CMSC 313 Lecture 20

- Homework 4 Questions
- Karnaugh Map examples

Last Time

- Combinational logic components
- Introduction to Karnaugh Maps

Karnaugh Maps

- **Implicant:** rectangle with 1, 2, 4, 8, 16 ... 1's
- **Prime Implicant:** an implicant that cannot be extended into a larger implicant
- **Essential Prime Implicant:** the only prime implicant that covers some 1
- **K-map Algorithm (not from M&H):**

    1. Find ALL the prime implicants. Be sure to check every 1 and to use don't cares.

    2. Include all essential prime implicants.

    3. Try all possibilities to find the minimum cover for the remaining 1's.

Notes on K-maps

- Also works for POS
- Takes 2^n time for formulas with n variables
- Only optimizes two-level logic
  - Reduces number of terms, then number of literals in each term
- Assumes inverters are free
- Does not consider minimizations across functions
- Circuit minimization is generally a hard problem
- Quine-McCluskey can be used with more variables
- CAD tools are available if you are serious

Circuit Minimization is Hard

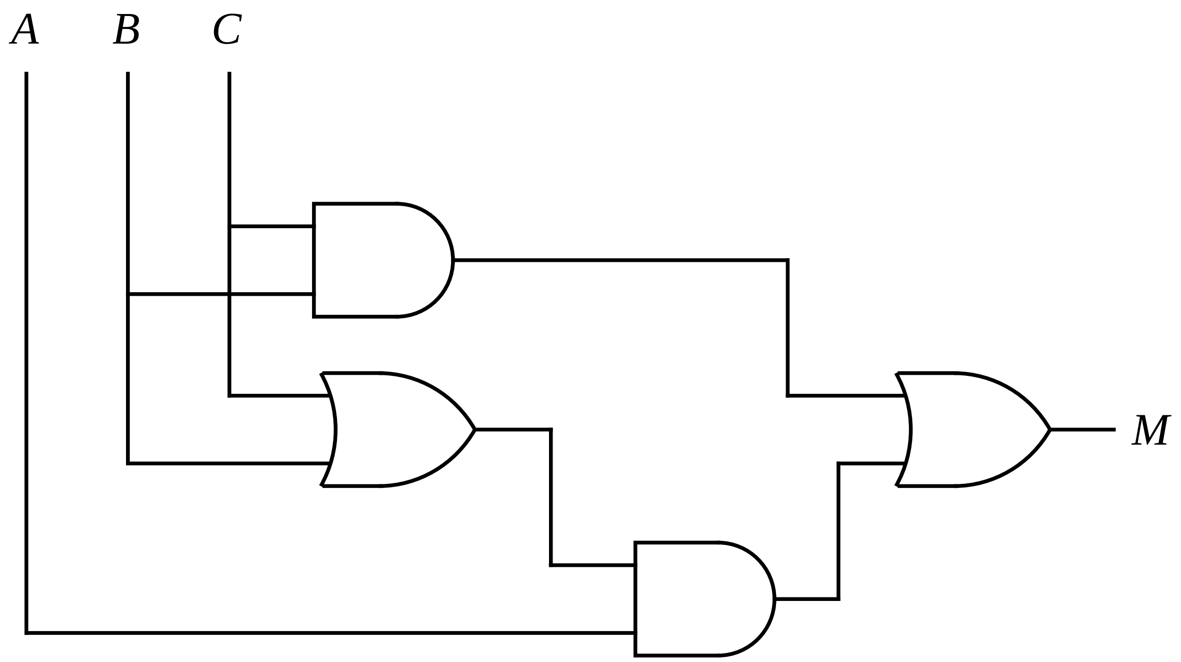
- Unix systems store passwords in encrypted form.
  - User types in x, system computes f(x) and looks for f(x) in a file.
- Suppose we us 64-bit passwords and I want to find the password x, such that f(x) = y. Let

  $g_i(x)$ = 0 if f(x) = y and the ith bit of x is 0

  1 otherwise.
- If the ith bit of x is 1, then $g_i(x)$ outputs 1 for every x and has a very, very simple circuit.
- If you can simplify every circuit quickly, then you can crack passwords quickly.

