

**CPE/EE 427, CPE 527  
VLSI Design I  
L10: Logical Effort**

Department of Electrical and Computer Engineering  
University of Alabama in Huntsville

Aleksandar Milenkovic ( [www.ece.uah.edu/~milenka](http://www.ece.uah.edu/~milenka) )  
[www.ece.uah.edu/~milenka/cpe527-05F](http://www.ece.uah.edu/~milenka/cpe527-05F)

**Course Administration**

- Instructor: Aleksandar Milenkovic  
[milenka@ece.uah.edu](mailto:milenka@ece.uah.edu)  
[www.ece.uah.edu/~milenka](http://www.ece.uah.edu/~milenka)  
EB 217-L  
Mon. 5:30 PM – 6:30 PM,  
Wen. 12:30 – 13:30 PM
- URL: <http://www.ece.uah.edu/~milenka/cpe527-05F>
- TA: Joel Wilder
- Labs: Lab#3: due 10/07/05
- Project: Proposals due 10/10/05
- Test I: 10/12/05
- Text: CMOS VLSI Design, 3rd ed., Weste, Harris
- Review: Chapters 1, 2, 3, 4;
- Today: Logical Effort (Chapter 4)

9/20/2005

VLSI Design I; A. Milenkovic

2

**Outline**

- Introduction
- Delay in a Logic Gate
- Multistage Logic Networks
- Choosing the Best Number of Stages
- Example
- Summary

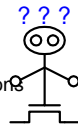
9/20/2005

VLSI Design I; A. Milenkovic

3

**Introduction**

- Chip designers face a bewildering array of choices
  - What is the best circuit topology for a function?
  - How many stages of logic give least delay?
  - How wide should the transistors be?
- Logical effort is a method to make these decisions
  - Uses a simple model of delay
  - Allows back-of-the-envelope calculations
  - Helps make rapid comparisons between alternatives
  - Emphasizes remarkable symmetries



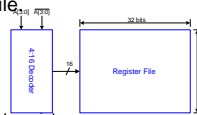
9/20/2005

VLSI Design I; A. Milenkovic

4

**Example**

- Ben Bitiddle is the memory designer for the Motorola 68W86, an embedded automotive processor. Help Ben design the decoder for a register file.
- Decoder specifications:
  - 16 word register file
  - Each word is 32 bits wide
  - Each bit presents load of 3 unit-sized transistors
  - True and complementary address inputs A[3:0]
  - Each input may drive 10 unit-sized transistors
- Ben needs to decide:
  - How many stages to use?
  - How large should each gate be?
  - How fast can decoder operate?



9/20/2005

VLSI Design I; A. Milenkovic

5

**Delay in a Logic Gate**

- Express delays in process-independent unit

$$d = \frac{d_{abs}}{\tau}$$

$$\tau = 3RC$$

$$\approx 12 \text{ ps in } 180 \text{ nm process}$$

$$40 \text{ ps in } 0.6 \mu\text{m process}$$

9/20/2005

VLSI Design I; A. Milenkovic

6

### Delay in a Logic Gate

- Express delays in process-independent unit

$$d = \frac{d_{abs}}{\tau}$$

- Delay has two components

$$d = f + p$$

9/20/2005

VLSI Design I: A. Milenkovic

7

### Delay in a Logic Gate

- Express delays in process-independent unit

$$d = \frac{d_{abs}}{\tau}$$

- Delay has two components

$$d = f + p$$

- Effort delay  $f = gh$  (a.k.a. stage effort)
  - Again has two components

9/20/2005

VLSI Design I: A. Milenkovic

8

### Delay in a Logic Gate

- Express delays in process-independent unit

$$d = \frac{d_{abs}}{\tau}$$

- Delay has two components

$$d = f + p$$

- Effort delay  $f = gh$  (a.k.a. stage effort)
  - Again has two components
- $g$ : logical effort
  - Measures relative ability of gate to deliver current
  - $g \equiv 1$  for inverter

9/20/2005

VLSI Design I: A. Milenkovic

9

### Delay in a Logic Gate

- Express delays in process-independent unit

$$d = \frac{d_{abs}}{\tau}$$

- Delay has two components

$$d = f + p$$

- Effort delay  $f = gh$  (a.k.a. stage effort)
  - Again has two components
- $h$ : electrical effort =  $C_{out} / C_{in}$ 
  - Ratio of output to input capacitance
  - Sometimes called fanout

9/20/2005

VLSI Design I: A. Milenkovic

10

### Delay in a Logic Gate

- Express delays in process-independent unit

$$d = \frac{d_{abs}}{\tau}$$

- Delay has two components

$$d = f + p$$

- Parasitic delay  $p$ 
  - Represents delay of gate driving no load
  - Set by internal parasitic capacitance

