Logic Design I (17.341)

Fall 2011

Lecture Outline

Class # 05

October 17, 2011

Dohn Bowden

Today's Lecture

- Administrative
- Main Logic Topic
- Homework

Course Admin

Administrative

- Admin for tonight …
  - Syllabus Review

Syllabus

- Syllabus
  - Moved the Lab Lecture to next week

Syllabus Review

Week	Date	Topics	Chapter	Lab Report Due
~~1~~	~~09/12/11~~	~~Introduction to digital systems and number systems~~	~~1~~	
~~2~~	~~09/19/11~~	~~**Binary Codes and** Boolean Algebra~~	~~2~~	
~~3~~	~~09/26/11~~	~~Boolean Algebra (continued)~~	~~3~~	
~~4~~	~~10/03/11~~	~~**Examination 1**~~		
~~**X**~~	~~**10/10/11**~~	~~**No Class - Holiday**~~		
5	10/17/11	Application of Boolean Algebra	4	
6	10/24/11	Karnaugh Maps and **Lab lecture**	5	
7	10/31/11	Multi-Level Gate Circuits. NAND and NOR Gates	7	1
8	11/07/11	**Examination 2**		
9	11/14/11	Combinational Circuit Design and Simulation Using Gates	8	2
10	11/23/11	Multiplexers, Decoders. Encoder, and PLD	9	
11	11/28/11	Introduction to VHDL	10	3
12	12/05/11	**Examination 3**		
13	12/12/11	Review		4
14	12/19/11	**Final Exam**		

Exam #1

- Exam #1
  - Will not be giving back this week as we have a student with an excused absence who need to take the exam

Questions?

Chapter 4 ...

APPLICATIONS OF BOOLEAN ALGEBRA
MINTERM AND MAXTERM EXPANSIONS

Applying Boolean Algebra

- Applying Boolean Algebra learned thus far to the design of combinational logic circuits

- The three main steps in designing a single-output combinational switching circuit are …

    1. Find a switching function that specifies the desired behavior of the circuit

    2. Find a simplified algebraic expression for the function

    3. Realize the simplified function using available logic elements

Conversion of English Sentences to Boolean Equations

Conversion of English Sentences to Boolean Equations

- For simple problems … may go directly from a word description of the desired behavior of the circuit … to …
  - An algebraic expression for the output function
- In other cases … first specify the function by means of …
  - A truth table … and then … derive an algebraic expression from the truth table

Conversion of English Sentences to Boolean Equations

- First step in designing a logic circuit is to …
  - Translate sentences into Boolean equations … by …
    - Breaking down each sentence into phrases … and …
      - Associate a Boolean variable with each phrase

Conversion of English Sentences to Boolean Equations

- If a phrase can have a value of … true or false …
  - Then we can represent that phrase by a Boolean variable
  - Phrases such as …
    - "she goes to the store"
    - " today is Monday"
      - Can be either true or false
  - But a command like "go to the store" has no truth value

Conversion of English Sentences to Boolean Equations

- If a sentence has several phrases … mark each phrase with a brace
- For example … the following sentence

Mary watches TV if it is Monday night and she has finished her homework

Conversion of English Sentences to Boolean Equations

Mary watches TV if it is Monday night and she has finished her homework

- Has three phrases …

Conversion of English Sentences to Boolean Equations

Mary watches TV if it is Monday night and she has finished her homework

- Has three phrases …

Conversion of English Sentences to Boolean Equations

Mary watches TV if it is Monday night and she has finished her homework

- Has three phrases …

Conversion of English Sentences to Boolean Equations

Mary watches TV if it is Monday night and she has finished her homework

- Has three phrases …

English Sentences to Boolean Equation
Example

Mary watches TV if it is Monday night and she has finished her homework

- Define a two-valued variable to indicate the truth or falsity of each phrase …

English Sentences to Boolean Equation
Example

Mary watches TV if it is Monday night and she has finished her homework

F

- Define a two-valued variable to indicate the truth or falsity of each phrase …
  - F = 1 … if "Mary watches TV" is true … otherwise … F = 0

English Sentences to Boolean Equation
Example

Mary watches TV if it is Monday night and she has finished her homework

F A

- Define a two-valued variable to indicate the truth or falsity of each phrase …
  - F = 1 … if "Mary watches TV" is true … otherwise … F = 0
  - A = 1 … if "it is Monday night" is true … otherwise … A = 0

English Sentences to Boolean Equation
Example

Mary watches TV if it is Monday night and she has finished her homework

F A B

- Define a two-valued variable to indicate the truth or falsity of each phrase …
  - F = 1 … if "Mary watches TV" is true … otherwise … F = 0
  - A = 1 … if "it is Monday night" is true … otherwise … A = 0
  - B = 1 … if "she has finished her homework" is true … otherwise B = 0

English Sentences to Boolean Equation
Example

Mary watches TV if it is Monday night and she has finished her homework

F   A   B

- Define a two-valued variable to indicate the truth or falsity of each phrase …
  - F = 1 … if "Mary watches TV" is true … otherwise … F = 0
  - A = 1 … if "it is Monday night" is true … otherwise … A = 0
  - B = 1 … if "she has finished her homework" is true … otherwise B = 0
  - Because F is "true" if A and B are both "true" …

$F = A \bullet B$

Example **... Word statement of a problem directly to an algebraic expression which represents the desired circuit behavior**

English Sentences to Boolean Equation
Example

The alarm will ring iff the alarm switch is turned on and …

the door is not closed, or it is after 6 PM and the window is not closed

- Word statement to circuit

English Sentences to Boolean Equation
Example

The alarm will ring iff the alarm switch is turned on and ...

the door is not closed, or it is after 6 PM and the window is not closed

- First ... associate a Boolean variable with each phrase in the above sentence ...

English Sentences to Boolean Equation
Example

The alarm will ring iff the alarm switch is turned on and ...

the door is not closed, or it is after 6 PM and the window is not closed

- First ... associate a Boolean variable with each phrase in the above sentence ... the value will have ...

    A value of 1 ... when the phrase is true

    A value of 0 ... when it is false

English Sentences to Boolean Equation
Example

The alarm will ring iff the alarm switch is turned on and ...

Z

the door is not closed, or it is after 6 PM and the window is not closed

- First ... associate a Boolean variable with each phrase of the above sentence ... the assignment implies ...

  Z = 1 ... the alarm will ring

English Sentences to Boolean Equation

Example

The alarm will ring iff the alarm switch is turned on and ...

Z (The alarm will ring) — A (the alarm switch is turned on)

the door is not closed, or it is after 6 PM and the window is not closed

- First ... associate a Boolean variable with each phrase of the above sentence ... the assignment implies ...

A = 1 ... the alarm switch is turned on ... and ...

English Sentences to Boolean Equation
Example

The alarm will ring iff the alarm switch is turned on and ...

Z A

the door is not closed, or it is after 6 PM and the window is not closed

C

- First ... associate a Boolean variable with each phrase of the above sentence ... the assignment implies ...

A = 1 ... the alarm switch is turned on ... and ...

it is after 6 PM ... therefore ... C = 1

English Sentences to Boolean Equation
Example

The alarm will ring iff the alarm switch is turned on and ...

Z   A

the door is not closed, or it is after 6 PM and the window is not closed

B'   C

- First ... associate a Boolean variable with each phrase of the above sentence ... the assignment implies ...

    B = 1 ... the door is closed ... therefore ...

    B' = 1 (B = 0) ... the door is not closed

English Sentences to Boolean Equation
Example

The alarm will ring iff the alarm switch is turned on and ...

Z A

the door is not closed, or it is after 6 PM and the window is not closed

B' C D'

- First ... associate a Boolean variable with each phrase of the above sentence ... the assignment implies ...

