Національний технічний університет «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»

Емблема кафедри (за наявності)

Кафедра обчислювальної техніки

Computer architecture.Part-1
Arithmetic and control devices
Working program of the credit module (Syllabus)

Requisites of the academic discipline

Level of higher education	*First (Bachelor's degree)*
Branch of knowledge	*12 Information technology*
Specialty	*123 Computer engineering*
Educational program	*Computer systems and networks*
The status of the discipline	*Normative*
Form of education	*Full-time (daytime) education*
Year of training, semester	*2 course, spring semester*
Scope of discipline	*5,5 credits, 165 hours / Lectures -36 hours, laboratory - 18 hours, individual work of students - 111 hours*
Semester control/ control measures	*Exam, calendar control*
Lessons schedule	*According to the schedule for the spring semester of the current academic year at the address* rozklad.kpi.ua
Language of instruction	*English*
Information about the course leader / teachers	*Lectures:* Dr. Sci. (Engin.), professor *Zhabin Valerii Ivanovych* viz.kpi@gmail.com *Laboratory: phD, Docent Verba Oleksandr Andriiovych,* olverba@gmail.com
Course placement	*Lecture materials*: https://bbb.comsys.kpi.ua/b/

Program of the academic discipline

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

Credit module "Computer architecture.Part-1. Arithmetic and control devices" is the first software component of the "Computer Architecture" discipline. The study of this credit module allows students to develop competencies necessary for solving practical problems of professional activity, regarding the organization of computing processes in computers.

The purpose and main tasks of the credit module are to study the architecture of arithmetic and control tools of computers for various purposes, the basics of the organization of computing processes, ways of distributing functions between software, firmware and hardware tools, the principles and methods of their interaction, the functional hierarchy of tools, methods of evaluating their technical economic indicators**.**

The subject is mathematical, algorithmic and hardware methods of data processing, construction of arithmetic and control devices in various elementary bases, their modeling, research of the main characteristics and optimal selection of implementation methods.

As a result of studying the discipline, the student should get the following.

Knowledge:

- trends in the development of science and technology in the field of computer engineering;
- interrelationship of sections of the discipline and their connection with other disciplines;
- basic terms and definitions in the field of computer data processing;
- principles of software, firmware and hardware management of computing processes;
- principles of building the architecture of computers for various purposes;
- types of command systems, ways of addressing operands, data structure organization;
- principles of construction of arithmetic and control devices;
- principles of software management for the organization of computing processes in a computer, computer characteristics at the architectural and structural levels;
- knowledge of command systems, ways of addressing operands, organization of data structure;
- methods of studying the characteristics of computer systems;
- comparative analysis of research results and selection of the optimal result.

Skills:

- use formal languages for describing hardware and software;
- develop the architecture of arithmetic and control devices with distributed and centralized logic;
- develop the architecture of control devices with rigid and flexible logic;
- develop command systems, formats and structure of commands, ways of addressing commands and operands;
- develop microalgorithms and microprograms for the implementation of various commands, determine the effectiveness of the selected options;
- apply modern means of increasing the productivity, reliability and functionality of computing and control tools;
- use simulation and other types of modeling to study the characteristics of system components at various stages of design;
- perform the calculations necessary to determine the effectiveness of the decisions made.

Experience:

- formulation of practical problems in terms of hardware, microprogram and software description languages for data processing;
- choosing rational options for solving data processing problems;
- set tasks correctly, give a comparative description of various solutions at the design stages of digital devices;
- defend the adopted technical decision in a professional discussion;
- conduct an objective analysis of the effectiveness of adopted technical solutions;
- apply methods of avoiding failures in digital circuits;

- using large integrated circuits (ICs), including programmable ones, to build computing devices.

The credit module provides the **following competencies and program results** of the educational and practical program of the first (bachelor) level of higher education (EPP): ЗК2, ФК1, ФК5, ФК13, ФК14, ФК16, ПРН1, ПРН3, ПРН4, ПРН5, ПРН7, ПРН13, ПРН15, ПРН22.

