georg.von-der-brueggen [☺] tu-dortmund.de
ulrich.gabor [☺] tu-dortmund.de
pascal.libuschewski [☺] tu-dortmund.de

Exercises for
Rechnerstrukturen
Winter Semester 2016

Exercise Sheet 3 (Block A - 3)

(16 points)

Submission until Wednesday, 16th November 2016, 16:00o'clock
Discussion begins on Tuesday, 22nd November 2016

Please see notes at the end of the document for the submission.

3.1 Calculation with Two's Complement (4 points)

Calculate without transferring the numbers into the decimal system:

a. $0110\,0100 + 0011\,0010$

b. $1001\,0011 + 0101\,0110$

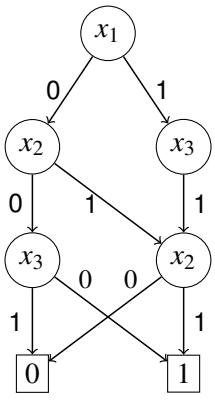
c. $0001\,0111 - 1001\,1110$

d. $0011\,0010 - 0110\,0010$

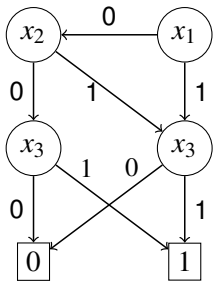
The numbers are given as two's complement (8-bit patterns). Also use this representation for the results. Mark if a result is invalid.

3.2 OBBDs (4 points)

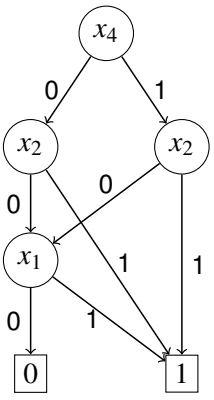
The following four graphs G_1, G_2, G_3, G_4 that represent functions $f_1, f_2, f_3, f_4 : B^4 \to B$ are given. Decide if they are πOBDDs and explain why. If the graph is a πOBDD reduce it step by step to a minimal πOBDD and explain the rules you use in every step.



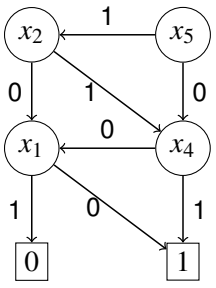
(a) G_1



(b) G_2



(c) G_3



(d) G_4

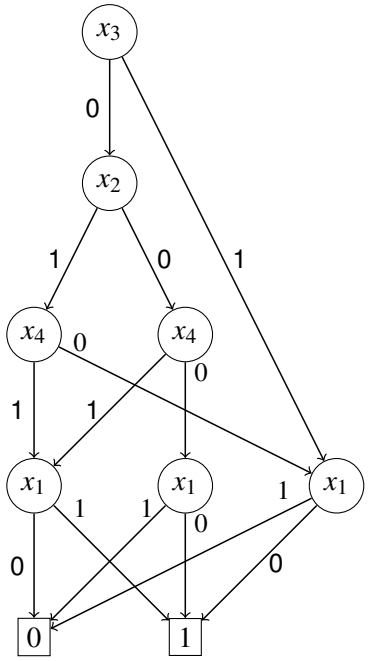
3.3 OBBDs (4 points)

Reduce the given πOBDD step by step using the reduction rules. Use one one reduction rule per step.

a. Use the numerated list to write down the rules you use and mark the nodes where the rules are used by writing the line number to the related node. You do not need to draw a new πOBDD after each time you used a rule.

b. Draw a graph for the reduced πOBDD.

c. It is not necessary to find as many rules that can be used as there are numerated lines.



1. _____
2. _____
3. _____
4. _____
5. _____

3.4 Combinatorial Circuit for an Adder (4 points)

Two binary coded absolute values $x = (x_7, \ldots, x_0)_2$ and $y = (y_7, \ldots, y_0)_2$ shall be added. The result shall be displayed in $s = (s_7, \ldots, s_0)_2$. As you know you need to calculate the carry bit $c = (c_8, \ldots, c_1)_2$ where c_i is the carry that happens when x_{i-1}, y_{i-1} and c_{i-1} are added up.

a. Sketch the combinatorial circuit for an 8-bit Ripple-Carry-Adder. Use as few half adders and full adders as possible.

b. Sketch the combinatorial circuit for a half adder. You only need “AND” gates and “XOR” gates.

c. Sketch the combinatorial circuit for a full adder. You only need “AND” gates, “OR” gates, and “XOR” gates.

d. That is the total runtime until a result is present at c_8? The runtime of one gate is assumed to be t. Assume that x_i and y_i are present at the in-ports of all half adders and full adders at the same time. The runtime of the half adders and full adders depends on the inner construction of the combinatorial circuit and has to be accounted for accordingly.

Notes:
Submission until Wednesday, 16th November 2016, 16:00 pm in the mailbox number 40 at Otto-Hahn-Straße 12.

You can find the mailboxes in the first floor of the Otto-Hahn-Straße 12 near the transition to the ground floor of the Otto-Hahn-Straße 14. The mailboxes are labeled with “Rechnerstrukturen”, the exercise group number and time/place of the exercise. The English exercise group is number 31 and the mailbox is number 40.

Please write your **name**, your **student registration number** and your **exercise group number** at the top right corner of your submission. You can make submissions in teams with up to two more students. To make a team submission put names, student registrations numbers and group numbers of all members of the team on the submission. Only one submission per team has to be made.

Tack you submission. Please do not fold your submission and do not put it into an envelope. Use the correct mailbox, you will need your exercise group number for that.

In total there are 12 exercises in 3 blocks (A, B, C). In each block you have to achieve at least 30 points of 64 possible ones to get access to the exam.