ECE 274 Digital Logic

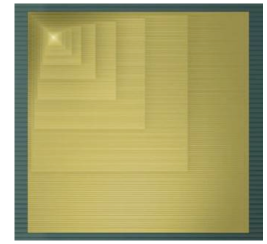
Datapath Components – Adders and Incrementers

Digital Design 4.3, 4.6

Digital Design

Chapter 4: Datapath Components

Slides to accompany the textbook *Digital Design*, First Edition, by Frank Vahid, John Wiley and Sons Publishers, 2007.
http://www.ddvahid.com



Datapath Components

Adders

- Adds two N-bit binary numbers
  - 2-bit adder: adds two 2-bit numbers, outputs 3-bit result
  - e.g., 01 + 11 = 100 (1 + 3 = 4)
- Can design using combinational logic design process, but doesn't work well for reasonable-size N
  - *Why not?*

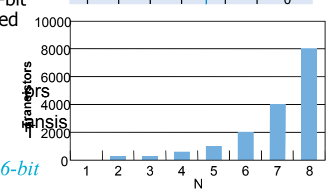
Inputs				Outputs		
a1	a0	b1	b0	c	s1	s0
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	0	1	1
1	0	1	0	1	0	0
1	0	1	1	1	0	1
1	1	0	0	0	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	1
1	1	1	1	1	1	0

Datapath Components

Why Not Use Standard Combinational Design Process

- Truth table too big
  - 2-bit adder's truth table shown
    - Has $2^{(2+2)}$ = 16 rows
  - 8-bit adder: $2^{(8+8)}$ = 65,536 rows
  - 16-bit adder: $2^{(16+16)}$ = ~4 billion rows
  - 32-bit adder: ...
- Big truth table with numerous 1s/0s yields big logic
  - Plot shows number of transistors for N-bit adders, using state-of-the-art automated combinational design tool

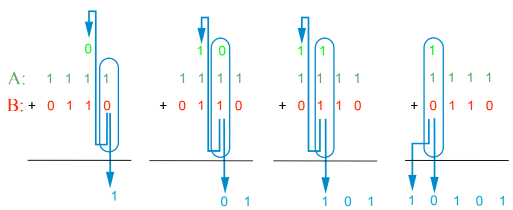
Inputs				Outputs		
a1	a0	b1	b0	c	s1	s0
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	0	1	1
1	0	1	0	1	0	0
1	0	1	1	1	0	1
1	1	0	0	0	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	1
1	1	1	1	1	1	0



How many of transistors are needed for a 16-bit adder?

Datapath Components
Alternative Method: Imitate Adding by Hand

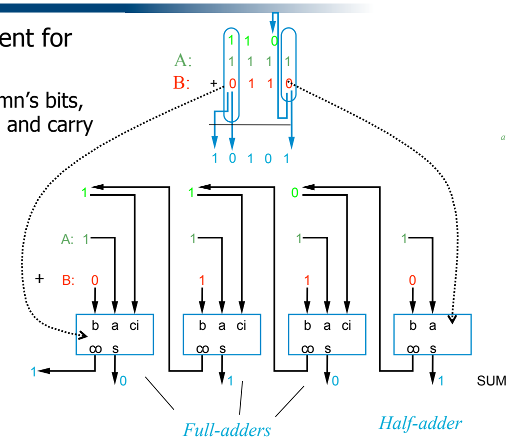
- Alternative Adder Design Method
  - Mimic how people do addition by hand
  - One column at a time
    - Compute sum, add carry to next column



5

Datapath Components
Alternative Method: Imitate Adding by Hand

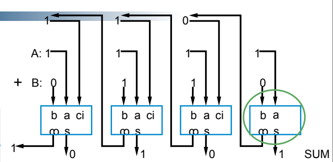
- Create component for each column
  - Adds that column's bits, generates sum and carry bits



6

Datapath Components
Half-Adder

- ***Half-adder***: Adds 2 bits, generates sum and carry
- Design using combinational design process from Ch 2



Step 1: Capture the function

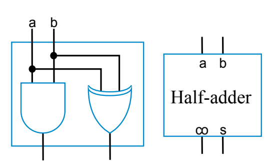
Inputs a	Inputs b	Outputs co	Outputs s
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Step 2: Convert to equations

co = ab

s = a'b + ab' (same as s = a xor b)

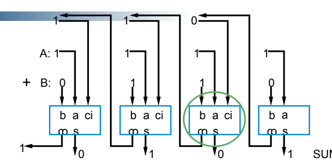
Step 3: Create the circuit



7

Datapath Components
Full-Adder

- ***Full-adder:*** Adds 3 bits, generates sum and carry
- Design using combinational design process from Ch 2