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 content of the credit module "Computer Architecture-1. Arithmetic and control devices" it is advisable to use the knowledge obtained during the study of previous disciplines: ЗО10 "Discrete Math", ЗО14 "Computer electronics", ПО1"Computer logic".

Credit module "Computer architecture-1. Arithmetic and control devices" allows students to more productively master the knowledge and skills of the disciplines taught after this module, namely: ПО6 "Computer architecture", ПО11 "Computer systems", ПО19 "Computer circuit engineering", ПО13 "Algorithms and calculation methods".

3. Credit module content

Chapter 1. Introduction.

Topic 1.1. Substantive provisions.

Chapter 2. Introduction to computer architecture.

Topic 2.1. Typical computer architecture.

Chapter 3. Architecture of arithmetic devices and control devices.

Topic 3.1. Fundamentals of digital information processing in arithmetic devices.

Topic 3.2. Management of computational processes at the level of operations.

Chapter 4. Organization of program management.

Topic 4.1. Command systems. Management of their implementation.

Topic 4.2. Stages of execution of commands of various purposes.

Chapter 5. Fundamentals of the organization of firmware management of data conversion in computers.

Topic 5.1. Architecture of computers with firmware management.

Topic 5.2. Microprograms for executing commands for various purposes.

4. Educational materials and resources

Basic:

1. Zhabin V.I. Arithmetic and control devices of digital computers. Study guide / V. I. Zhabin, I. A. Zhukov, I. A. Klymenko, S. G. Stirenko. - K.: VEK+, 2008. - 176 p. (the seal of the Ministry of Education and Culture of Ukraine). https://www.twirpx.com/file/1797051/; https://campus.kpi.ua/tutor/index.php?mode=mob&show&irid=220770.

2. Computer architecture 1. Arithmetic and control devices: Workshop [Electronic resource]: teaching. manual for students specialty 123 "Computer engineering" / V. I. Zhabin, I. A. Klymenko, V. V. Tkachenko. – Electronic text data. – Kyiv: KPI named after Igor Sikorskyi, 2018. – 53 p. (The letterhead was provided by the Methodical Council of the Igor Sikorskyi KPI, protocol No. 3 dated November 15, 2018).https://ela.kpi.ua/handle/123456789/29525.

3. Computer architecture. Arithmetic and control devices. Practicum [Electronic resource]: education. manual for bachelor's degree holders in the educational program "Computer systems and networks" specialty 123 "Computer engineering" / Editors: V. I. Zhabin, O. A. Verba; KPI named after Igor Sikorsky. – Electronic text data (1 file: 3.03 MB). – Kyiv: KPI named after Igor Sikorskyi, 2022. – 80 p. (The letterhead was provided by the Methodical Council of the Igor Sikorskyi KPI, protocol No. 1 dated September 2, 2022). https://ela.kpi.ua/handle/123456789/50132.

4. Zhabin V.I. Microprocessor systems: Education. Guide / V. I. Zhabin, I. A. Zhukov, V. V. Tkachenko, I. A. Klymenko. - K.: Publishing House "SPD Guralnyk O.Yu.", 2009. - 492 p. (Vulture of the Ministry of Education and Culture of Ukraine).https://www.twirpx.com/file/1575788/.

5. Matvienko M.P. Computer architecture. Study guide / M.P. Matvienko, V.P. Rosen, O.M. Zakladnyi. - K.: Pub. Lira-K, 2013. 264 p.

6. Melnyk A.O. Computer architecture / Melnyk A.O. – Lutsk; Volyn regional printing house, 2008. – 470p.

Additional:

7 . Zhabin V.I. Digital machines. Workshop / V.I.Zhabin, V.V.Tkachenko. - K.: VEK+, 2004 - 160 p.

