

# General

## Student Group

First Name	Surname	Matrikel Nr.

## Table of Contents

<b>General</b> .....	2
<b>Conversion of the number systems</b> .....	2
<b>Simplifying Boolean expressions</b> .....	2
<b>Setting up the KNF and DNF</b> .....	2
<b>KV diagrams</b> .....	2
<i>Creating a KV diagram</i> .....	3
<i>Reading out the prime implicants of a KV diagram.</i> .....	3
<i>State machines</i> .....	3

# General

- **No electronic aids are allowed** in the exam (no calculator, no mobile phone, no laptop). Prepare for the calculation by hand.
- **Results without derivation will not be graded.**
- **Please read the questions carefully.** Work on the questions in the requested way (e.g. use gate circuits, truth table, or explicit indication of the boolean rule if requested).
- **Take the chance to prepare documents.** A collection of formulas (e.g. with laws of Boolean algebra) shortens the search in the documents. Furthermore, it is advisable to equip the script with sticky notes, so that the search is easier.  
**Please note, however, that neither formularies nor scripts are allowed in the exam.** (so it's only for the preparation)

## Conversion of the number systems

- Please always indicate the number system (e.g. hexadecimal numbers: 0x10, 10H, 10h).
- The conversion from hex to binary or back (1101b  $\leftrightarrow$  Dh) is easiest for one nibble at a time using a table.
- Converting from hex to binary or back is usually easier than converting from decimal to either system. Converting from binary to decimal is a little more involved. If the conversion from decimal to hex and binary is required, then it is almost always easier to first convert the decimal number to a hex number and then to a binary number (55d = 37h = 0011 0111b).
- The decimal places are also combined into nibbles starting from the decimal point.
- Note that the decimal places must of course also be converted (0.5h  $\neq$  0.5d !).

## Simplifying Boolean expressions

- In truth tables of gate circuits, for example, columns for intermediate results are useful. Otherwise, if the result is wrong, I cannot reconstruct the intermediate steps and thus cannot award any points.
- Shorten logical expressions so that as few logical operations as possible are needed (often a shortening is still possible via De Morgan's law, distributive law or the definition of the XOR operation).
- Use this method to write binary numbers quickly among each other.

## Setting up the KNF and DNF



- You can avoid careless errors in setting up the DNF and KNF if you use the same sorting of the inputs as in the function table.
- If a minimization of the normal form is required, then you should apply the Boolean laws (see Simplifying Boolean Expressions).

## KV diagrams

## Creating a KV diagram

- If you are to minimize the Boolean function for many outputs and are faced with a KV diagram for which you have no template, you can proceed as follows: First, create a KV diagram in which you enter only the decimal numbers of the positions. Two coordinates are quickly entered in. The 0 is always in the cell where all inputs are negated (usually top left). The maximum value is always in the cell where all inputs are non-negated (usually in the middle).

## Reading out the prime implicants of a KV diagram.

1. First, determine what is being looked for:
  1. are you looking for minterms = conjunctive form? → “groups of 1s” or
  2. are you looking for maxterms = disjunctive form? → “groups of 0s”.
2. Search for the respective implicants (0s or 1s).
3. Starting from the implicants, try to form a prime implicant (= area / group with the same binary value) as large as possible. “Don't care” states (marked with “d.c.”, “-” or “x”) may be considered as a suitable implicant. Note that only certain types of groups are allowed as prime implicants:
  1. The size of prime implicants can only be a power to the base 2 (1, 2, 4, 8, ...).
  2. Only rectilinear implicants can be connected, or implicants that are not only adjacent across corners (e.g. , ).
4. When you have found all the largest possible prime implicants, then you need to check what dependencies they have on each other. Often there is a constellation of prime implicants that has the fewest terms.
5. On the basis of the plotted prime implicants the logical function can be derived in equation form. In general, this can still be simplified (e.g. via definition of the XOR or De Morgan's law).

## State machines

The following procedure is recommended for every rear state machine synthesis:

1. Careful reading of the specifications
2. Determine which specifications already exist, e.g.
  1. Type of automaton (asynchronous = Mealy automaton, synchronous = Medvedev automaton or Moore automaton)
  2. Type of flip-flops to be used (D-FF, JK-FF, SR-FF, ...)
3. Determining the input, state and output variables
4. Determining the number of flip-flops required
  1. For Mealy/Moore automata via the number of different states
  2. For Medvedev automata via the maximum value to be output
5. Setting up a state transition diagram
  1. For Mealy and Moore automata, the assignment of state to output values can be arbitrary.
  2. For Medvedev automata, the assignment of state values to output values is directly specified.
6. Creating the state transition table from the state transition diagram.
  1. The column arrangement below is suitable.  
This means that the inputs and outputs for the different switching networks (transition and output network) are directly next to each other.

Zustands- übergangs- tabelle	Aktueller Zustand			Eingangswert des Automaten	Nächster Zustand			Ausgangswert des Automaten	
	Z2	Z1	Z0	X0	Z2'	Z1'	Z0'	Y1	Y0
	0	0	0	0	0	0	1	0	0
	0	0	0	1	0	1	0	1	0
	0	0	1	0	1	1	0	1	0
...	...	...	...	...	...	...	...	...	...

Übergangnetzwerk (ÜNW)			
Eingangswert des ÜNW	Aktueller Zustand	Eingangswert des Automaten	
Ausgangswert des ÜNW *)			Nächster Zustand

Ausgangnetzwerk (ANW)			
Eingangswert des ANW	Aktueller Zustand	nur bei Mealy: Eingangswert	
Ausgangswert des ANW			Ausgangswert des Automaten

- \*) The output value of the transition network is only the next state if the switching mechanism is implemented on DD flip-flops. If the switching mechanism is not implemented on D-FF, a conversion to FF input values must be taken into account.

7. The CT diagrams for the respective switching networks can be created from the columns for the transition network and output network.

From: <https://mexle.te.hs-heilbronn.de/> - MEXLE Wiki

Permanent link: [https://mexle.te.hs-heilbronn.de/introduction\\_to\\_digital\\_systems/tips\\_for\\_the\\_exam](https://mexle.te.hs-heilbronn.de/introduction_to_digital_systems/tips_for_the_exam)

Last update: 2024/01/30 18:36

