of modern intelligent embedded systems; input/output systems and data transfer and storage facilities and mechanisms; familiarity with languages and features of programming for Real Time systems.

Studying the discipline is aimed at students mastering the following professional competencies:

- knowledge of fundamental concepts, paradigms and basic principles of functioning of intelligent embedded systems;
- the ability to use theoretical, logical and arithmetic foundations for the development of drivers and software for Real Time systems and the ability to apply them in solving professional tasks;
- the ability to develop drivers for individual components of intelligent embedded systems, including using modern design automation systems;
- ability to develop and use drivers of architecture-dependent elements based on knowledge of general principles of organization and functioning of Real Time systems;

- the ability to participate in team work on the design of drivers and software for intelligent embedded systems;
- the ability to form and provide requirements for the reliability of Real Time systems in accordance with customer requirements, specifications and standards.
-

The discipline enhances the following general and professional competencies:

- ЗК2 - ability to learn and acquire modern knowledge;
- ФК1 - ability to apply the legislative and regulatory framework, as well as national and international requirements, practices and standards in order to carry out professional activities in the field of computer engineering.
- ФК2 - ability to use modern methods and programming languages for development of algorithms and software.
- ФК5 - ability to use the tools and systems of design automation to development of components of computer systems and networks, Internet applications, cyber-physical systems, etc.
- ФК9 - ability to systematically support, use, adapt and operate existing information technologies and systems.
- ФК13 - ability to solve problems in the field of computer and information technologies, determine the limitations of these technologies.
- ФК14 - ability to design systems and their components taking into account all their aspects. life cycle and task, including creation, configuration, operation, maintenance and disposal.
- ФК16 - ability to design, implement and maintain the high-performance parallel and distributed computer systems and their components using FPGAs, modules and CAD systems.

In accordance with the above, strengthened general and professional competencies will provide the following learning outcomes.

2. Pre-requisites and post-requisites of the discipline (place in the structural and logical scheme of training according to the relevant educational program)

When studying the discipline "Technology of development of intelligent systems", it is advisable to use the knowledge obtained during the study of previous disciplines: "Introduction to the Linux operating system", "Computer logic", "Theory of electric circuits and signals", "Computer electronics", " Computer circuit engineering", "Computer architecture. Part 1. Control and arithmetic devices", "Computer architecture. Part 2. Processors", "Programming", "Algorithms and data structures", "Programming technologies in C for embedded systems" (selective), "Foreign language".

The discipline is basic for the courses: "Intelligent systems design technologies" (elective), "Technologies of testing (QA) of embedded systems" (elective), "Infrastructure IT project management" (elective), "Computer modeling", "Organization of computing processes ", "Computer architecture. Part 3. Microprocessor devices", "Computer architecture. Course work", "System programming", "Computer networks", "Computer systems", "Technologies of designing computer systems" (selective), "Research and design of parallel systems" (selective), "Technologies of parallel programming computer systems" (selective).

3. The structure of the credit module

Introduction. Overview of Real Time systems
Chapter 1. Typical programs for Real Time systems
Topic 1.1. Sample data systems.
Topic 1.2. Examples of management hierarchy.
Topic 1.3. Bandwidth requirements.
Chapter 2. Hard vs. Soft in real-time systems
Topic 2.1. Time limits for Hard and Soft.
Topic 2.2. The reasons for the need for a time guarantee.
Chapter 3. Reference model of real-time systems

Topic 3.1. Processors and resources.
Topic 3.2. Time parameters for REAL-TIME load.
Topic 3.3. Fixed, Slow and Sporadic Release Times.
Chapter 4. Commonly used real-time scheduling approaches
Topic 4.1. Approach "by the clock".
Topic 4.2. Priority approach.
Topic 4.3. Dynamic vs Static systems.
Topic 4.4. Optimality of EDF and LST algorithms.
Chapter 5. Clock-driven scheduling
Topic 5.1. A static timer-driven scheduler.
Topic 5.2. General structure of cyclic schedulers.
Topic 5.3. Improving the average response time of aperiodic tasks.
Chapter 6. Priority planning of periodic tasks
Topic 6.1. Fixed-priority vs Dynamic-priority algorithms.
Topic 6.2. Fast-monotonic and deadline-monotonic algorithms.
Topic 6.3. Well-known dynamic algorithms.
Chapter 7. Scheduling non-periodic and sporadic tasks in priority-driven systems
Topic 7.1. Goals, correctness and optimality.
Topic 7.2. Alternative approaches.
Topic 7.3. Algorithm of preventive weighted "fair" queue.
Chapter 8. Resources and resource access control.
Topic 8.1. Assumptions about resources and their use.
Topic 8.2. Application of mutual exclusion and critical sections.
Topic 8.3. Resource conflicts and deadlocks.

