Lecture 7: Minimization with Karnaugh Maps

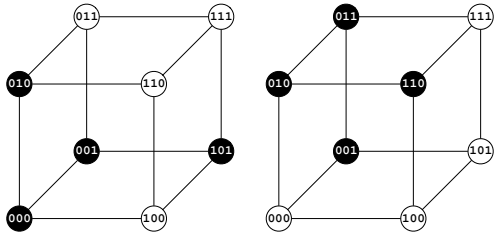
CSE 370, Autumn 2007
Benjamin Ylvisaker

Daily Quiz

- Draw Boolean cubes for these two functions. I recommend putting the input vector X=0,Y=0,Z=0 in the lower left corner, making the X axis left-right, the Y axis up-down and the Z axis depth.

$F(X,Y,Z) = \Sigma m(0,1,2,5)$ $G(X,Y,Z) = \Pi M(0,4,5,7)$

Daily Quiz Solution



Last Lecture's Daily Quiz

- In product-of-sums form, an input 1 leads to inversion and an input 0 leads to no inversion
- Don't optimize when the exercise tells you not to optimize
- Some students drew circuits that were not sum-of-products or product-of-sums form
- It's a lot faster to draw 2-level circuits without the input inverters

Where We Are

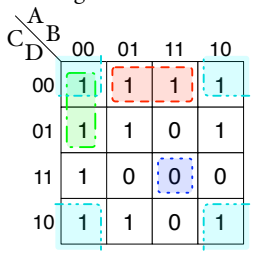
- Last lecture: Boolean cubes and K-maps
- This lecture: Minimization with K-maps
- Next lecture: Combinational Verilog
- Homework 2 due today
- Lab 2 ongoing

Two-Level Simplification

- Key tool: The uniting theorem
  - $A(B + \neg B) = A$
  - $A + (B\neg B) = A$
- We will start with lots of "big" terms, then shrink and eliminate them with the uniting theorem

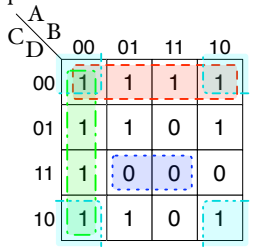
Implicants

- Any valid rectangle



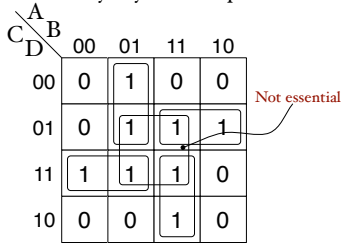
Prime Implicants

- Implicants that are not "contained" within a larger implicant



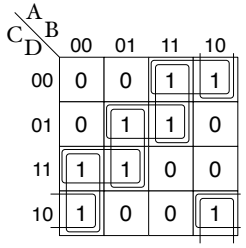
Essential Prime Implicants

- Prime implicants that cover individual squares not covered by any other implicant



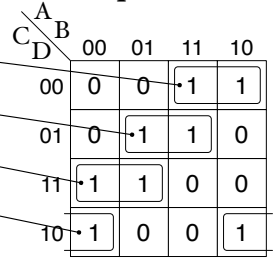
Interesting Example

- No essential prime implicants
- You choose a "cover"
  - Set of implicants
  - Together include all 1's
  - Usually prime
  - Usually non-redundant



Translate Each Implicant

- A¬C¬D
- B¬CD
- ¬ACD
- ¬BC¬D

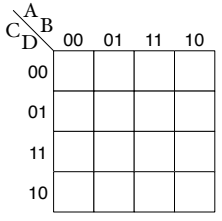


- F(A,B,C,D) = A¬C¬D + B¬CD + ¬ACD + ¬BC¬D

Example with Don't Cares

- BCD increment

A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	X	X	X	X	X
1	1	X	X	X	X	X	X



Top Bit

- BCD increment

A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	X	X	X	X	X
1	1	X	X	X	X	X	X

C D \ A B	00	01	11	10
00	0	0	X	1
01	0	0	X	0
11	0	1	X	X
10	0	0	X	X

One Way to Cover the 0's

- BCD increment

A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	X	X	X	X	X
1	1	X	X	X	X	X	X

C D \ A B	00	01	11	10
00	0	0	X	1
01	0	0	X	0
11	0	1	X	X
10	0	0	X	X

And the Resulting Expression

- BCD increment

A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	X	X	X	X	X
1	1	X	X	X	X	X	X

C D \ A B	00	01	11	10
00	0	0	X	1
01	0	0	X	0
11	0	1	X	X
10	0	0	X	X

$E = (A+C)(B+\neg D)(\neg C+D)$

Now the Second Bit

- BCD increment

A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	X	X	X	X	X
1	1	X	X	X	X	X	X

C D \ A B	00	01	11	10
00				
01				
11				
10				

Now the Second Bit

- BCD increment

A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	X	X	X	X	X
1	1	X	X	X	X	X	X

C D \ A B	00	01	11	10
00	0	1	X	0
01	0	1	X	0
11	1	0	X	X
10	0	1	X	X

Now the Second Bit

- BCD increment

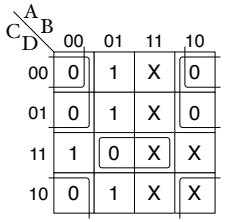
A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	X	X	X	X	X
1	1	X	X	X	X	X	X

C D \ A B	00	01	11	10
00	0	1	X	0
01	0	1	X	0
11	1	0	X	X
10	0	1	X	X

And the Resulting Expression

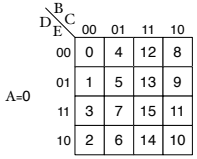
- BCD increment

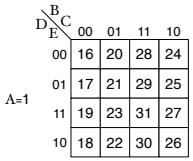
A	B	C	D	E	F	G	H
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	0	0	0	0
1	0	1	X	X	X	X	X
1	1	X	X	X	X	X	X



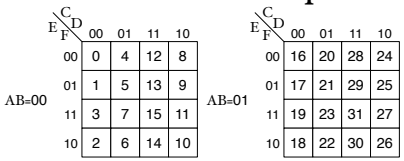
F = (B+C)(B+D)(¬B+¬C+¬D)

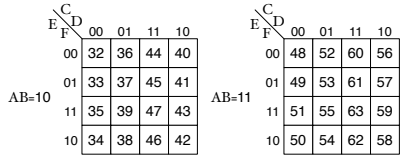
5 Variable K-maps





6 Variable K-maps





K-map Minimization Summary

- Fill out the table with 1's 0's and x's
- Find all the prime implicants
  - Try to "grow" non-prime implicants in each direction
- Select cover
  - All essential prime implicants
  - However many additional implicants are needed

Thank You for Your Attention

- Read lab 2
- Continue homework 2
- Continue reading the book