Computer Science CSCI 355

Digital Logic and Computer Organization

Dr. Peter Walsh
Department of Computer Science
Vancouver Island University
peter.walsh@viu.ca

Functional Equivalence

$$(x + y)(x + \overline{y}) = x$$

It would be useful to have a standard form such that functional equivalence can be established.

○ Canonical Forms

- truth tables
- sum of products (SOP)
- product of sums (POS)
- binary decision diagrams (BDD)

Sum of Products (SOP)

a b c	Minterm		Majority (f)
0 0 0	m0	a'b'c'	0
0 0 1	m1	a'b'c	0
0 1 0	m2	a'bc'	0
0 1 1	m3	a'bc	1
1 0 0	m4	ab'c'	0
1 0 1	m5	ab'c	1
1 1 0	m6	abc'	1
1 1 1	m7	abc	1

SOP f = a'bc + ab'c + abc' + abc

$\sum_m (3, 5, 6, 7)$

Sum of Products (SOP) cont.

a b c	Minterm		Majority (f)
0 0 0	m0	a'b'c'	0
0 0 1	m1	a'b'c	0
0 1 0	m2	a'bc'	0
0 1 1	m3	a'bc	1
1 0 0	m4	ab'c'	0
1 0 1	m5	ab'c	1
1 1 0	m6	abc'	1
1 1 1	m7	abc	1

One of the minterms evaluates to 1 for each combination of inputs for which the output is 1.

Product of Sums (POS)

a b c	Maxterm		Majority (f)
0 0 0	M0	a+b+c	0
0 0 1	M1	a+b+c'	0
0 1 0	M2	a+b'+c	0
0 1 1	M3	a+b'+c'	1
1 0 0	M4	a'+b+c	0
1 0 1	M5	a'+b+c'	1
1 1 0	M6	a'+b'+c	1
1 1 1	M7	a'+b'+c'	1

POS f = (a+b+c)(a+b+c')(a+b'+c)(a'+b+c)

$\prod_M (0, 1, 2, 4)$

Product of Sums (POS) cont.

a b c	Maxterm		Majority (f)
0 0 0	M0	a+b+c	0
0 0 1	M1	a+b+c'	0
0 1 0	M2	a+b'+c	0
0 1 1	M3	a+b'+c'	1
1 0 0	M4	a'+b+c	0
1 0 1	M5	a'+b+c'	1
1 1 0	M6	a'+b'+c	1
1 1 1	M7	a'+b'+c'	1

One of the maxterms evaluates to 0 for each combination of inputs for which the output is 0.

Karnaugh (K) Map Minimization

Geometric representation of a logic expression that facilitates logic minimization.

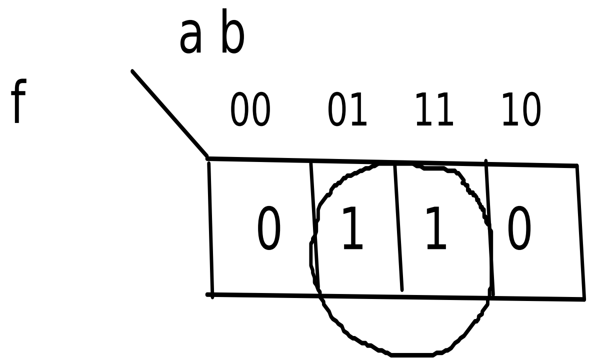
a	b	m
0	0	0
0	1	1
1	0	2
1	1	3

a \ b	00	01	11	10
	m0	m1	m3	m2

Note the Gray Code ordering

Minimization Example

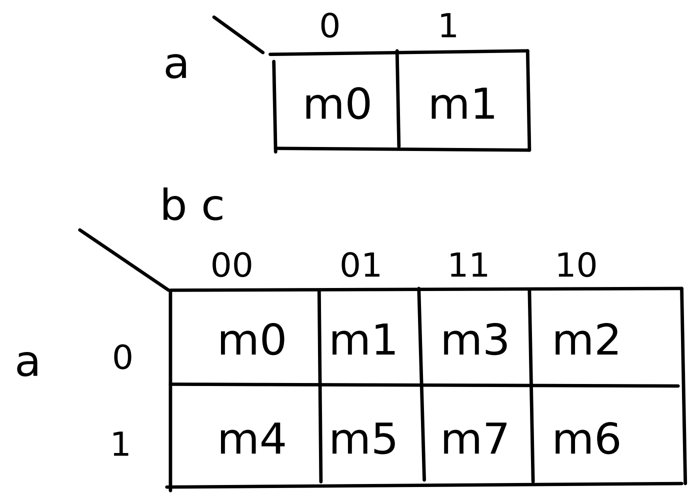
- Minimize $\Sigma m(1,3)$



f = a'b + ab = b

1, 3 Variable K Maps

- Minterm Layout



4 Variable K Map

- Minterm Layout

ab \ cd	00	01	11	10
00	m0	m1	m3	m2
01	m4	m5	m7	m6
11	m12	m13	m15	m14
10	m8	m9	m11	m10

Gray Code Construction Example

0	0	0
0	0	1
0	1	1
0	1	0
1	1	0
1	1	1
1	0	1
1	0	0

Reflect

5 Variable K Map

- Minterm Layout

a=0

bc \ de	00	01	11	10
	m0	m1	m2	m3
	m4	m5	m7	m6
	m12	m13	m15	m14
	m8	m9	m11	m10

a=1

de	00	01	11	10
	m16	m17	m19	m18
	m20	m21	m23	m22
	m28	m29	m31	m30
	m24	m25	m27	m26

K Map Terminology (SOP)

- Implicant
  - any (power of 2) grouping of adjacent 1s
- Cover
  - a set of implicants that include all the 1s
- Prime Implicant (PI)
  - an implicant that can not be "grown" any bigger
- Essential Prime Implicant
  - a PI that must be included in a cover
- Secondary Prime Implicant (Non Essential PI)
  - an implicant that is not an essential PI

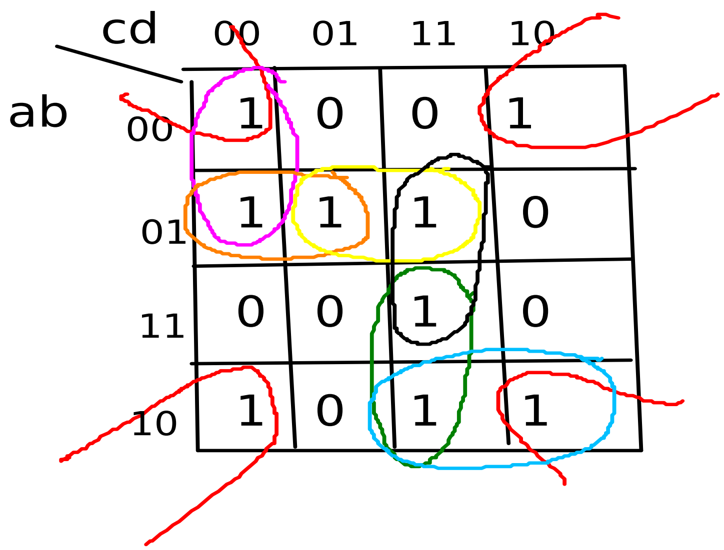
K Map Minimization Algorithm (SOP)

1. Identify all prime implicants.

2. Identify the set of essential prime implicants E.

3. Select the minimum set of non-essential prime implicants N such that $\{E\} \cup \{N\}$ forms a cover.

Example



b'd' is the only essential PI
(red grouping)