8. Pavlyuk M.F. Computer architecture. Course of lectures / M.F. Pavlyuk, L.M. Demchuk. — Ivano-Frankivsk: Play, 2012. — 198 p. https://www.twirpx.com/file/2023192/.

9. Karachka A.F. Computer architecture / A.F. Karachka, P.R. Strubytskyi, O.I. Dudko. – Education. manual. – Ternopil, 2006. – 152 p. https://www.twirpx.com/file/353857/.

10. Tararak V.D. Architecture of computer systems Study guide. — Zhytomyr: Zhdtu, 2018. — 383 p. https://www.twirpx.com/file/2720671/.

Information resources:

11.Computer hardware. https://studfile.net/preview/5342216/.

12.Computer architecture. http://www.dut.edu.ua/ru/lib/1/category/1212.

Equipment needed for classes

Lecture classes are held in a classroom equipped with a projector, laboratory classes are held in a computer laboratory.

Educational content

Methods of mastering an educational discipline (educational component)

4. ***The structure of the credit module***

Names of sections, topics	Number of hours					
	In total	lectures	practical	seminars	laboratory	individual work of student
1	2	3	4	5	6	7
Chapter 1. Introduction						
Topic 1.1. Substantive provisions.	7	2				5
Chapter 2. Introduction to computer architecture.						
Topic 2.1. Typical computer architecture.	10	2			2	6
Chapter 3. Architecture of arithmetic devices and control devices.						
Topic 3.1. Fundamentals of digital information processing in arithmetic devices.	32	12			2	18
Topic 3.2. Management of computational processes at the level of microoperations.	24	6			4	14
Chapter 4. Organization of program management.						
Topic 4.1. Command systems, management of their execution.	14	4			2	8
Topic 4.2. Stages of execution of commands of various purposes.	18	2			2	14
Chapter 5. Fundamentals of the organization of firmware management of data conversion in EOM.						
Topic 5.1. Computer architecture with firmware management.	14	2			4	8
Topic 5.2. Microprograms for executing commands for various purposes.	16	6			2	8
Preparation for the exam	30					30
Total per semester	165	36			18	111

5. ***Lecture classes***

№ з/п	Title and content of the lecture
1	Lecture 1. Purpose and classification of computers. Basic concepts and definitions. Hierarchical principle of computer hardware and software construction. Tasks for individual work of student: methods of determining computer performance. literature[5, c. 9-16; 6. c. 17-26].
2	Lecture 2. Von Neumann's architecture. Von Neumann's principles of building universal computers. Harvard Architecture. Characteristics of computer devices and nodes. Tasks: examples of non-Fonneiman architecture. literature[5, c. 12-16; 6, c. 28-31].
3	Lecture 3. Data formats in computers. Features of processing fixed and floating point numbers. Operating schemes, microoperations and microalgorithms. Tasks: non-traditional ways of presenting data. literature[1, c. 8-27; 5, c. 61-72; 6, c. 63-72].
4,5	Lecture 4, 5. Arithmetic-logic devices with distributed logic. Performing basic arithmetic operations in arithmetic-logic devices with distributed logic. Task: estimation of the execution time of operations in arithmetic-logic devices with distributed logic. literature[1, c. 29-42].
6	Lecture 6. Arithmetic-logic devices with concentrated logic. Performing basic arithmetic operations. Tasks: estimation of execution time of operations in arithmetic-logic devices with concentrated logic. literature[1, c 42-52].
7	Lecture 7. Performing fixed-point operations. Tasks: processing of number signs when performing shift operations. literature [1, c 20-24, c. 102-107; 6, c. 204-208].
8	Lecture 8. Processing of floating point data. Tasks on SRS: methods of rounding the result when performing floating-point operations. literature [1, c 20-24, c. 102-107; 6, c. 275-277].
9	Lecture 9. Functions and general organization of management in computers. Types of control devices. Implementation of control devices with hard logic. Tasks: to repeat the methods of synthesis of Milli and Moore automata. literature[1, c. 56-57; 6, c. 283-285].
10	Lecture 10. Microprogram control devices with flexible logic. Ways of addressing microcommands. Microcommand memory programming. Task: comparative assessment of the speed of control devices with rigid and flexible logic. literature[1, c. 58-68].
11	Lecture 11. Synthesis of control devices with forced, relative and functional addressing of microcommands. Task: assessment of the complexity of devices with different ways of addressing microcommands. literature[1, c. 68-84].
12	Lecture 12. Systems and classification of commands. Team structure and formats. Address space of computers. Tasks on SRS: characteristics of command systems for RISC and CISC computers. literature[4, c. 250-278; 6, c. 87-98, c. 105-121].
13	Lecture 13. Command execution management. Ways of addressing commands and operands. Tasks on SRS: alternative queues of commands in modern computers. literature[6, c. 107-117].
14	Lecture 14. Stages of execution of commands. Characteristics of the comparative effectiveness of computer command systems. Tasks on SRS: features of system command execution. literature[6, c. 102-105].