3-Level Majority Circuit

- K-Map Reduction results in a reduced two-level circuit (that is, AND followed by OR. Inverters are not included in the two-level count). Algebraic reduction can result in multi-level circuits with even fewer logic gates and fewer inputs to the logic gates.



Due: Thursday, November 13, 2003

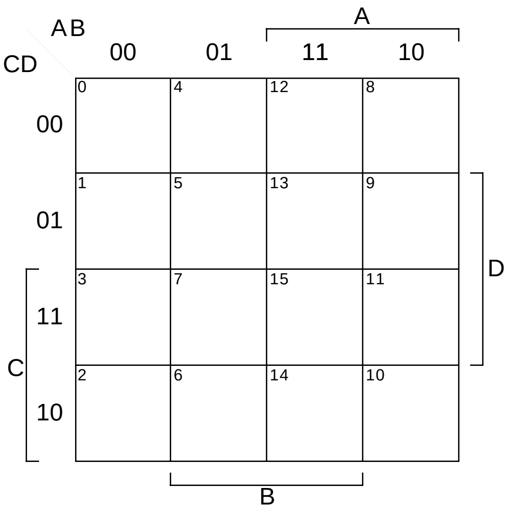
1. (10 points) Question A.12, page 494, Murdocca & Heuring
2. (10 points) Question A.16, page 495, Murdocca & Heuring
3. (10 points) Question A.17, page 495, Murdocca & Heuring
4. (50 points) In the following, the notation $\sum m(x_1, \ldots, x_j)$ indicates a Boolean function that is the sum of the minterms $x_1, \ldots, x_j$, where x_i is the ith minterm in canonical ordering — i.e., the ith row of the truth table where the input values are ordered as binary numbers. Similarly,

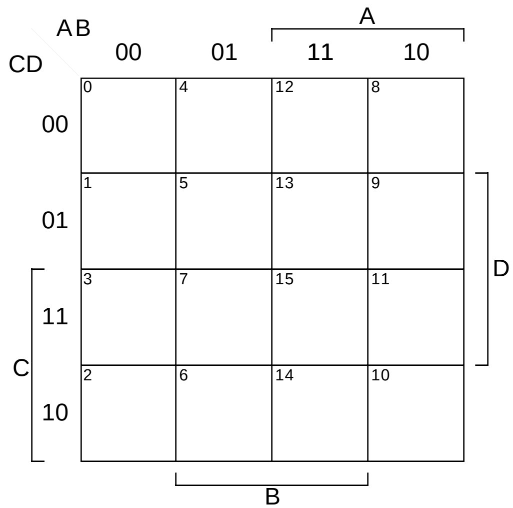
$$\sum m(x_1, \ldots, x_j) + d(y_1, \ldots, y_k)$$

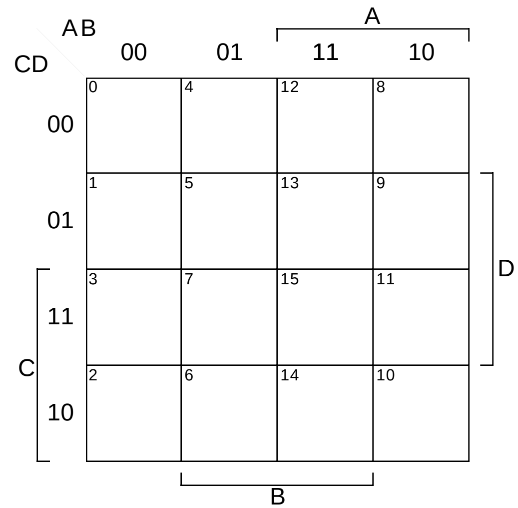
indicates a Boolean function that is the sum of the minterms $x_1, \ldots, x_j$ and whose values for rows $y_1, \ldots, y_k$ of the truth table are *don't cares.*