D = 1 ... the window is closed ... therefore ...

D′ = 1 (D = 0) ... the window is not closed

English Sentences to Boolean Equation
Example

The alarm will ring iff the alarm switch is turned on and …

Z A

the door is not closed, or it is after 6 PM and the window is not closed

B' C D'

- Summing up .. we now have …

  Z = 1 … the alarm will ring

  A = 1 … the alarm switch is turned on … and …

  it is after 6 PM … therefore … C = 1

  B′ = 1 (B = 0) … the door is not closed

  D′ = 1 (D = 0) … the window is not closed

English Sentences to Boolean Equation
Example

- Using the following assignment of variables …

  Z = 1 … the alarm will ring

  A = 1 … the alarm switch is turned on … and …

  it is after 6 PM … therefore … C = 1

  B′ = 1 (B = 0) … the door is not closed

  D′ = 1 (D = 0) … the window is not closed

The alarm will ring iff the alarm switch is turned on and …
the door is not closed, or it is after 6 PM and the window is not closed

- The sentence can be translated into the following Boolean equation

$$Z = AB' + CD'$$

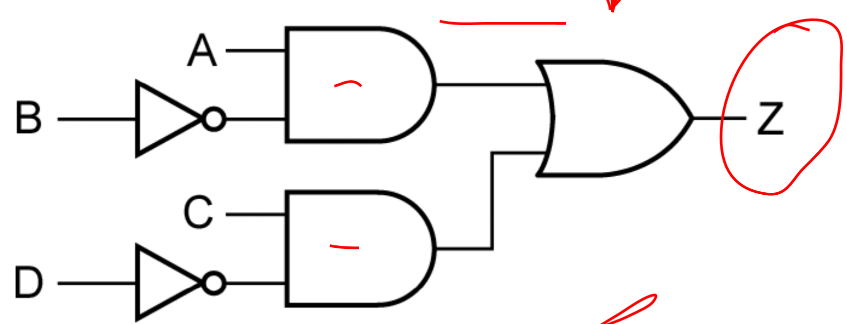
English Sentences to Boolean Equation
Example

- Finally … the sentence …

The alarm will ring iff the alarm switch is turned on and …
the door is not closed, or it is after 6 PM and the window is not closed

- And the resulting Boolean equation … will translate to the below circuit

$$Z = AB' + CD'$$



Combinational Logic Design Using a Truth Table

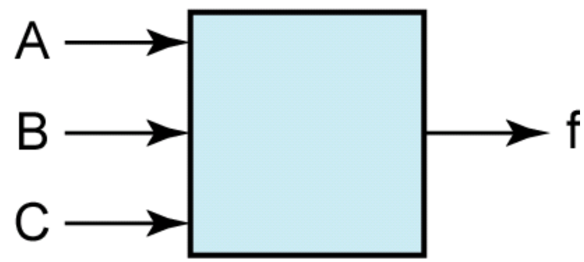
Combinational Logic Design Using a Truth Table

- Logic design using a truth table …
  - First translate the word description into a truth table
  - Then … derive and algebraic equation
  - Two standard algebraic forms of the function can be derived …
    - Standard sum of products (minterm expansion)
    - Standard product of sums (maxterm expansion)
  - Lastly the realization of the circuit using AND and OR gates

Combinational Logic Design Using a Truth Table – Example

- Illustration of Logic design using a truth table via an example …

- A Switching circuit that has … three inputs (A, B, C represent input bits of a binary number N) … and … one output



(a)

- Suppose we want the output of a circuit to be …

$$f = 1 \text{ if } N \geq 011_2 \text{ and } f = 0 \text{ if } N < 011_2$$

Combinational Logic Design Using a Truth Table – Example

- Using the criteria specified …

$f = 1$ if $N \geq 011_2$ and $f = 0$ if $N < 011_2$

- The truth table can be developed …

A	B	C	f	f′
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

(b)

Combinational Logic Design Using a Truth Table – Example

- Next, we will derive an algebraic expression for *f* from the truth table by …
  - For example … using the combinations of values of A, B, and C for which *f* = 1 … the term A′BC is 1 only if A = 0, B = 1, and C = 1

A'BC

A	B	C	f	f′
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

(b)

Combinational Logic Design Using a Truth Table – Example

- Using the ***White Board*** … Find all terms in the truth table such that *f* = 1 and ORing them together

A	B	C	f	f'
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

(b)

Combinational Logic Design Using a Truth Table – Example

- From the Whiteboard we derived …

$$f = A'BC + AB'C' + AB'C + ABC' + ABC$$

- The equation was simplified by first combining terms and then eliminating A′ to …

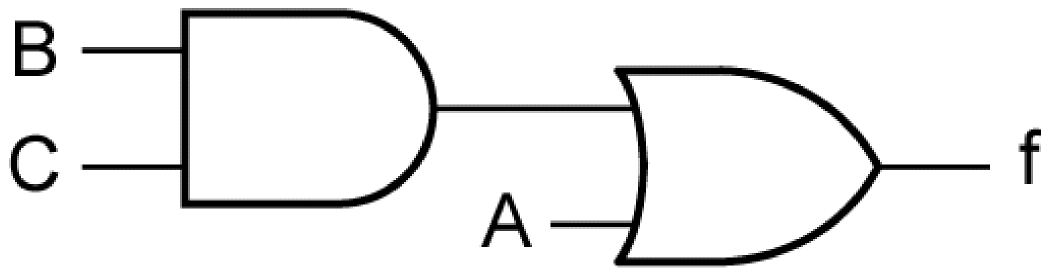
$$f = A'BC + AB' + AB = A'BC + A = A + BC$$

Combinational Logic Design Using a Truth Table – Example

- This equation …

$$f = A'BC + AB' + AB = A'BC + A = \mathbf{A + BC}$$

- Which leads directly to the following circuit …



Algebraic Expression for

f = 0 vice f = 1

Combinational Logic Design Using a Truth Table – Example (f = 0)

- Instead of writing *f* in terms of the 1's of the function as we just did in the last example …
  - We may also write *f* in terms of the 0's of the function
- Observe that the term A + B + C is 0 only if A = B = C = 0

A	*B*	*C*	*f*	*f′*
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

(b)

Combinational Logic Design Using a Truth Table – Example (f = 0)

- Using the ***White Board*** … Find all terms in the truth table such that *f* = 0 and ANDing all of these '0' terms together …

A	*B*	*C*	*f*	*f'*
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

(b)

Combinational Logic Design Using a Truth Table – Example

- From the Whiteboard we derived …

$$f = (A + B + C)(A + B + C')(A + B' + C)$$

- The equation was simplified by first combining terms and then using the second distributive law to …

$$f = (A + B)(A + B' + C) = A + B(B' + C) = A + BC$$

- Same result as we previously obtained

Minterm and Maxterm Expansions

Minterm Expansions

Minterm Expansions

- Each of the terms in the equation from our prior example …

$$f = A'BC + AB'C' + AB'C + ABC' + ABC$$

- Recall that a *literal* is a variable or its complement
- Is referred to as a *minterm* …
  - In general … a *minterm* of *n* variables is …
    - A product of *n* *literals* in which …
      - Each variable appears exactly once in either true or complemented form … but not both

Minterm Expansions

- Each minterm has a value of **1** for exactly one combination of values of the variables A, B, and C
- If A = B = C = 1 … then … A′ B′ C′ = 1
- If A = B = 0 and C = 1 then … A′ B′ C = 1
- Etc …
- Minterms are often written in abbreviated form …

  A′B′C′ is designated m_0
  A′B′C is designated m_1
  etc …

Minterm Expansions

- The following Table … lists all of the minterms of the three variables A, B, and C

