Lecture 5: 2-Level Logic and Canonical Forms

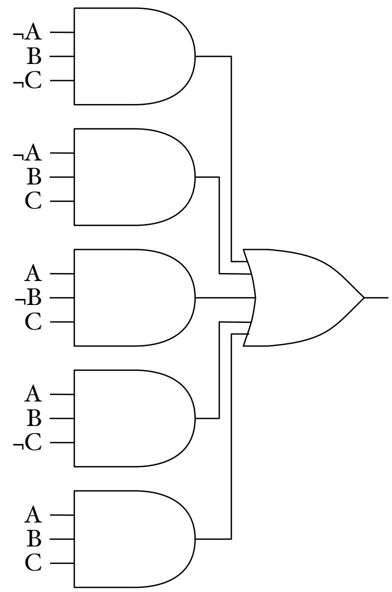
CSE 370, Autumn 2007
Benjamin Ylvisaker

Where We Are

- Last lecture: Truth tables and more functions
- This lecture: 2-level implementations and canonical forms
- Next lecture: Boolean cubes
- Homework 1 in the grading pipeline; start 2
- How was lab 2?
    - Start looking at lab 3
- Tutoring available

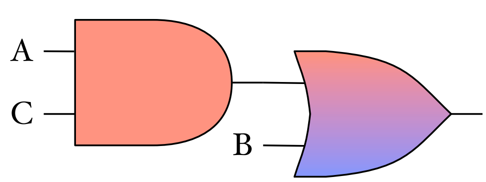
Every Function Can Be Implemented in 2 Levels

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



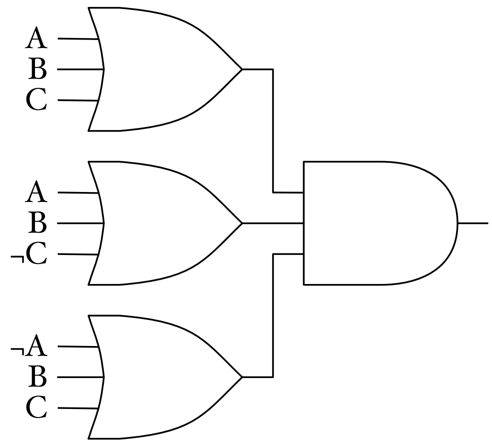
But We Can Be More Clever

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



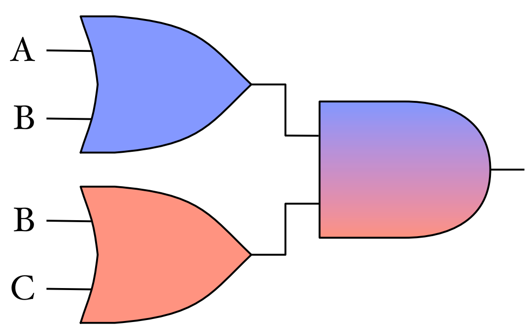
We Can Also Look At the 0's

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Again With the Cleverness

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



There Are Lots of Ways to Implement a Function

- … even if we only consider circuits of the 2 level style
- Sometimes we only want one possible representation for a given function
  - Makes it easy to decide when two people (or programs) have the same function
- Canonical forms to the rescue!

Minterms and Maxterms

Row#	A	B	AB	A+B	$\overline{AB}$	$\overline{A+B}$	$A \oplus B$	$\overline{A \oplus B}$
0	0	0	0	0	1	1	0	1
1	0	1	0	1	1	0	1	0
2	1	0	0	1	1	0	1	0
3	1	1	1	1	0	0	0	1

- $AB = \Sigma m(3) = \Pi M(0,1,2)$
  $A+B = \Sigma m(1,2,3) = \Pi M(0)$
  $\neg(AB) = \Sigma m(0,1,2) = \Pi M(3)$
  $A \oplus B = \Sigma m(1,2) = \Pi M(0,3)$

Gray Code

- In the coming examples we will use a system called gray code
- Successive numbers differ in exactly 1 bit position
- ```
  0 = 000
  1 = 001
  2 = 011
  3 = 010
  4 = 110
  5 = 111
  6 = 101
  7 = 100
  ```


Gray Code Successor Function

Input:	Output:
000	001
001	011
011	010
010	110
110	111
111	101
101	100
100	000

- We can treat each bit (each column) of the output as its own 3-variable Boolean function
  - The three functions taken together give us the complete successor
  ```

Gray Code Successor Function Truth Table

- Input: Output:

```
000     001
001     011
011     010
010     110
110     111
111     101
101     100
100     000
```

A	B	C	D	E	F
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	1	0	0
1	1	0	1	1	1
1	1	1	1	0	1

Gray Code Successor Function(s) in Minterm Notation

A	B	C	D	E	F
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	1	0	0
1	1	0	1	1	1
1	1	1	1	0	1

$D(A,B,C) = \Sigma m(2,5,6,7)$
$E(A,B,C) = \Sigma m(1,2,3,6)$
$F(A,B,C) = \Sigma m(0,1,6,7)$

Order of the variables matters!

$D(A,C,B) = \Sigma m(1,5,6,7)$
$E(A,C,B) = \Sigma m(1,2,3,5)$
$F(A,C,B) = \Sigma m(0,2,5,7)$

Now in Maxterm Notation

A	B	C	D	E	F
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	1	0	0
1	1	0	1	1	1
1	1	1	1	0	1

D(A,B,C) $= \Pi M(0,1,3,4)$
E(A,B,C) $= \Pi M(0,4,5,7)$
F(A,B,C) $= \Pi M(2,3,4,5)$

Order of the variables matters!

D(B,A,C) $= \Pi M(0,1,2,5)$
E(B,A,C) $= \Pi M(0,2,3,7)$
F(B,A,C) $= \Pi M(2,3,4,5)$

Binary-Coded Decimal (BCD)

- BCD is an encoding for more directly representing decimal numbers with binary digits
- Each 4 bits represents 1 decimal digit
- Useful in some numerical programs
- 0 = 0000 5 = 0101
  1 = 0001 6 = 0110
  2 = 0010 7 = 0111
  3 = 0011 8 = 1000
  4 = 0100 9 = 1001

BCD to Gray Code Converter

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

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

We Can Compact the Table

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

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

We Can Compact the Table

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

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

E(A,B,C,D) = Σm(8,9) + Σd(10-15)

E(A,B,C,D) = ΠM(0-7)ΠD(10-15)

We Can Compact the Table

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

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

F(A,B,C,D) = Σm(4-9) + Σd(10-15)

F(A,B,C,D) = ΠM(0-3)ΠD(10-15)

We Can Compact the Table

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

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

$G(A,B,C,D) = \Sigma m(2\text{-}5) + \Sigma d(10\text{-}15)$

$G(A,B,C,D) = \Pi M(0,1,6\text{-}9)\Pi D(10\text{-}15)$

We Can Compact the Table

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

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

$H(A,B,C,D) = \Sigma m(1,2,5,6,9) + \Sigma d(10\text{-}15)$

$H(A,B,C,D) = \Pi M(0,3,4,7,8)\Pi D(10\text{-}15)$

Lots of Representations

- Boolean algebra expressions/functions
- Digital circuit diagrams
- Truth tables
- Minterm and maxterm notation
- Next time: Boolean cubes & Karnaugh maps
- BDDs: {Boolean/Binary} Decision Diagrams
  - Not discussed in 370

Thank You for Your Attention

- Collect your quizzes
- Continue work on homework 2
- Start looking at lab 2
- Continue reading the book