№ з/п	Title and content of the lecture
15	Lecture 15. Computer architecture with firmware control. Interpretation of commands in a computer with microprogram control. Micro command systems. Types of languages for describing microcommands and microprograms. literature[1, c. 56-76; 6, c. 138-150]. Task: Using microassembler to describe firmware.
16	Lecture 16. Microprograms for accessing data in RAM and ports of external devices. Microprograms for performing arithmetic and logical operations in ALP with general microoperations Tasks: algorithms for initializing ports of external devices. . literature[6, c. 140-150].
17	Lecture 17. Implementation of conditional and unconditional transitions in microprograms. Reference to microroutines. Nested microroutines. Tasks: the structure of the hardware stack for working with microroutines. literature [6, c. 90-95].
18	Lecture 18. Construction of microprograms for execution of data input-output commands. Tasks on SRS: microprograms for initializing ports of external devices to implement data exchange in interrupt mode. literature[6, c. 96-102].

6. Laboratory classes

The purpose of the laboratory work is to acquire skills and practical application of logical methods of analysis and synthesis of arithmetic devices. Laboratory classes can be performed both on specially created laboratory models and with the use of simulation systems on computers.

№ з/п	Name of laboratory work (computer workshop)	Number of classroom hours
1	Laboratory work #1. Synthesis of arithmetic and logical devices with distributed logic. (Chapter 3, topic 3.1)	2
2	Laboratory work #2. Synthesis of microprogram control blocks. (Chapter 3, topic 3.2)	4
3	Software complex for performing laboratory work COMP_I_R. Micro assembler programming. (Chapter 3, topic 3.2; Chapter 4, topics 4.1, 4.2)	2
	Laboratory work #3. Data conversion in eom with firmware control. (Chapter 1, topic 1.2, Chapter 4; topics 4.1, 4.2)	4
4	Laboratory work #4. Information processing in a computer at the software and firmware levels. (Chapter 3, topic 3.1; Chapter 5, topic 5.1, 5.2)	4
5	Laboratory work #5. Formation of the system of processor commands, development of programs and microprograms for processing information in a computer. (Chapter 3, topic 3.1; Chapter 5, topic 5.1, 5.2)	2

	Total:	18

7. Individual work

№ з/п	The name of the topic submitted for independent processing	Number of hours
1	Topic 1.1. Substantive provisions.	5
2	Topic 2.1. Typical computer architecture.	6
3	Topic 3.1. Fundamentals of digital information processing in arithmetic devices.	18
4	Topic 3.2. Management of computational processes at the level of microoperations.	14
5	Topic 4.1. Command systems, management of their execution.	8
6	Topic 4.2. Stages of execution of commands of various purposes.	14
7	Topic 5.1. Computer architecture with firmware management.	8
8	Topic 5.2. Microprograms for executing commands for various purposes.	8
9	Preparation for the exam	30
10	Total in the semester	111

Independent work involves:

- *preparation for lectures;*
- *preparation for laboratory classes;*
- *preparation for the test*

Policy and control

8. Policy of academic discipline (educational component)

During credit module classes, students must adhere to certain disciplinary rules:

-it is forbidden to be late for classes;

- at the entrance of the teacher, as a sign of greeting, persons studying at KPI named after Igor Sikorsky should stand up;

- extraneous conversations or other noise that interferes with classes are not allowed;

- leaving the classroom during the lesson is allowed only with the teacher's permission.