Row No.	$A\ B\ C$	Minterms	Maxterms
0	0 0 0	$A'B'C' = m_0$	$A + B + C = M_0$
1	0 0 1	$A'B'C = m_1$	$A + B + C' = M_1$
2	0 1 0	$A'BC' = m_2$	$A + B' + C = M_2$
3	0 1 1	$A'BC = m_3$	$A + B' + C' = M_3$
4	1 0 0	$AB'C' = m_4$	$A' + B + C = M_4$
5	1 0 1	$AB'C = m_5$	$A' + B + C' = M_5$
6	1 1 0	$ABC' = m_6$	$A' + B' + C = M_6$
7	1 1 1	$ABC = m_7$	$A' + B' + C' = M_7$

- In general … the minterm which corresponds to row i of the truth table is designated m_i … NOTE … i is usually written in decimal

Minterm Expansions

- Minterm expansion for a function is unique
- The following equation …

$$f = A'BC + AB'C' + AB'C + ABC' + ABC$$

- Can be rewritten in terms of m-notation as …

$$f(A, B, C) = m_3 + m_4 + m_5 + m_6 + m_7$$

Minterm Expansions

- The following …

$$f(A, B, C) = m_3 + m_4 + m_5 + m_6 + m_7$$

- Can be further abbreviated by listing only the decimal subscripts in the form ...

$$f(A, B, C) = \Sigma\, m(3, 4, 5, 6, 7)$$

Maxterm Expansions

Maxterm Expansions

- Each of the sum terms (factors) in the equation from our prior example …

$$f = (A + B + C)(A + B + C')(A + B' + C)$$

- Is referred to as a *maxterm* …
  - In general … a *maxterm* of n variables is …
    - A sum of n *literals* in which …
      - Each variable appears exactly once in either true or complemented form … but not both

Maxterm Expansions

- Each maxterm has a value of **0** for exactly one combination of values of the variables A, B, and C
- If A = B = C = 0 … then … A + B + C = 0
- If A = B = 0 and C = 1 then … A + B + C′ = 0
- Etc …
- Maxterms are often written in abbreviated form … M_i

Maxterm Expansions

- The following Table … lists all of the maxterms of the three variables A, B, and C

Row No.	A B C	Minterms	Maxterms
0	0 0 0	$A'B'C' = m_0$	$A + B + C = M_0$
1	0 0 1	$A'B'C = m_1$	$A + B + C' = M_1$
2	0 1 0	$A'BC' = m_2$	$A + B' + C = M_2$
3	0 1 1	$A'BC = m_3$	$A + B' + C' = M_3$
4	1 0 0	$AB'C' = m_4$	$A' + B + C = M_4$
5	1 0 1	$AB'C = m_5$	$A' + B + C' = M_5$
6	1 1 0	$ABC' = m_6$	$A' + B' + C = M_6$
7	1 1 1	$ABC = m_7$	$A' + B' + C' = M_7$

Maxterm Expansions

- Maxterm expansion for a function is unique
- The following equation …

$$f = (A + B + C)(A + B + C')(A + B' + C)$$

- Can be rewritten in terms of M-notation as …

$$f(A, B, C) = M_0\ M_1\ M_2$$

Maxterm Expansions

- The following …

$$f(A, B, C) = M_0\ M_1\ M_2$$

- Can be further abbreviated by listing only the decimal subscripts in the form ...

$$f(A, B, C) = \prod M(0, 1, 2)$$

- Where ... $\prod$ ... means a product

Examples

Minterm and Maxterm Expansions

Minterm Expansions

- Using the ***White Board*** ... Find the minterm expansion of ...

$$f(a,b,c,d) = a'(b' + d) + acd'$$

Minterm Expansions

- Using the ***White Board*** ... Find the maxterm expansion of ...

$$f(a,b,c,d) = a'(b' + d) + acd'$$

General

Minterm and Maxterm Expansions

General Minterm and Maxterm Expansions

- The following represents a truth table for a general function of three variables
- Each a_i is a constant with a value of 0 or 1

A B C	F
0 0 0	a_0
0 0 1	a_1
0 1 0	a_2
0 1 1	a_3
1 0 0	a_4
1 0 1	a_5
1 1 0	a_6
1 1 1	a_7

$$F = a_0 m_0 + a_1 m_1 + a_2 m_2 + \cdots + a_7 m_7 = \sum_{i=0}^{7} a_i m_i$$

General Minterm and Maxterm Expansions

- Minterm expansion for a general function of three variables is …

$$F = a_0 m_0 + a_1 m_1 + a_2 m_2 + \cdots + a_7 m_7 = \sum_{i=0}^{7} a_i m_i$$

- Maxterm expansion for a general function of three variables is …

$$F = (a_0 + M_0)(a_1 + M_1)(a_2 + M_2) \cdots (a_7 + M_7) = \prod_{i=0}^{7} (a_i + M_i) \quad (4\text{-}13)$$

- Terms will drop out if a_i is 0 (minterm) or 1 (maxterm)

General Minterm and Maxterm Expansions

- Table below summarizes the procedures for conversion between minterm and maxterm expansions of F and F′

DESIRED FORM

GIVEN FORM	Minterm Expansion of *F*	Maxterm Expansion of *F*	Minterm Expansion of *F′*	Maxterm Expansion of *F′*
Minterm Expansion of *F*	______________	maxterm nos. are those nos. not on the minterm list for *F*	list minterms not present in *F*	maxterm nos. are the same as minterm nos. of *F*
Maxterm Expansion of *F*	minterm nos. are those nos. not on the maxterm list for *F*	______________	minterm nos. are the same as maxterm nos. of *F*	list maxterms not present in *F*

Application of Conversion of Forms

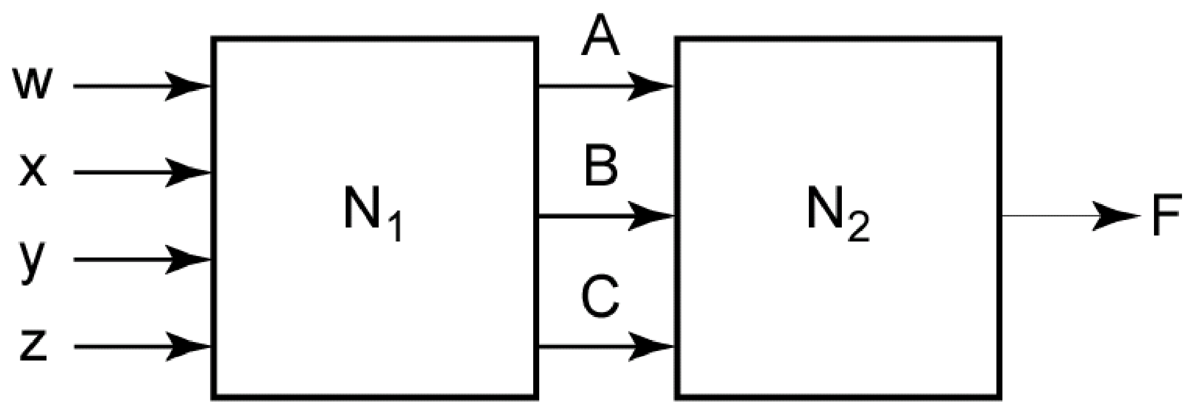
GIVEN FORM \ DESIRED FORM	Minterm Expansion of F	Maxterm Expansion of F	Minterm Expansion of F'	Maxterm Expansion of F'
Minterm Expansion of F	______	maxterm nos. are those nos. not on the minterm list for F	list minterms not present in F	maxterm nos. are the same as minterm nos. of F
Maxterm Expansion of F	minterm nos. are those nos. not on the maxterm list for F	______	minterm nos. are the same as maxterm nos. of F	list maxterms not present in F