4. Educational resources and materials

4.1. Basic literature

1. Architecture of computers 3. Microprocessor tools. Part 2: Programming for STM32 microcontrollers. Study guide for degree candidates: Laboratory workshop [Electronic resource] : study guide for the student's educational program "Computer Systems and Networks" in specialty 123 "Computer Engineering" / I. A. Klymenko, V. V. A. Taraniuk, V. Tkachenko, A. V. Kaplunov; Igor Sikorsky Kyiv Polytechnic Institute - Electronic text data (). - Kyiv : Igor Sikorsky Kyiv Polytechnic Institute, 2022. - 125 p.
2. Testing and quality control (QA) of embedded systems [Electronic resource]: tutorial for bachelor's study degree applicants under the educational program "Computer systems and networks" specialty 123 "Computer engineering" / I. A. Klymenko,V. A. Taraniuk, V. V. Tkachenko, O.O. Pysarchuk; Igor Sikorsky KPI. – Electronic text data (1 file: XX MB). – Kyiv: Igor Sikorsky KPI, 2022. – 58 p.
3. Architecture of computers 3. Microprocessor tools. Part 1: Programming for the Cortex M4 processor. Study guide for degree candidates: Laboratory workshop [Electronic resource] : study guide for the student's educational program "Computer Systems and Networks" in specialty 123 "Computer Engineering" / I. A. Klymenko, V. V. Tkachenko, A. V. Kaplunov; Igor Sikorsky Kyiv Polytechnic Institute - Electronic text data (). - Kyiv : Igor Sikorsky Kyiv Polytechnic Institute, 2022. - 50 p.
4. FPGA Programming Technologies : Laboratory workshop [Electronic resource]: training . help _ for studies _ of the educational program "Computer systems and networks" by specialty 123 "Computer engineering" / O. A. Verba, V. I. Zhabin , I. A. Klymenko, V. V. Tkachenko; KPI named after Igor Sikorsky. – Electronic text data (1 file: 8.64 MB). – Kyiv: KPI named after Igor Sikorskyi, 2019. – 110 p.

4.2. Additional literature

5. Hermann Kopetz (2011) Real-Time Systems: Design Principles for Distributed Embedded Applications (Real-Time Systems Series), 2nd ed. Springer.
6. Maryline Chetto (2014) Real-time Systems Scheduling 1: Fundamentals. Wiley-ISTE.
7. JANE W. S. LIU (2000) Real-Time Systems. PEARSON INDIA
8. Rob Williams (2006) Real-Time Systems Development. Butterworth-Heinemann

9. Seyed Morteza Babamir (2012) Real-Time Systems, Architecture, Scheduling, and Application. IN-TECH
10. Rajib Mall (2006) Real-Time Systems: Theory and Practice, 1nd ed. Pearson Education India.
11. C. M. Krishna (1997) Real-Time Systems. McGraw Hill Higher Education.

4.3. Information resources

12. Course of video lectures - on the distance learning platform "Sikorsky" in the Google Workspace for Education environment: https://classroom.google.com/c/NTI1OTI0NTcxNzY4?cjc=qbqro7w.

5. Laboratory work

The purpose of the laboratory work is to acquire the skills and abilities to apply in practice the principles of designing and developing software for Real Time systems and their individual functional nodes. To perform laboratory work, emulators of Linux-oriented architectures (QEMU ARM), microcontroller programming tools (vim, Eclipse, Cube IDE) and hardware (Raspbery Pi, BeagleBone Black, STM StarterKit GlobalLogic) are used.

Topics of laboratory works:

Laboratory work 1. Development of drivers for GPIO. Connecting the LEDs on the STM32F4 board.
Laboratory work 2. Connecting the matrix keyboard to the STM32F4 DISCOVERY board.
Laboratory work 3. Interaction with the virtual COM port on the STM32F4 DISCOVERY board.
Laboratory work 4. Development of drivers for character displays. Working with the display on the STM32F4 Discovery board.
Laboratory work 5. Working with an external CS43L22 DAC amplifier on the STM32F4 DISCOVERY board.
Laboratory work 6. Development of own project. An intelligent embedded system using an accelerometer on the STM32F4 DISCOVERY board.