- the use of mobile phones and other technical means is not allowed without the teacher's permission.

9. Types of control and rating system for evaluating learning outcomes (RSO)

Distribution of study time by types of classes and tasks in the discipline according to the working study plan. 165 hours and 5.5 credits are allocated to the credit module.

Academic	The number of hours according to the curriculum							
semester	total	lectures	Practical training	Laboratory classes	homework control work	modular control work	Individual work	Examination
1	165	36		18			111	Examination

9.1. The student's semester rating from the credit module is calculated based on a 100-point scale. The rating consists of points that the student receives for performing 5 laboratory works $R_Л$ and examination R_E.

9.2. The maximum number of points for each laboratory work is 12, i.e $R_Л$ =60 (12·5=60).

Points are awarded for:

- timeliness of preparation of the protocol for the laboratory session, completeness of the theoretical task: 0-1 points;

- written colloquium on the subject of laboratory work for admission to the practical part of the work: 0-3 points;

- the correctness of the functioning of the developed models on the software emulator (complex of programs for debugging models): 0-5 points,

- protection of the results obtained in the work, answers to the teacher's theoretical questions, completeness of the work protocol: 0-3 points.

9.3. The maximum number of points for the exam is equal R_E =40.

The examination ticket contains 4 tasks (one theoretical and three practical) on the subject of lectures and laboratory work performed during the semester. Each question is evaluated from 0 to 10 points.

Evaluation criteria for each question at four levels:

- correct and meaningful answer - 9-10 points;
- correct answer, incomplete explanations - 6-8 points;
- the answer contains errors - 3-5 points;
- there is no answer or the answer is incorrect - 0 points.

9.4. 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".

9.5. A prerequisite for a student's admission to the exam is the completion and defense of all laboratory work with a total of at least 30 points.

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

$$R_C = R_Л + R_E .$$

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

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

Determination of the grade on the university scale

R_C	**Rating**
95-100	Perfectly
85-94	Very well
75-84	Good
65-74	Satisfactorily
60-64	Enough
$R_C < 60$	Unsatisfactorily
Laboratory work not performed	Not allowed

Additional information on the discipline (educational component)

List of questions for independent preparation for the exam, semester control, colloquium.

Purpose and classification of computers.
Basic concepts of the hierarchical principle of computer hardware and software construction.
Von Neumann architecture. Von Neumann principles.
Purpose of the main devices of the computer.
Data formats in computers. Features of processing fixed and floating point numbers.
Operating schemes, microoperations and microalgorithms.
Arithmetic logic devices with distributed logic.
Arithmetic logic devices with concentrated logic.
Describe the four basic methods of multiplying numbers.
How to calculate the bit rate of operating device nodes?
Define the concepts: operation, microalgorithm, microoperation.
What is a microalgorithm operation?