GIVEN FORM \ DESIRED FORM	Minterm Expansion of f	Maxterm Expansion of f	Minterm Expansion of f'	Maxterm Expansion of f'
$f = \Sigma m(3, 4, 5, 6, 7)$	______	$\Pi M(0, 1, 2)$	$\Sigma m(0, 1, 2)$	$\Pi M(3, 4, 5, 6, 7)$
$f = \Pi M(0, 1, 2)$	$\Sigma m(3, 4, 5, 6, 7)$	______	$\Sigma m(0, 1, 2)$	$\Pi M(3, 4, 5, 6, 7)$

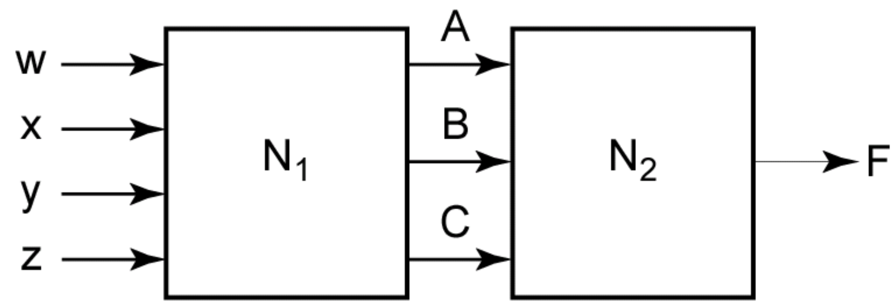
Incompletely Specified Functions

Incompletely Specified Functions

- A large digital system is usually divided into many subcircuits
- For example … the output of circuit N_1 below drives the input of circuit N_2



Incompletely Specified Functions



- Let us assume …
- The output of N_1 does not generate all possible combinations of values for *A*, *B*, and *C*
- Assume there are no combinations of values for *w, x, y,* and *z* …
- Which cause *A, B,* and *C* to assume values of 001 or 110

A B C	*F*
0 0 0	1
0 0 1	X
0 1 0	0
0 1 1	1
1 0 0	0
1 0 1	0
1 1 0	X
1 1 1	1

Incompletely Specified Functions

- When we realize the function … we must specify values for the don't-cares

- It is desirable to choose values which will help simplify the function

  – In other words …

    – Values which leads to the simplest solution

Incompletely Specified Functions

- If we assign the value 0 to both X's, then ...

$$F = A'B'C' + A'BC + ABC = A'B'C' + BC$$

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

Incompletely Specified Functions

- If we assign 1 to the first X and 0 to the second, then ...

$$F = A'B'C' + A'B'C + A'BC + ABC = A'B' + BC$$

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

Incompletely Specified Functions

- If we assign 1 to both X's, then …

$$F = A'B'C' + A'B'C + A'BC + ABC' + ABC = A'B' + BC + AB$$

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

Incompletely Specified Functions

- In summary … the second choice of values leads to the simplest solution …

- If we assign the value 0 to both X's, then

$$F = A'B'C' + A'BC + ABC = A'B'C' + BC$$

- **If we assign 1 to the first X and 0 to the second, then**

$$F = A'B'C' + A'B'C + A'BC + ABC = A'B' + BC$$

- If we assign 1 to both X's, then

$$F = A'B'C' + A'B'C + A'BC + ABC' + ABC = A'B' + BC + AB$$

Incompletely Specified Functions

- The minterm expansion for the Table is …

$$F = \Sigma\, m(0, 3, 7) + \Sigma d(1, 6)$$

- The maxterm expansion for Table is …

$$F = \Pi\, M(2, 4, 5) \bullet \Pi\, D\,(1, 6)$$

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

Examples
of
Truth Table Construction

Examples of Truth Table Construction

- Example … Design a simple binary adder that …
  - Adds two 1-bit binary numbers … *a* and *b …* to give … a 2-bit sum
  - The numeric values for the adder inputs and outputs are as follows …

a	b	Sum
0	0	00 $(0 + 0 = 0)$
0	1	01 $(0 + 1 = 1)$
1	0	01 $(1 + 0 = 1)$
1	1	10 $(1 + 1 = 2)$

Examples of Truth Table Construction

- Represent inputs to the adder by …
  - The logic variables *A* and *B*
- Represent the 2-bit sum by …
  - The logic variables *X* and *Y*

Examples of Truth Table Construction

- Because a numeric value of 0 is represented by a logic 0 … and …
- A numeric value of 1 by a logic 1 …
  - The 0's and 1's in the truth table are exactly the same as in the original table
- Resulting truth table …

A	*B*	*X*	*Y*
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Original Table …

a	*b*	Sum	
0	0	00	$(0 + 0 = 0)$
0	1	01	$(0 + 1 = 1)$
1	0	01	$(1 + 0 = 1)$
1	1	10	$(1 + 1 = 2)$

Examples of Truth Table Construction

A	B	X	Y
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

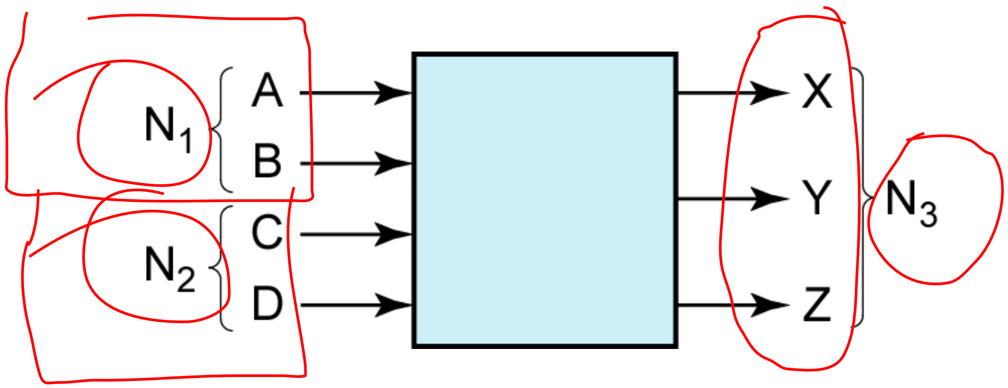
- From the truth table …

$$X = AB \text{ and } Y = A'B + AB' = A \oplus B$$

Example … Design an adder

Example --- Design an adder

- Design an adder which adds two 2-bit binary numbers to give a 3-bit binary sum. Find the truth table for the circuit. The circuit has four inputs and three outputs as shown …



Example --- Design an adder

- Inputs A and B taken together represent a binary number N_1
- Inputs C and D taken together represent a binary number N_2
- Outputs X, Y, and Z taken together represent a binary number N_3 …
  - Where $N_3 = N_1 + N_2$
    - … the “+” sign represents ordinary addition here

Example --- Design an adder

- Truth table is developed by performing addition of the inputs …

$$N_3 = N_1 + N_2$$

TRUTH TABLE:

N_1		N_2		N_3		
A	*B*	*C*	*D*	*X*	*Y*	*Z*
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	1
0	1	1	1	1	0	0

TRUTH TABLE:

N_1		N_2		N_3		
A	*B*	*C*	*D*	*X*	*Y*	*Z*
1	0	0	0	0	1	0
1	0	0	1	0	1	1
1	0	1	0	1	0	0
1	0	1	1	1	0	1
1	1	0	0	0	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	1
1	1	1	1	1	1	0