6. Students' self-study work

Types of independent work:
- preparation for classroom classes (0.5 hours x 18 lectures = 9 hours);
- preparation for express tests (recommended 1-2 hours x 2 tests = 2-4 hours)
- preparation and processing of calculations based on primary data obtained in laboratory classes, performance of laboratory work, solution of problems, posting of results on GitLab (recommended 2-4 hours x 6 laboratory works = 12-24 hours);
- execution of modular control work (2 MKR x 4-8 hours = 8-16 hours).

– Policy and control

5. Policy of academic discipline (educational component)

Deadlines are set for the performance of laboratory work and modular control work.

Performance of laboratory work outside of the established deadlines is accompanied by penalty points, which are deducted from the grade for the protocol. Modular control work is not accepted beyond the set time.

Penalty points are issued for: untimely submission of laboratory work. The number of penalty points is no more than 10.

Bonus points are awarded for: active participation in lectures; completing current homework, keeping a summary, preparing a message with a presentation on one of the topics of the SRS discipline, etc. The number of bonus points is no more than 10.

Some lecture topics are accompanied by short express tests (for 15 minutes), which include the material of the studied topic and questions that are asked for independent study. The points obtained for the test are included in the semester rating. Current tests are not retaken.

The performance of each laboratory work is preceded by the completion of an individual task and its preparation in the form of a protocol. A student who came to class without a completed protocol is not allowed to do laboratory work. In the first stage, the student defends the results obtained during the performance of an individual task for laboratory work, in the second stage - defends the theory through an oral survey or test. Most of the laboratory works are accompanied by tests to evaluate the studied theoretical and practical material for the laboratory work. The points obtained for the performance of laboratory work, for the test and for the protocol are included in the assessment for the laboratory work. Testing is carried out in a laboratory session after checking the results of laboratory work. A student who has not completed the individual task before the laboratory work and the test is not admitted.

Performance of laboratory work is mandatory for admission to semester control. The condition of admission to the semester control is the enrollment of all laboratory works and a starting rating of at least 30 points.

A modular test is written during a lecture session without the use of aids (mobile phones, tablets, etc.); the result is forwarded to the corresponding Google Drive directory via a Google form.

The modular control paper is not rewritten in case of a negative grade, a negative grade for the MCR (less than 9 points (<60%)) is equal to 0 points, in this case the modular control work is not counted.

The grade that a student can receive for each laboratory work and for each modular control work is given in table 1 of semester work evaluations, chapter 8 of the syllabus.

Thus, the minimum grade that a student must receive for admission to the semester exam is 60 points, the maximum is 100 points for the completion of all current works for the semester.

Applicants who have fulfilled all admission requirements (completed all laboratory work) and have a rating of less than 60 points, as well as applicants who wish to improve their rating, have the opportunity to pass a semester test in the form of a credit test at the last class on the schedule .

In the case of performance of credit control work, the rating is defined as the sum of points for credit control work and points for individual semester tasks.

The individual work of the student related to the performance of laboratory work is included in the individual semester tasks. The maximum number of points for individual work per semester is 60 points. The maximum mark for the test is 40 points. In this way, the applicant has the opportunity to increase his rating by writing a final test and adding additional points to the number of points received during the semester for individual semester work.

After completion of the credit control work, if the grade for the credit control work is higher than the rating, the applicant receives a grade based on the results of the credit control work. If the grade for the final test is lower than the rating, the applicant's previous rating (with the exception of points for the semester individual task) is canceled and he receives a grade based on the results of the final test. This option forms a responsible attitude of the applicant towards making a decision on the completion of the credit control work, forces him to critically assess the level of his training and carefully prepare for the credit.

6. Types of control and rating system for evaluating learning outcomes (RSE)

The student's semester rating from the credit module is calculated based on a 100-point scale. The rating consists of the points that the student receives for completing 8 laboratory works R_L, two modular control works R_{MCW} and expert tests R_{ET}.

The maximum number of points for laboratory work is 60 points, i.e $R_L = 60$.

