Section A Machine Architecture and Digital Electronics (Dr. M. Bordewich)

Question 1

You are designing an automatic ordering system for a cafe which has 11 dishes on its menu. When a client presses a button for one of these dishes, the signal is represented as a 4-bit binary number as follows: 0000: roast beef, 0001: asparagus tart (V), 0010: sausages, 0011: cheese sandwich (V), 0100: chicken pie, 0101: steak pie, 0110: grilled mushroom (V), 0111: quiche (V), 1000: salad (V), 1001: meatballs, 1010: tomato soup (V). You wish to create a (combinational) circuit which takes four inputs A,B,C and D which together encode this 4-bit number n, and has one output F indicating whether the dish is suitable for a vegetarian (indicated with a (V)). The output should be 1 if the signal represents a vegetarian dish. In ALL other cases the output should be 0.

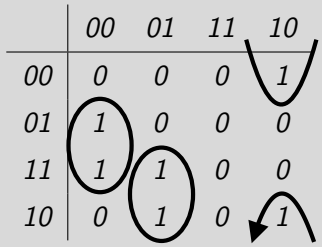
(a) Express the Boolean function F given above in unsimplified "sum of products" form. **[3 Marks]**

Solution: $F = \overline{ABC}D + \overline{AB}CD + \overline{A}BC\overline{D} + \overline{A}BCD + A\overline{BC}D + A\overline{B}C\overline{D}$

Notice here that the editor used has produced a formula that looks incorrect! The first term of the formula writes "Not(A and B and C) and D", while the correct term is "NotA and NotB and NotC and D", which is different to what is written. This is an issue to do with the editor used to provide the pdf file. Be careful when typing your answers so that you avoid such mistakes. Similarly, the second term should write "NotA and NotB and ..." rather than "Not(A and B) and ...". Same with the 5th term of the formula, which should write "A and NotB and NotC and NotD".

(b) Draw a Karnaugh map for this truth table and use it to produce a simpler formula for the same Boolean function. You should describe the construction of the Karnaugh map and process used to produce a simplified formula, and show your working on this example. **[10 Marks]**

Solution: *K-map:*



K-map must have row labels representing two of the variables values and col labels that represent the other two inputs variables values, and such that moving by one row or col results in only one input changing value. Thus

continued

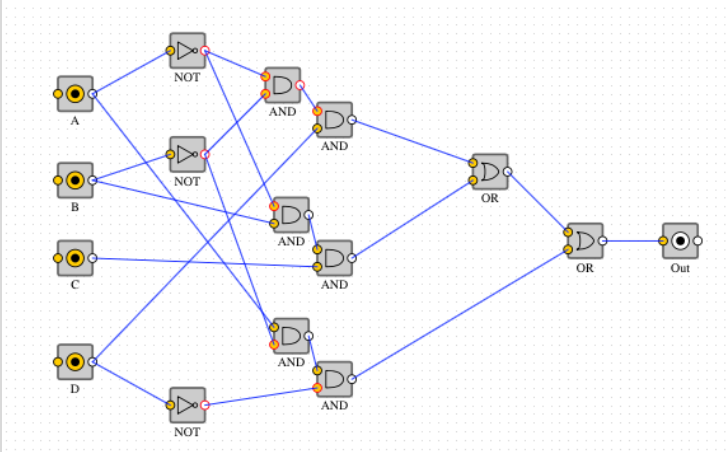
each cell represents a minterm with the variables labelled 1 as positive literals and the variables labelled 0 negated. The cell is filled with either a 1 or 0 depending on whether that minterm/entry in the truth table is true or false. You then circle all the 1s in the table using rectangles with side lengths powers of 2 such that you use as few rectangles as possible each of size as large as possible. Finally you read off the implicants corresponding to each rectangle: the variables that are constant within the circle.

Here the circled cells correspond to: $F = \overline{A}\overline{B}D + \overline{A}BC + A\overline{B}\overline{D}$

Same issue here, as in the solution of (a):
Term 1 should be "NotA and NotB and D". Term 3 should be "A and NotB and NotD".
To avoid such issues, make sure that you leave a clear space between two consecutive negated variables when including them in a product.

(c) Draw a circuit diagram that gives an appropriate circuit for this situation using 2-input AND, OR and NOT gates. **[2 Marks]**

Solution:



It is now decided that the output value does not matter for inputs representing numbers that do not correspond to dishes (since there are 16 possible input states and only 11 dishes). For input values representing dishes the value must be 1 or 0 as before. Explain how this could affect your calculations so far and produce a simpler formula for a Boolean function that satisfies this new requirement. **[5 Marks]**

Solution: *We can populate some cells of the K-map with Xs representing 'don't cares'. We then make as large rectangles as possible covering all the 1s and any Xs that are useful, though we don't have to cover the Xs. We use fewer and larger rectangles than before, so have fewer, smaller implicants. Correctly updated table:*

continued

(d)

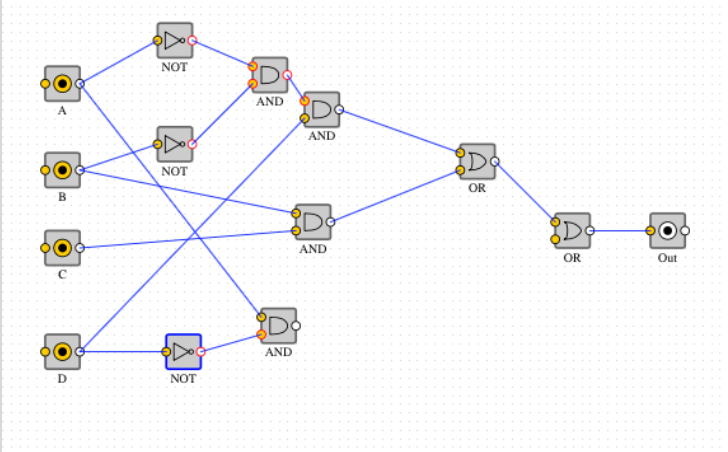
	00	01	11	10
00	0	0	X	1
01	1	0	X	0
11	1	1	X	X
10	0	1	X	1

Correctly updated formula: $F = \overline{A}\overline{B}D + BC + A\overline{D}$

Same issue here, as in the solution of (b) and (c): Term 1 should be "NotA and NotB and D".

(e) Draw a circuit diagram that gives an appropriate circuit for the revised situation using 2-input AND, OR and NOT gates. **[2 Marks]**

Solution:



If the contamination and propagation delay of AND, OR and NOT gates is given by $t_{cd}(\text{AND}) = 30, t_{cd}(\text{OR}) = 25, t_{cd}(\text{NOT}) = 20$ and $t_{pd}(\text{AND}) = 60, t_{pd}(\text{OR}) = 50$ and $t_{pd}(\text{NOT}) = 30$, what is the contamination and propagation delay of your final circuit. Show your working. **[3 Marks]**

Solution: *Critical path has a NOT gate, 2 AND gates and an OR gate: propagation delay 200ps. Short path has one AND and one OR gate: time 55ps.*

END OF PAPER