Define the main purpose of an arithmetic logic device in a computer.
List the types of arithmetic and logical devices and their main differences.
Describe the main stages of designing an arithmetic logic device with distributed logic.
What does the operation flow diagram show?
What does the functional diagram of the device show?
What is the difference between functional and structural microalgorithms?
Write the expressions defining the laws of operation of Mile and Moore automata.
What is the difference between Mile and Moore machines?
Draw a generalized structural diagram of the control automaton.
Describe the main stages of designing a control automaton.
How to go from meaningful microalgorithm to coded microalgorithm?
How to build an automaton graph?
How is the state of the machine evaluated?
How to determine the required duration of control signals?
What determines the number of triggers needed to build the automaton's memory?
How to make a structural table of the automaton?
Draw a transition table for JK, RS, T and D flip-flops. Give their conventional graphic notation.
Is it possible to transition the automaton to a state that is not predicted by the graph when using flip-flops with an internal delay (triggers controlled by the level of signals)?
When is it possible to have false control signals (unpredicted by the automaton graph) and how is their duration determined?
Give ways to eliminate short-term false control signals.
What is the essence of the "anti-race" coding of automaton states?
How to ensure a drop of the control signal in the case when the operator vertex with this signal is covered by a "loop"?
How to determine the transition time of the automaton from one state to another?
Give the main methods of implementing the machine division operation (calculation of the root).
Specify the composition of the equipment necessary for the implementation of an arithmetic-logic device with distributed logic for performing the operation of machine division (calculation of the root).
Give a comparative characteristic of an arithmetic-logic device with distributed and concentrated logic.
Give the general structural and functional structure of a device with microprogram control for performing arithmetic operations, explain the general purpose of the microprogram control unit and arithmetic logic device.
Give the stages of construction of an arithmetic logic device with distributed logic.
List the classifications of the firmware control unit.
What is the difference between teams and microteams?
Explain what is meant by the principle of firmware management.
How to ensure the duration of microoperations in the asynchronous method of controlling the execution of microcode in the firmware control unit
Give the word format of the microcommands in the microprogram control block obtained during the laboratory work and explain the purpose of each of the zones.

Functions and general organization of management in computers. Types of control devices. Implementation of control devices with hard logic.
Microprogram control devices with flexible logic.
Ways of addressing microcommands.
Logical organization of microcommand memory.
Synthesis of control devices with forced, relative and functional addressing of microcommands.
Systems and classification of teams. Team structure and formats. The address space of the computer.
Command execution management. Ways of addressing commands and operands. Stages of execution of commands.
Characteristics of the comparative effectiveness of command systems.

Methodical recommendations for performing laboratory work

The stages of performing laboratory work cause special difficulties for students.

The design of computer devices should begin with the analysis of the algorithm for solving the given problem and the study of the computer architecture. Initially, the command system includes commands that provide data input and output, work with subprograms, program disclosure, and enable and disable interrupts. After that, the system of commands is distributed by commands that, together with the existing ones, ensure the implementation of the given algorithm. The format of teams and their marks is being developed. Based on the specified accuracy of calculations and other factors, the form and bit rate of data presentation is selected. The program model of the processor is compiled, the memory organization is specified, and functional microalgorithms for command execution are developed. The structural diagram of the processor is being developed, the architecture is being specified. Next, the issues of information exchange with external devices (interruption mode, direct access, software mode) are considered. Based on the analysis of the developed microalgorithms of the commands and the selected means of interaction with external devices (software, firmware, hardware), the structural diagram of the computer is clarified.

Each laboratory work must be preceded by independent preparation of students, during which they study in detail the description of practical work, relevant sections of the lecture notes and literary sources. In the process of preparation, a report on practical work is drawn up, in which all points of the theoretical task must be reflected, as well as tables, algorithms, schemes, etc. prepared for the experimental part of practical work. Before starting laboratory work, the results of training are checked by the teacher. During such an examination, the student must present the prepared report and answer the control questions.

Before the start of the next lesson in the laboratory, the student presents a fully prepared report on the previous work to the teacher. The report should contain brief theoretical information necessary for the performance of the task, answers to control questions, schemes, formulas, algorithms, tables, diagrams, graphs, program code, compiler reports obtained during the performance of the task and in the process of modeling and experimental research of the developed devices, as well as conclusions The student receives credit for the performance of practical work after an interview on the subject of the completed work.

Working program of the academic discipline (syllabus):

Folded, Dr. Sci. (Engin.), professor, Жабін Валерій Іванович.

Approved by the Department of Computing (Protocol No. 10 dated 05/25/2022).

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