The generalized criteria for evaluating laboratory work are as follows:

- the timeliness of the preparation of the protocol for the laboratory session, completeness of the theoretical or practical task in the protocol, the protocol is posted on GitLab on time;
- correct functioning of the developed models on software or hardware, demonstration of own repository on GitLab with laboratory work materials and availability of commits;
- a survey on the subject of laboratory work for crediting the practical part of the work, protection of the results obtained in the work, answers to additional theoretical questions of the teacher, completeness of the report/protocol on the work on GitLab.

A detailed approach to the assessment of each laboratory work is given in Table 1.

Table 1. Details of the evaluation of each laboratory work

The name of the class	Form of control	Scores	Admission to the exam by automatic evaluation	Total points
Laboratory work 1.	Entrance test	4	2	4
Laboratory work 2	Completing the task	3	5	8
	Polling	3		
	Protocol on GitLab	2		
Laboratory work 3	Completing the task	3	5	8
	Polling on QA	3		
	Protocol on GitLab	2		
Laboratory work 4	Completing the task	3	5	8
	Polling on QA	3		
	Protocol on GitLab	2		
Laboratory work 5	Completing the task	3	5	8
	Polling	3		
	Protocol on GitLab	2		
Laboratory work 6	Completing the task	3	5	8
	Polling	3		
	Protocol on GitLab / demonstration	2		
Laboratory work 7	Completing the task	3	5	8
	Polling	3		
	Protocol on GitLab / demonstration	2		
Laboratory work 8	Completing the task	3	5	8
	Polling	3		
	Protocol on GitLab / demonstration	2		
Number of points for individual work				60
Express tests at lectures	2 x 5	10	5	10
Modular control work	MCW1 (Тест)	15	9	15
	MCW2	15	9	15
Total points		100	60	100

The maximum number of points per MCW R_{MCW} = 2x15 = 30 points.

MCW1 is conducted in the form of automated testing on the Google Workspace for Education platform. The test consists of 60 questions R_{MCW_2} = 0,25x60 = 15 points

Modular control work MCW2 is performed independently according to an individual task. MCW2 assessment criteria at four levels:

- correct and meaningful answer with explanations in the terms of the subject area: 13 - 15 points;
- correct answer, incomplete explanations: 11 - 12 points;
- the answer contains errors: 9 - 10 points;
- the answer contains significant errors, there are no explanations: 4-8 points;
- no answer: 0 points.

The score for MCW2 is reduced by:

- incorrect registration;
- lack of comments in meaningful terms;

– lack of explanations during calculations.

The maximum number of points for express tests is 10 points, tests are conducted during lectures in the form of automated testing on the Google Workspace for Education platform.

The maximum number of points for the credit control work is R_T = 40 points.

The credit control work is conducted in the form of automated testing on the Google Workspace for Education / moodle platform, consisting of selected questions that were during the semester in MCW, express tests, and defenses of laboratory works. The maximum score for the credit control work R_T = 40 points.

Calendar certification of students (for 8 and 14 weeks of semesters) in the discipline is carried out according to the value of the student's current rating at the time of certification. If the value of this rating is at least 50% of the maximum possible at the time of certification, the student is considered certified. Otherwise, the attestation information is marked as "uncertified".

The number of points a student receives per semester is determined by the formula

$$\mathbf{R = R_L + R_{MCW} + R_{ET}.}$$

The maximum number of points per semester does not exceed RS = 100.

Taking into account the received sum of points, the final grade is determined according to table 3.

If a student writes a test paper, the number of points the student receives per semester is determined by the formula

$$\mathbf{R = R_{IP} + R_T}$$

where, $\mathbf{R_{IP} = R_L}$.

The maximum number of points per semester does not exceed R = 100.

Taking into account the received sum of points, the final grade is determined by table 3.

Table 2. Determination of the semester grade

Scores	*Rating*
100-95	Perfectly
94-85	Very good
84-75	Good
74-65	Satisfactorily
64-60	Enough
Less than 60	Unsatisfactorily
Admission conditions not met	Not allowed

Working program of the academic discipline (syllabus)
Made by, Ph.D., associate professor, professor of the department of CE Klymenko Iryna Anatoliivna, assistant of the department of CE, assistant, Kaplunov Artem Volodymyrovych.

Approved by the Department of Computing Engineering (Protocol No.10 dated 25.05.2022 p.)

Agreed by the methodical commission of FICT (protocol No.10 dated 09.06.2022 p.)