Step 1: Capture the function

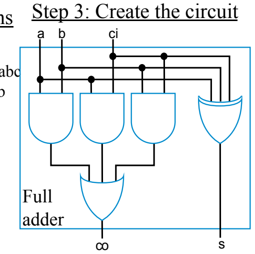
Inputs a	Inputs b	Inputs ci	Outputs co	Outputs s
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Step 2: Convert to equations

co = a'bc + ab'c + abc' + abc
co = a'bc +abc +ab'c +abc +abc' +abc
co = (a'+a)bc + (b'+b)ac + (c'+c)ab
co = bc + ac + ab

s = a'b'c + a'bc' + ab'c' + abc
s = a'(b'c + bc') + a(b'c' + bc)
s = a'(b xor c)' + a(b xor c)
s = a xor b xor c

Step 3: Create the circuit

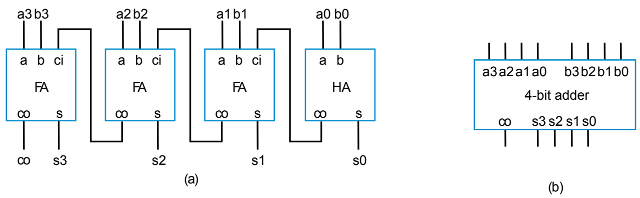


8

Datapath Components

Carry-Ripple Adder

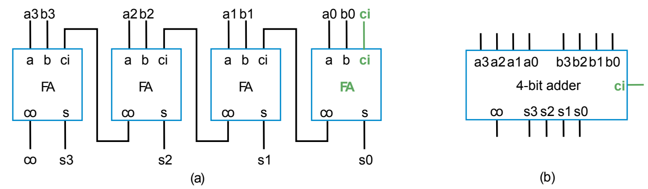
- Using half-adder and full-adders, we can build adder that adds like we would by hand
- Called a ***carry-ripple adder***
  - 4-bit adder shown: Adds two 4-bit numbers, generates 5-bit output
    - 5-bit output can be considered 4-bit "sum" plus 1-bit "carry out"
  - Can easily build any size adder



Datapath Components

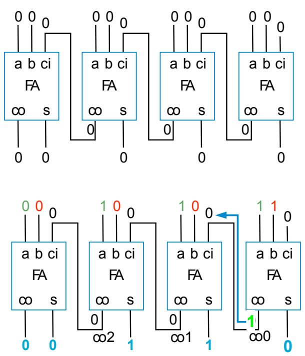
Carry-Ripple Adder

- Using full-adder instead of half-adder for first bit, we can include a "carry in" bit in the addition
  - Will be useful later when we connect smaller adders to form bigger adders



Datapath Components

Carry-Ripple Adder's Behavior



Assume all inputs initially 0

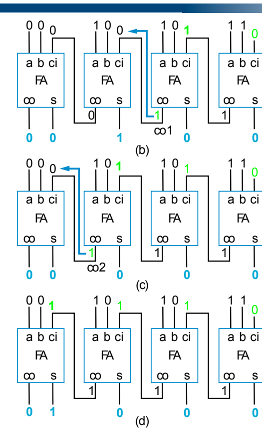
0111+0001
(answer should be 01000)

Output after 2 ns (1FA delay)

Datapath Components

Carry-Ripple Adder's Behavior

0111+0001
(answer should be 01000)



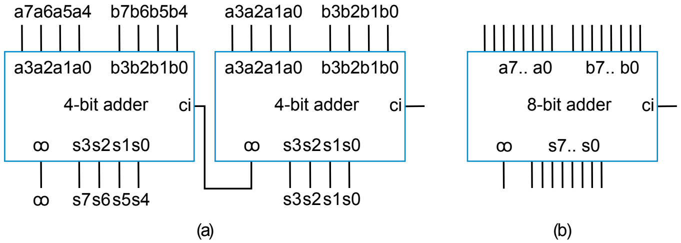
Outputs after 4ns (2 FA delays)

Outputs after 6ns (3 FA delays)

Output after 8ns (4 FA delays)

Correct answer appears after 4 FA delays

Datapath Components
Cascading Adders



Datapath Components
Incrementer

- Counter design used incrementer
- Incrementer design
  - Could use carry-ripple adder with B input set to 00...001
    - But when adding 00...001 to another number, the leading 0's obviously don't need to be considered -- so just two bits being added per column
  - Use half-adders (adds two bits) rather than full-adders (adds three bits)