GDAŃSK UNIVERSITY OF TECHNOLOGY

Subject card

| | | | |
|---|---|---|---|
| Subject name and code | Basic of Computer Systems Organization, PG_00047821 | | |
| Field of study | Biomedical Engineering | | |
| Date of commencement of studies | October 2024 | Academic year of realisation of subject | 2026/2027 |
| Education level | first-cycle studies | Subject group | Optional subject group<br>Subject group related to scientific research in the field of study |
| Mode of study | Full-time studies | Mode of delivery | at the university |
| Year of study | 3 | Language of instruction | Polish |
| Semester of study | 5 | ECTS credits | 3.0 |
| Learning profile | general academic profile | Assessment form | exam |
| Conducting unit | Department of Automatic Control -> Faculty of Electronics, Telecommunications and Informatics | | |
| Name and surname of lecturer (lecturers) | Subject supervisor | dr inż. Paweł Raczyński | |
| | Teachers | dr inż. Paweł Raczyński | |

**Lesson types and methods of instruction**

| Lesson type | Lecture | Tutorial | Laboratory | Project | Seminar | SUM |
|---|---|---|---|---|---|---|
| Number of study hours | 30.0 | 0.0 | 15.0 | 0.0 | 0.0 | 45 |

E-learning hours included: 0.0

**Learning activity and number of study hours**

| Learning activity | Participation in didactic classes included in study plan | Participation in consultation hours | Self-study | SUM |
|---|---|---|---|---|
| Number of study hours | 45 | 3.0 | 27.0 | 75 |

**Subject objectives**

The main aim of the subject is to gain knowledge about the most common computer systems organization and basic knowledge of computer system components and principles of their operation.

**Learning outcomes**

| Course outcome | Subject outcome | Method of verification |
|---|---|---|
| [K6_W04] knows and understands, to an advanced extent, the principles, methods and techniques of programming and the principles of computer software development or programming devices or controllers using microprocessors or programmable elements or systems specific to the field of study, and organisation of systems using computers or such devices | The student knows the rules for creating software for embedded systems with PCs. The student knows the rules for using the PC104, VME and Compact PCI buses. The student knows the rules of practical use of Linux, Windows and other operating systems. The student knows the techniques of input and output interface software. The student knows the techniques of creating real-time software. The student knows the rules for implementing the elements of self-diagnosis of embedded computer systems. | [SW1] Assessment of factual knowledge |
| [K6_U04] can apply knowledge of programming methods and techniques as well as select and apply appropriate programming methods and tools in computer software development or programming devices or controllers using microprocessors or programmable elements or systems specific to the field of study | The student knows how to create software for embedded systems with PCs. Student and knows how to use PC104, VME and Compact PCI buses in practice. The student knows how to use the operating systems Linux, Windows and others. The student knows how to use the input and output interface software techniques in practice. The student knows and is able to put into practice the techniques of creating real-time software. The student knows how to use the diagnostic elements of embedded computer systems in practice. | [SU4] Assessment of ability to use methods and tools<br>[SU1] Assessment of task fulfilment |

<table>
<tr><td>Subject contents</td><td>1. Lecture organization, credit rules, literature 2. Microprocessor Intel x86 architecture, general purpose registers, arithmetic and logic unit, flags 3. Addressing space, memory and input-output addressing, memory segmentation, addressing modes 4. Microprocessor programming model, instruction cycle 5. Instructions and techniques of data transfer, block data transfer 6. Arithmetic instructions, number formats, acting on multi precision numbers, floating point calculations - software emulation, using arithmetic coprocessor 7. Operations on bits and strings 8. Unconditional and conditional branch instructions, call and ret instruction, stack usage 9. Processor control organization, bus interface unit and instruction execute unit, instruction queue 10. Interrupt system, vectored interrupts, multi level interrupt service 11. Processor working in real and protected modes 12. Basics of programming in assembler, instruction mnemonics, variables, labels, directives, assembler syntax 13. Program assembly techniques, name dictionary, error reports, linking 14. Macroinstructions, subroutines, parameter passing into subroutines, stack frame 15. Memory models and its consequences, static and dynamic memory allocation techniques 16. Mixed programming, calling convention used in C and PASCAL languages 17. Input and output device standards, input and output devices service techniques 18. Parallel and serial data exchange, hardware support techniques 19. Hardware and software interrupts service techniques, programmable interrupt controller 20. Direct memory access (DMA), DMA controller, CPU-DMA interaction, data transfer programming and performance phase 21. Basic of x86-32 and x86-64 architecture, CISC and RISC processors 22. PC architecture 23. Mass data storage devices, hard disks, optic drives, FLASH memories 24. BIOS organization and BIOS service subroutines 25. User console, keyboard and mouse service techniques, data buffering methods 26. Screen service in text and graphic modes 27. Hardware interrupts service techniques in PC standard computers 28. Software interrupts, parameter passing into BIOS service routines 29. Real time clock and system clock 30. Operating system, organization, OS service functions 31. Introduction into embedded systems 32. Embedded systems based on PC standard computers 33. Modular computers based on PC104 standard bus 34. Modular computers based on VME standard bus 35. Modular computers based on COMPACT PCI standard bus 36. Controlled or monitoring object interface organization 37. Operating system in embedded systems: WINDOWS embedded, Linux, QNX 38. Embedded system software specificity 39. Object interface - device handlers 40. Hardware interrupt handling techniques: interrupt service routines, interrupt initiated task for service requests 41. Real time - techniques of implementation 42. Software handlers for standard communication interfaces 43. Dedicated software: mini kernel techniques, interrupt handling procedures techniques, software loop techniques 44. Software diagnostics 45. Embedded system examples</td></tr>
<tr><td>Prerequisites and co-requisites</td><td>No requirements</td></tr>
<tr><td>Assessment methods and criteria</td><td>
<table>
<tr><th>Subject passing criteria</th><th>Passing threshold</th><th>Percentage of the final grade</th></tr>
<tr><td>2 partial exams</td><td>51.0%</td><td>40.0%</td></tr>
<tr><td>Practical exercise</td><td>51.0%</td><td>60.0%</td></tr>
</table>
</td></tr>
<tr><td>Recommended reading</td><td>
<table>
<tr><td>Basic literature</td><td>A. Skorupski, Podstawy budowy i działania komputerów, WKŁ B. Zieliński, Układy mikroprocesorowe. Przykłady rozwiązań, Helion 2002 Katalogi, strony WWW i podręczniki firmowe. Metzger P. "Anatomia PC", HELION, 2008. Niederliński A. Mikroprocesory mikrokomputery mikrosystemy. WSiP 1988. W. Komorowski, Krótki kurs architektury i organizacji komputerów, Mikom 2004</td></tr>
<tr><td>Supplementary literature</td><td>No requirements</td></tr>
<tr><td>eResources addresses</td><td>Adresy na platformie eNauczanie:</td></tr>
</table>
</td></tr>
<tr><td>Example issues/ example questions/ tasks being completed</td><td></td></tr>
<tr><td>Work placement</td><td>Not applicable</td></tr>
</table>