Example --- Design an adder

- Derive the switching functions for the output variables …
- From inspection of the table, the output functions are …

$$X(A, B, C, D) = \sum m(7, 10, 11, 13, 14, 15)$$

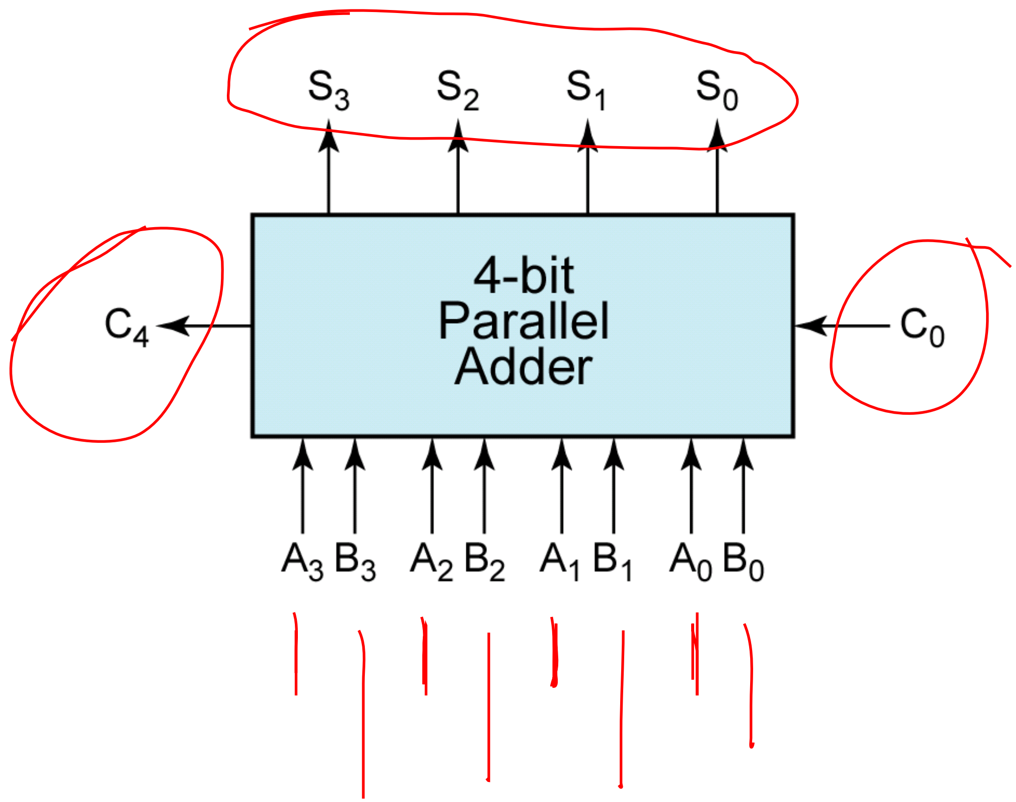
$$Y(A, B, C, D) = \sum m(2, 3, 5, 6, 8, 9, 12, 15)$$

$$Z(A, B, C, D) = \sum m(l, 3, 4, 6, 9, 11, 12, 14)$$

Design of Binary Adders and Subtractors

Design of Binary Adders and Subtractors

- We will design a parallel adder that …
  - Adds two 4-bit unsigned binary numbers … and …
  - Has a carry input to give a 4-bit sum and a carry output



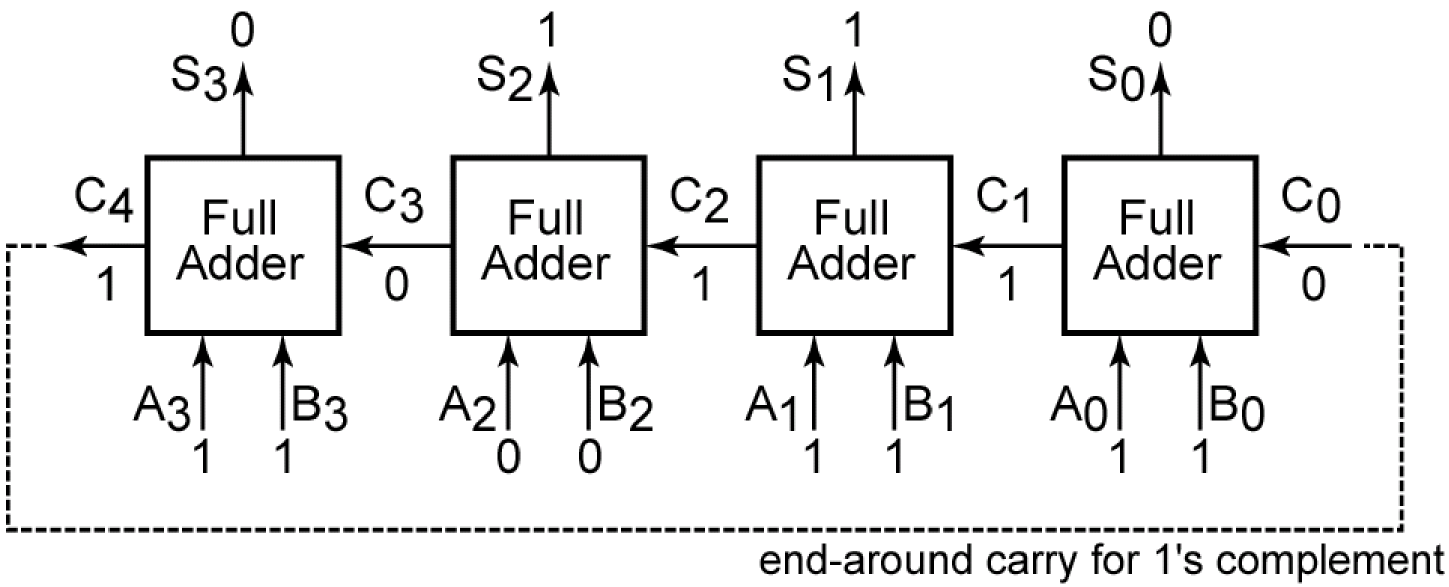
Design of Binary Adders and Subtractors

- One approach would be to …
  - Construct a truth table with …
    - Nine inputs … and … five outputs
    - Then derive and simplify the five output equations

Design of Binary Adders and Subtractors

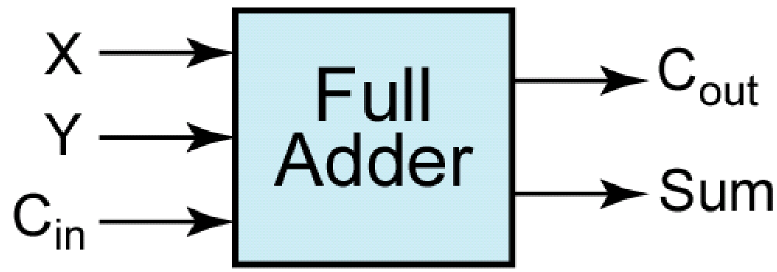
- A better method is to …
  - Design a logic module that …
    - Adds two bits and a carry … and …
    - Then connect four of these modules together to form a 4-bit adder

Design of Binary Adders and Subtractors



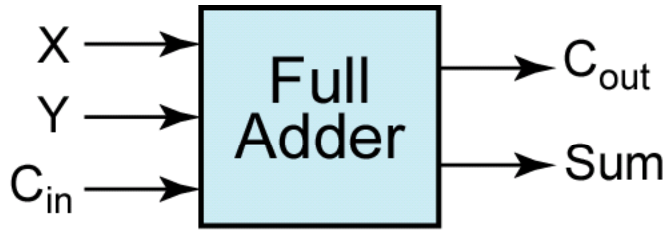
Design of Binary Adders and Subtractors

- The full adder …



- Truth table output for the full adder can be developed as folllows …
  - Each row of the table are found by adding up the input bits … and …
  - Splitting the result into a carry out … and … a sum bit

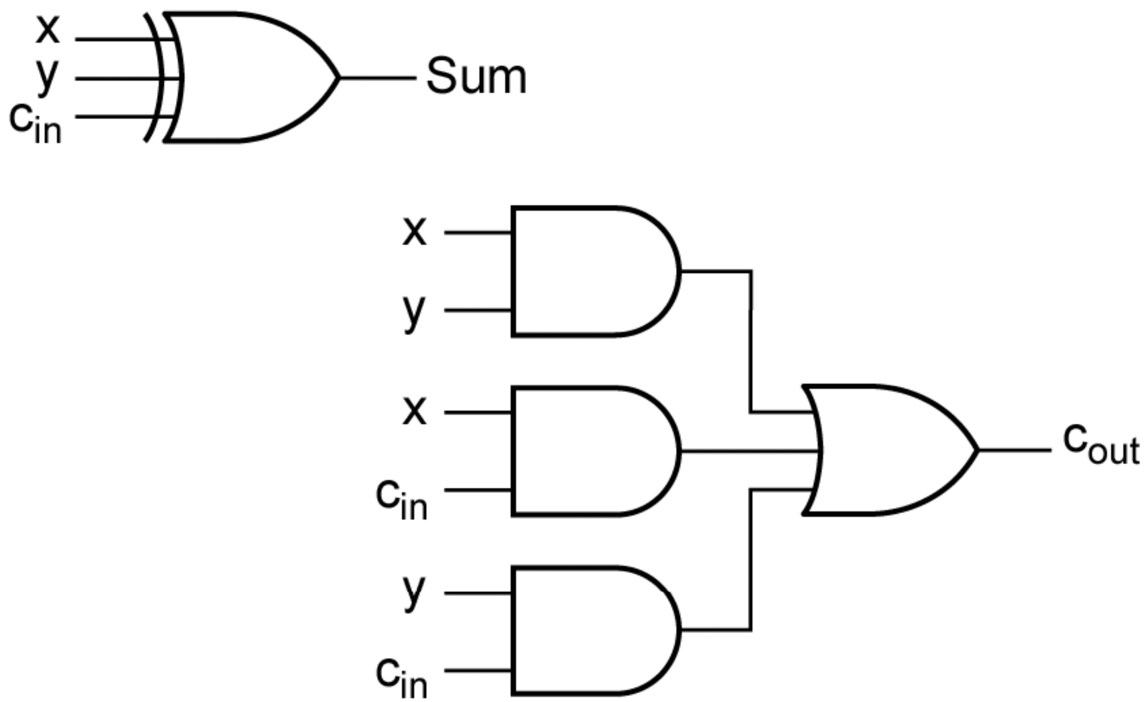
Design of Binary Adders and Subtractors



X	Y	C_{in}	C_{out}	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Design of Binary Adders and Subtractors

- Logic circuits for the equations derived are …



Design of Binary Adders and Subtractors

- Using the ***White Board*** ... derive the logic output equations for the full adder from the truth table

X	Y	C_{in}	C_{out}	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Overflow for Signed Binary Numbers

Overflow for Signed Binary Numbers

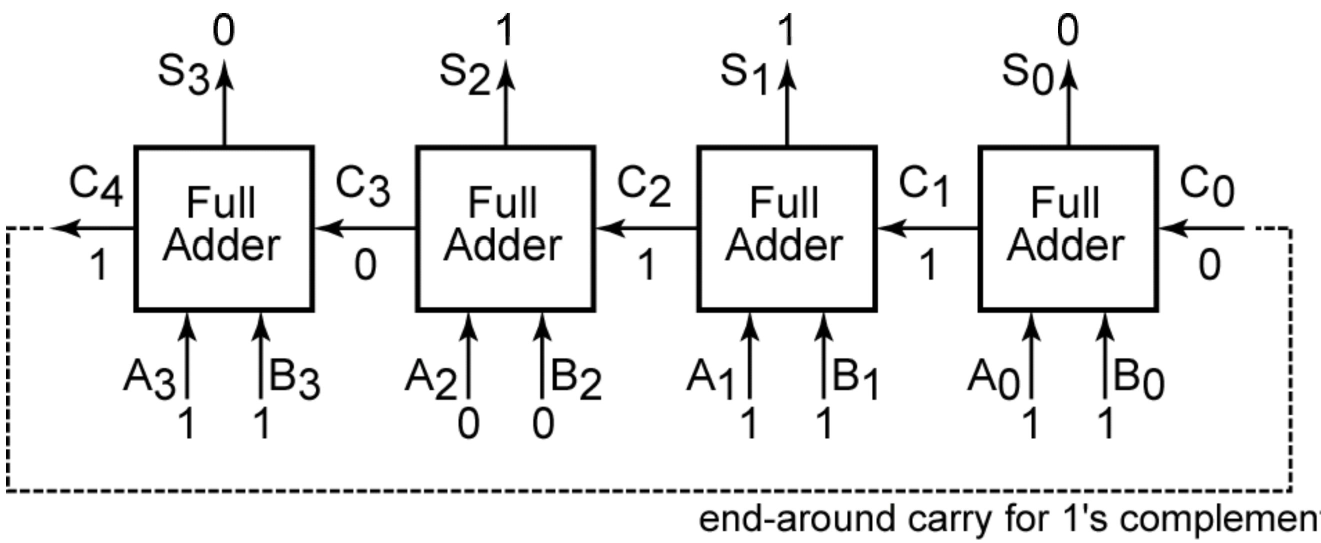
- An overflow has occurred if …
  - Adding two positive numbers gives a negative result

  … or …
  - Adding two negative numbers gives a positive result
- We define an overflow signal … V = 1 … if an overflow occurs

