

**School of Computer Science**

**Faculty of Science**

**COMP-2650: Computer Architecture I: Digital Design**

**Winter 2020**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Assignment# | Date | Title | Due Date | Grade Release Date |
| Lec 08 | Week 08 | **Karnaugh Map** | March 02, 2021  Tuesday Midnight [AoE](https://www.timeanddate.com/time/zones/aoe)  Wednesday 7 AM EDT | March 08, 2021 |

The objectives of the lecture (weekly) assignments are to practice on topics covered in the lectures as well as improving the student’s critical thinking and problem-solving skills in ad hoc topics that are closely related but not covered in the lectures. Lecture assignments also help students with research skills, including the ability to access, retrieve, and evaluate information (information literacy).

**Lecture Assignments Deliverables**

You should answer two of the below questions based on your preference using an editor like MS Word, Notepad, and the likes or pen in papers. You have to write and scan the papers clearly and merge them into a single file in the latter case. In the end, you have to submit all your answers in one single pdf file Lec08\_UWinID.pdf containing the following items:

1. Your name, UWinID, and student number
2. The question Id for each answer. Preferably, the questions should be answered in order of increasing Ids. *Please note that if your answers cannot be read, you will lose marks.*
3. Including the questions in your submission pdf file is optional.

*Please follow the naming convention as you lose marks otherwise.* Instead of UWinID, use your own UWindsor account name, e.g., mine is [hfani@uwindsor.ca](mailto:hfani@uwindsor.ca), so, my submission would be: Lec08\_hfani.pdf

**Lecture Assignments**

**(Select Only 2 Questions based on your preference)**

**Use K-Map method for all simplification tasks!**

1. Simplify the following Boolean functions, using three-variable maps:
2. F(x, y, z) = ∑(0, 2, 4, 5)
3. F(x, y, z) = ∑(0, 2, 4, 5, 6)
4. F(x, y, z) = ∑(0, 1, 2, 3, 5)
5. F(x, y, z) = ∑(1, 2, 3, 7)
6. Simplify the following Boolean expressions, using three-variable maps:
7. xy + x’y’z’ + x’yz’
8. x’y’ + yz + x’yz’
9. x’y + yz’ + y’z’
10. x’yz + xy’z’ + xy’z
11. Simplify the following Boolean functions, using Karnaugh maps:
12. F (A, B, C, D) = ∑ (4, 6, 7, 15)
13. F (A, B, C, D) = ∑ (3, 7, 11, 13, 14, 15)
14. F (w, x, y, z) = ∑ (2, 3, 12, 13, 14, 15)
15. F (w, x, y, z) = ∑ (11, 12, 13, 14, 15)
16. F (w, x, y, z) = ∑ (8, 10, 12, 13, 14)
17. Simplify the following Boolean expressions, using four-variable maps:
18. w'z + xz + x’y + wx’z
19. AD’ + B’C’D + BCD’ + BC’D
20. AB’C + B’C’D’ + BCD + ACD’ + A’B’C + A’BC’D
21. wxy + xz + wx’z + w’x
22. Find the minterms of the following Boolean expressions by first plotting each function in a map:
23. xy + yz + xy’z
24. C’D + ABC’ + ABD’ + A’B’D
25. wyz + w’x’ + wxz’
26. A’B + A’CD + B’CD + BC’D’
27. Convert the following Boolean function from a sum-of-products form to a simplified product-of-sums form: F(x, y, z) = ∑(0, 1, 2, 5, 8, 10, 13)
28. Simplify the following Boolean functions:
29. F(A, B, C, D) = ∏(1, 3, 5, 7, 13, 15)
30. F(A, B, C, D) = ∏(1, 3, 6, 9, 11, 12, 14)
31. Simplify the following Boolean function F, together with the don’t-care conditions d, and then express the simplified function in sum-of-minterms form:
32. F(x, y, z) = ∑ (0, 1, 4, 5, 62) + d(2, 3, 7)
33. F(A, B, C, D) = ∑ (0, 6, 8, 13, 14), d(A, B, C, D) = ∑ (2, 4, 10)
34. F(A, B, C, D) = ∏(5, 6, 7, 12, 14, 15), D(A, B, C, D) = ∏(3, 9, 11, 15)
35. F(A, B, C, D) = ∏(4, 12, 7, 2, 10) + D(10, 6, 8)
36. Simplify the following functions, and implement them with two-level NOR gate circuits:
37. F = wx’ + y’z’ + w’yz’
38. F(w, x, y, z) = ∑(0, 3, 12, 15)
39. F(x, y, z) = [(x + y)(x = z)]’
40. Draw the simplified NOR-only circuit for the following expression: CD(B + C) A + (BC’ + DE’)
41. Draw the simplified NAND-only circuit for the following expression: w(x + y + z) + xyz
42. Obtain the simplified Boolean expressions for output F and Gin terms of the inputvariables in this circuit:

