LECTURE 15
How to prepare for the exam

• Exam will be held through MS Teams
• Examinee has to register for a term before the deadline through the ZMiTAC database
  – if you change your mind, withdraw your registration
  – without registration you will not be allowed to take the exam
  – registration is valid only if you are allowed to take an exam (it depends on passing tests in classes)
• Be prepared in good time, before the exam starts
  – late arrivals will not be allowed to participate
  – for each term all examinees will be verified with respect to the list of those who registered and who are allowed to take the exam
How to prepare for the exam

• Examinee has to have
  – ID with a photo – without valid ID you will be not allowed to take the exam
  – paper – at least 5 separate sheets (each task must be solved on a single, separate sheet of paper)
  – something to write with
• Examinee cannot use
  – any materials
• Electronic devices can be used only for the purpose of access and upload of student work
How to prepare for the exam

• Additional work at home corresponding to the general structure of the exam
  – use tasks from lectures, classes, didactic materials, academic books, etc.
  – time yourself

• Prepare all allowed elements before exam starts
  – own paper for all solutions needs to be signed
  – pens, pencils, etc. – it is best to have more than one
  – clear the space around you of all elements that are not on the list of allowed items
  – you need to be alone in a room (nobody else is allowed to be present in the same room during the exam)
  – make sure that there is no background noise
How to prepare for the exam

• Total exam time: 1/2 hour per each task
• Structure of the exam and grading system
  – 5 tasks
  – from each a grade is in the range from 0 till 5 (and treated as such for the calculated average, which also ranges from 0 to 5)
  – you pass the exam when the average is equal or higher than some threshold, which is dynamically set for each term (always at least 50 % of marks, but also taking into account the number of solved tasks)
  – you do NOT need to obtain a positive grade from each part to pass the exam
  – you always write all tasks (even if for the second or third attempt)
  – if you are allowed to take an exam, your attendance at any term is NOT mandatory – you can choose a term
How to prepare for the exam

• Part 1 – Design and implementation of combinational circuits
  – preparation (truth tables, Karnaugh maps)
  – obtaining definition for the circuit in the requested form
  – implementation with given elements (gates, commutators, PLDs)
How to prepare for the exam

• Part 1 – Design and implementation of combinational circuits

Design an iterative circuit with $N$ inputs $a_i$ and $N$ inputs $b_i$, executing comparison of binary numbers $A_i$ and $B_i$ (where $A_i \rightarrow a_ia_{i-1}...a_0$, and $B_i \rightarrow b_ib_{i-1}...b_0$). The output $Y_i$ should equal 1 when $A_i$ and $B_i$ differ on exactly two positions. Provide all necessary assumption for correct operation of the circuit, and equations and a logical diagram for a universal cell using PLA element.
How to prepare for the exam

• Task 2 – Design of asynchronous sequential circuits with Switching Sequence Table method + implementation
  – construction of a SST for a circuit
  – verification of solvability
  – for unsolvable tables
    • introduction of auxiliary state variables through the process of placing borders between sections
    • encoding of sections
    • obtaining solvable table
  – obtaining definition of the circuit in the requested form
  – implementation
    • gates and feedback loops
    • flip-flops
How to prepare for the exam

• Task 2 – Design of asynchronous sequential circuits with Switching Sequence Table method + implementation

Using Switching Sequence Table method design a circuit with a working cycle given by a timing chart. Present a solution in the form of minimal equations describing a circuit. Draw logical diagram with 2-input NOR gates.
How to prepare for the exam

• Task 3 – Design of asynchronous sequential circuits with Huffman’s method + implementation
  – construction of a primitive flow table for a circuit
  – reduction of equivalent states
  – merging of compatible states
  – state assignment, with taking into account the problem of races
  – obtaining definition of the circuit in the requested form
  – implementation
    • gates and feedback loops
    • flip-flops
How to prepare for the exam

• Task 3 – Design of asynchronous sequential circuits with Huffman’s method + implementation

Using Huffman’s method design a circuit with the working cycle defined by the timing chart given. Obtain minimal number of rows in the flow table for Moore machine. Present a solution in the form of minimal equations for implementation with $sr$ flip-flops, without drawing logical diagram.
How to prepare for the exam