Overflow for Signed Binary Numbers

- For circuit below …

$$V = A_3'B_3'S_3 + A_3B_3S_3'$$

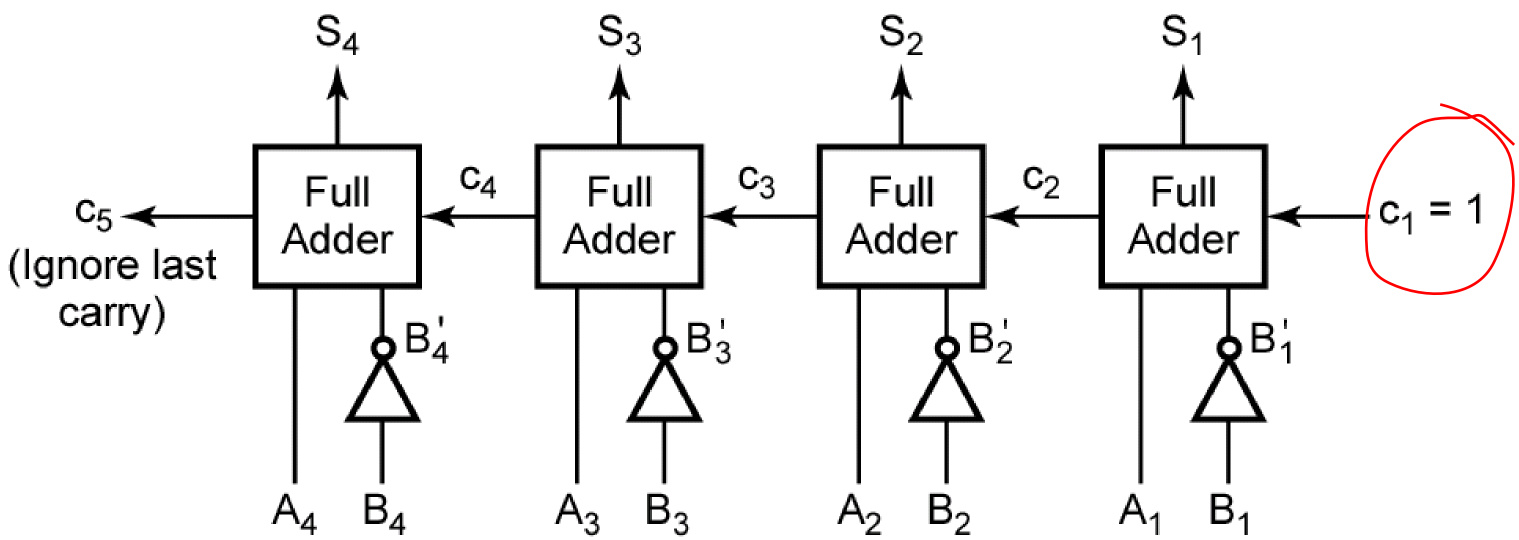


Subtraction

Subtraction

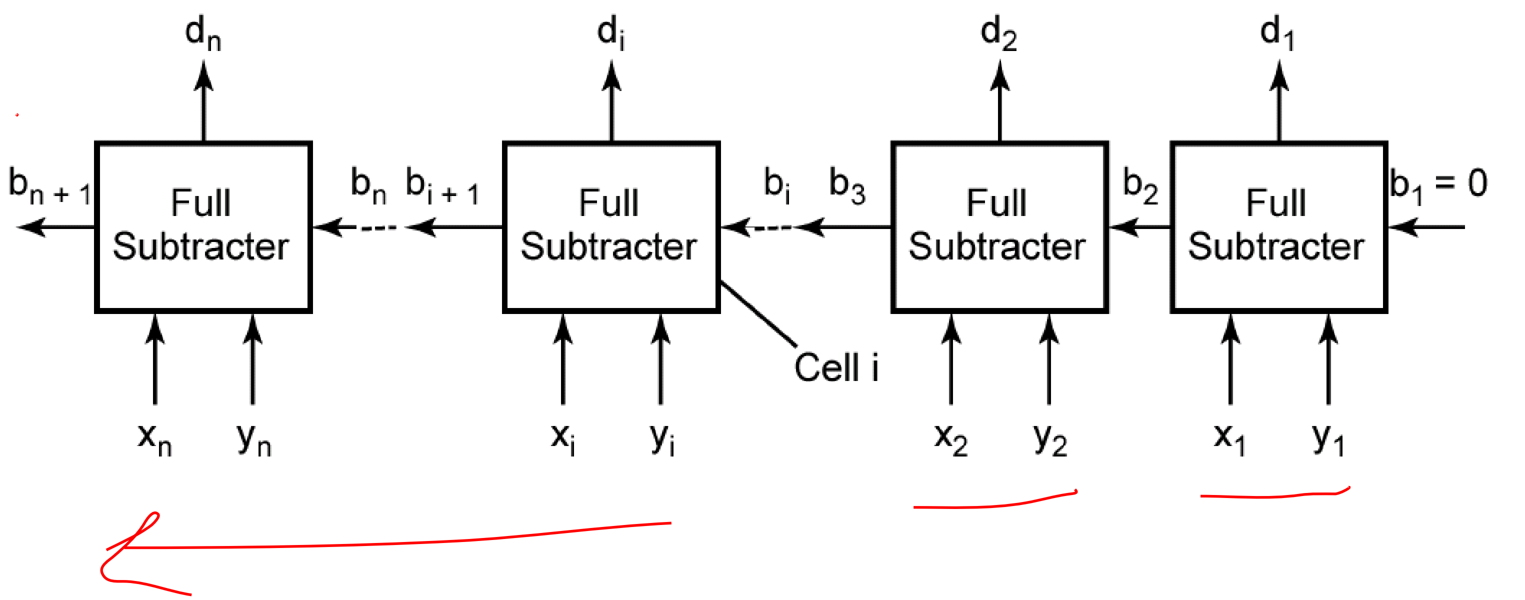
- Full Adders may be used to form A – B using the 2's complement representation for negative numbers

- The 2's complement of B can be formed by first finding the 1's complement and then adding 1 (carry input of first adder)



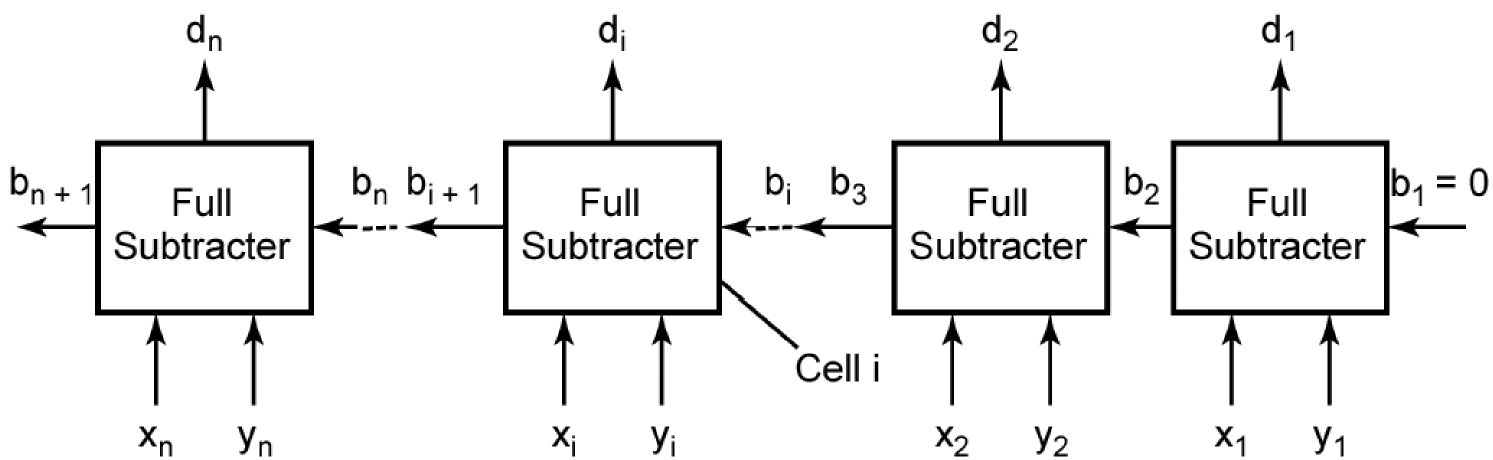
Subtraction

- Alternatively … direct subtraction can be accomplished by employing a full subtracter in a manner analogous to a full adder