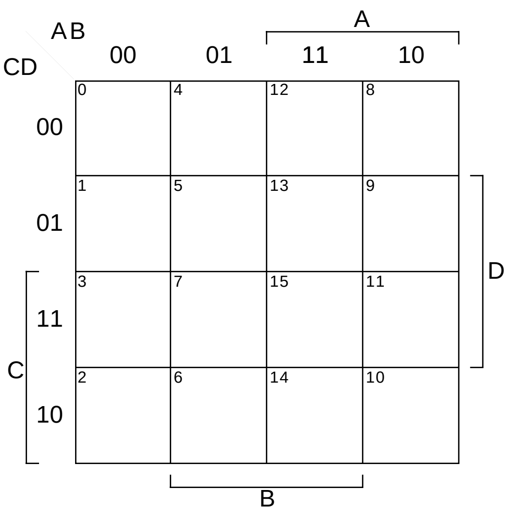
Minimize the following functions using Karnaugh maps. Then, write down a Boolean formula in sum-of-products or product-of-sums form for each function. Show your work (including the Karnaugh maps).

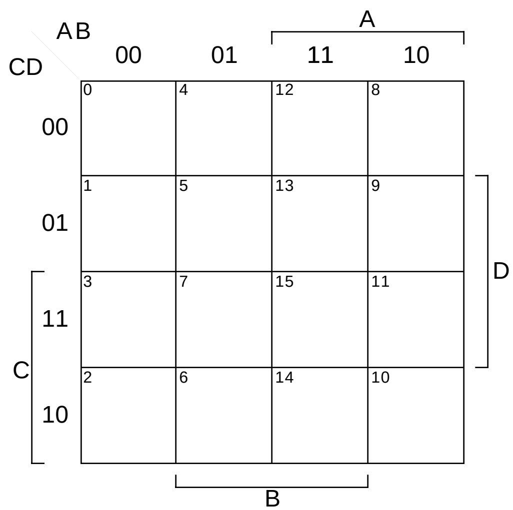
(a) $f(A, B, C) = \sum m(2, 3, 4, 5)$
(b) $f(A, B, C, D) = \sum m(0, 1, 4, 6, 9, 13, 14, 15)$
(c) $f(A, B, C, D) = \sum m(0, 1, 2, 8, 9, 10, 11, 12, 13, 14, 15)$
(d) $f(A, B, C, D) = \sum m(2, 9, 10, 12, 13) + d(1, 5, 14)$
(e) $f(A, B, C, D) = \sum m(1, 3, 6, 7) + d(4, 9, 11)$

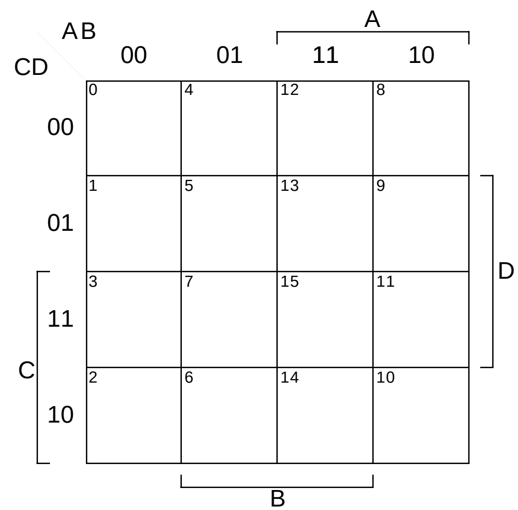

AB
CD
A
00 01 11 10
00
01
11
10
0 4 12 8
1 5 13 9
3 7 15 11
2 6 14 10
C
D
B


AB
CD
A
00 01 11 10
00
01
11
10
0 4 12 8
1 5 13 9
3 7 15 11
2 6 14 10
C
D
B


AB
CD
A
00 01 11 10
00
01
11
10
0 4 12 8
1 5 13 9
3 7 15 11
2 6 14 10
C
D
B


AB
CD
A
00 01 11 10
00
01
11
10
0 4 12 8
1 5 13 9
3 7 15 11
2 6 14 10
C
D
B


AB
CD
A
00 01 11 10
00
01
11
10
0 4 12 8
1 5 13 9
3 7 15 11
2 6 14 10
C
D
B


AB
CD
A
00 01 11 10
00
01
11
10
0 4 12 8
1 5 13 9
3 7 15 11
2 6 14 10
C
D
B

Next Time

- Quine-McCluskey
- Flip Flops