• Task 4 – Design of synchronous sequential circuits
  – construction of state diagrams
  – design of parallel counters and frequency dividers
  – Huffman’s method for synchronous circuits
  – present state/next state tables
  – illegal state recovery mechanisms
  – implementation for synchronous flip-flops
How to prepare for the exam

• Task 4 – Design of synchronous sequential circuits

Design a synchronous circuit detecting that for any 3 consecutive bits, occurring anywhere in the sequence on the serial input \( x \), the middle one is different in logic value from its two neighbours. Detection should be indicated by setting the output \( W=1 \) for the time not longer than one clock period. Present a solution in the form of minimal logic expressions for JK and T flip-flops (both types used in one design), without drawing a logical diagram.
How to prepare for the exam

• Task 5 – Design of microprogrammable circuits
  – choice of the structure of microprogrammable circuit
  – memory content corresponding to logical diagram
  – logical diagram corresponding to the list of microinstructions
  – considerations on possible optimisation (reduction of memory element)
How to prepare for the exam

Task 5 – Design of microprogrammable circuits

Design a microprogrammable circuit for a system responsible for steering of electrical engine that drives a coal-truck:

- input position detectors A and B (1 when truck detected above, 0 otherwise),
- input “start” push-release button (1 when pressed, 0 when released),
- outputs L (left rotation), R (right rotation).

When the system is switched on, the truck is always located in the left-most position (A=1) and stopped (L=0, R=0).

After pressing Start button (S=1), truck moves to the right (L=0, R=1). Assume that the button may be released before the truck leaves sensor A detection range (A=1), or may be kept pressed (A=0).

When the truck reaches the end of transportation rail (B=1) engine changes immediately rotation direction from right to left (L=1, R=0). The truck continues its travel back (A=0, B=0) as it finally reaches the starting position (A=1), where work sequence ends (L=0, R=0). Write the content of the memory and draw matching it a complete logical structure for the circuit. Note that the grade depends on size of the memory element used (including the selection of the applied structure). You should not use any auxiliary logic elements apart from those inherent for the chosen structure.
Summary of material

• Basis of Boolean algebra
  – definition and postulates,
  – operators, binary variables,
  – theorems,

• Boolean functions
  – logic formulas,
  – various forms of expression,

• Minimisation of logic functions by
  – algebraic transformations
  – Karnaugh maps
  – algorithms for systematic generation of prime implicants
  – minimal cover of implicants

• Forms of presentation of working cycle for switching circuits

• Classification of logic circuits
  – combinational,
  – sequential
Summary of material

- Fundamental switching elements of combinational type
  - gates and their logic symbols,
  - functionally complete systems,
  - implementations for various sets of logic elements
  - implementation on gates with limited inputs
- Commutators
  - Inside structures of MUX and DMUX and implemented functions
  - Simple implementations for MUX and DMUX
  - Advanced structures (MUX+gates, DMUX-MUX, Tree of MUX)
  - Approaches to obtain sub-functions (Karnaugh maps, Algebraic transformations, Matrix method)
- Flip-flops
  - elements describing work
  - methods for obtaining input excitation functions
  - triggering for synchronous flip-flops
- Delay units
- Programmable Logic Devices
- Implementation technologies
Summary of material

• Problems resulting from circuit dynamics
  – synthesis of hazard-free functions
  – analysis with respect to hazards
  – races
  – essential hazard

• Synthesis of general type combinational circuits

• Sequential circuits
  – working mode and structure for circuits with memory,
  – categories of circuits

• Binary codes

• Circuits operating on codes
  – Encoders
  – Decoders
  – Translators

• Synthesis of iterative circuits

• Circuits with time dependencies

• Arithmetic circuits
Summary of material

• Asynchronous sequential circuits
  – SST method,
  – Huffman’s method,
  – implementation with gates and flip-flops

• Synchronous sequential circuits
  – state diagrams,
  – Huffman’s method,
  – present state / next state method

• Specialised sequential circuits
  – frequency dividers,
  – parallel counters,
  – illegal state recovery,
  – microprogrammable circuits,
  – serial counters,
  – registers