Subtraction

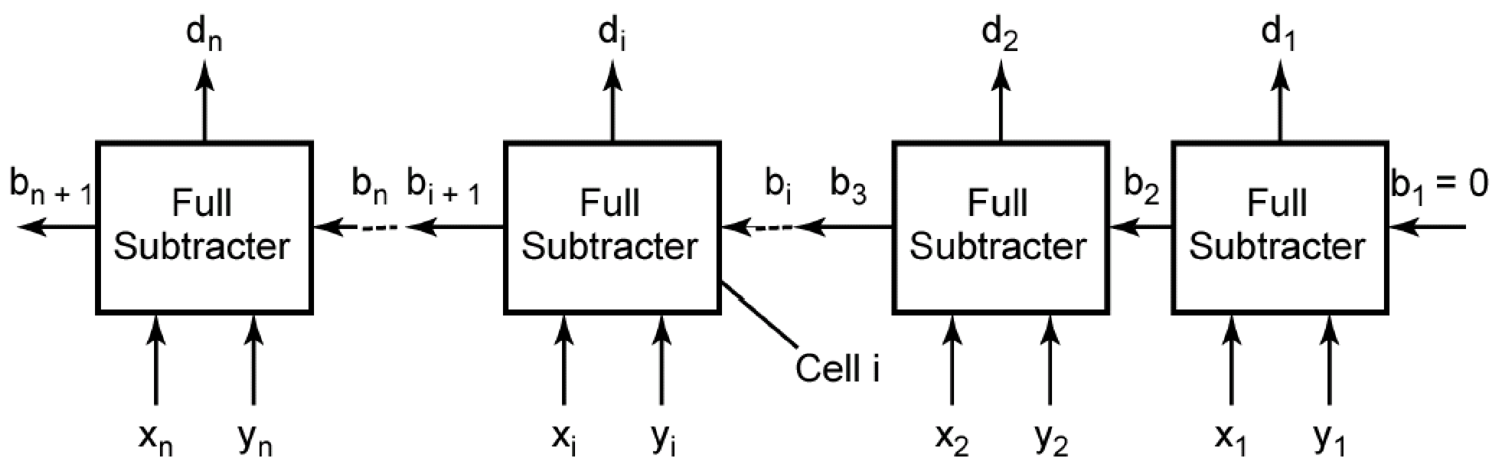
- The first two bits are subtracted in the rightmost cell to give a difference d_1 … and … a borrow signal ($b_2 = 1$) is generated … if … it is necessary to borrow from the next column



Subtraction

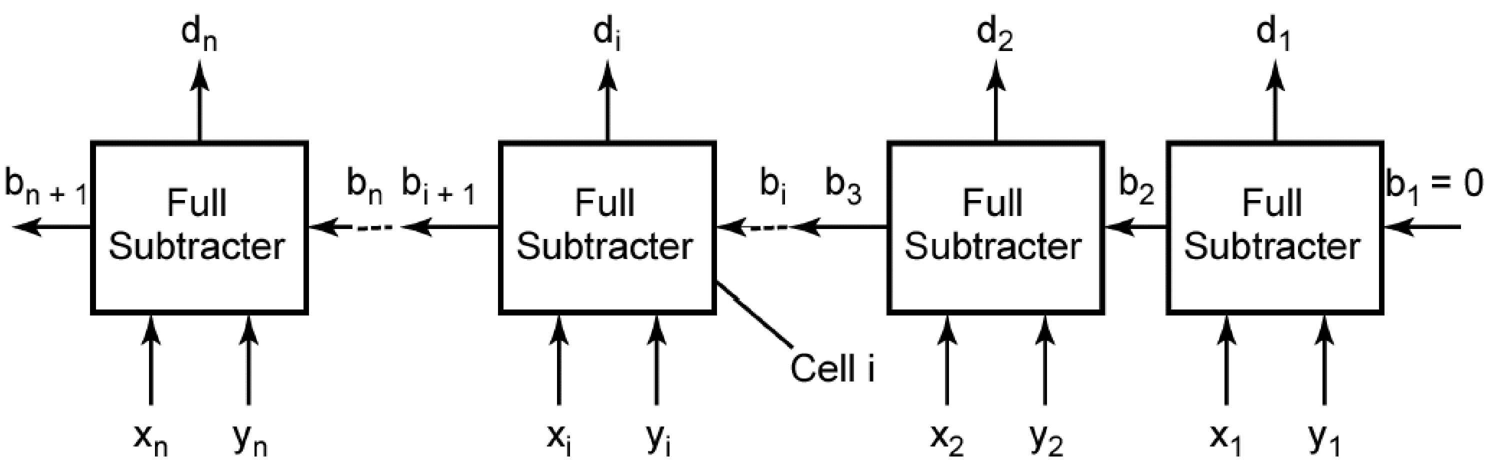
- A typical cell (cell i) has …
  - Inputs x_i, y_i, and b_i

  …and …
  - Outputs b_{i+1} and d_i



Subtraction

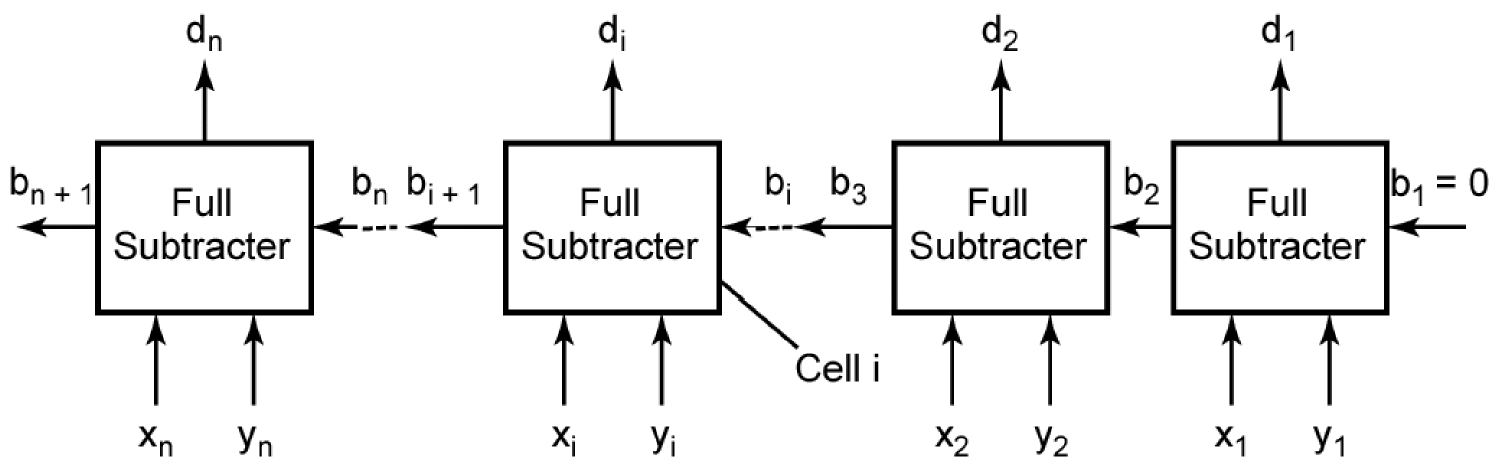
- An input $b_i = 1$ indicates that we must …
  - Borrow 1 from x_i in that cell …
    - Borrowing 1 from x_i is equivalent to subtracting 1 from x_i



Subtraction

- In cell i …
  - Bits b_i and y_i are subtracted from x_i to form the difference d_i

  … and …
  - A borrow signal ($b_{i+1} = 1$) is generated if it is necessary to borrow from the next column



Subtraction

- Truth Table for Binary Full Subtracter …

x_i	y_i	b_i	b_{i+1}	d_i
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Subtraction

- Consider $x_i = 0$ … $b_i = 1$ … and … $y_i = 1$

	Column i Before Borrow	Column i After Borrow	
x_i	0	10	
$-b_i$	-1	-1	
$-y_i$	-1	-1	
d_i		0	$(b_{i+1} = 1)$

- Note in column i … we cannot immediately subtract y_i and b_i from x_i
- Hence … we must borrow from column i + 1
- Borrowing 1 from column i + 1 is equivalent to setting b_{i+1} to 1 and adding 10 (2_{10}) to x_i
- We then have $d_i = 10 - 1 - 1 = 0$

Lab

Lab

- No topics this week

Next Week ...

Next Week Topics

- Karnaugh Maps (K-Maps)
  - Chapter 5
- Lab introduction

Home Work

Homework

1. Read Chapters 4 and 5 …

2. Solve the following Chapter 4 problems …

    - 4.1 … (a)
    - 4.2 … (a)
    - 4.7 … (a) and (b)
    - 4.9 … (a), (b), (c), and (d)

References

1. None