9/20/2005

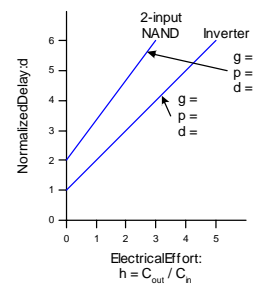
VLSI Design I: A. Milenkovic

11

### Delay Plots

$$d = f + p$$

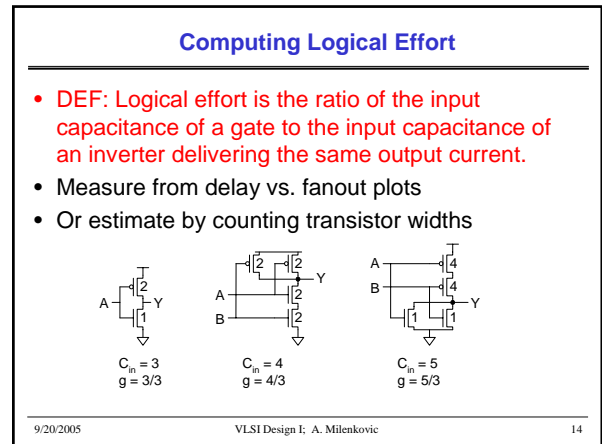
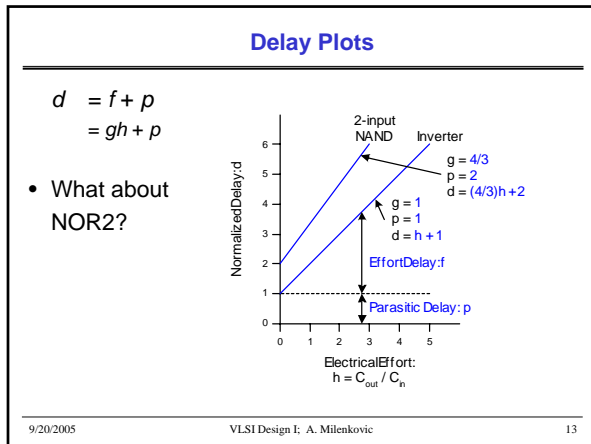
$$= gh + p$$



9/20/2005

VLSI Design I: A. Milenkovic

12



### Catalog of Gates

- Logical effort of common gates

Gate type	Number of inputs				
	1	2	3	4	n
Inverter	1				
NAND		4/3	5/3	6/3	(n+2)/3
NOR		5/3	7/3	9/3	(2n+1)/3
Tristate / mux	2	2	2	2	2
XOR, XNOR		4, 4	6, 12, 6	8, 16, 16, 8	

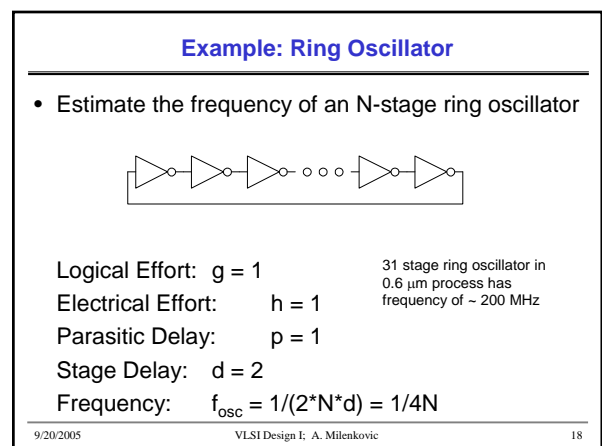
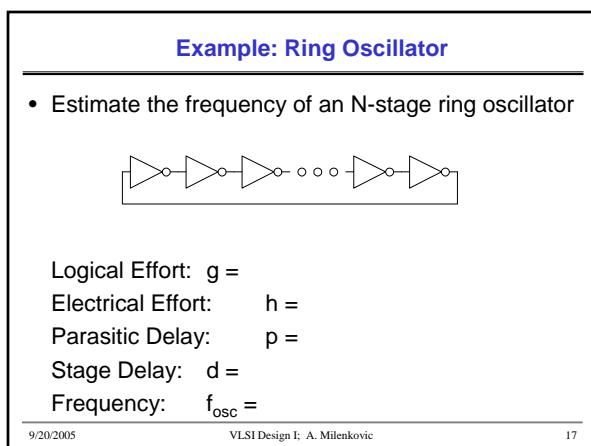
9/20/2005 VLSI Design I: A. Milenkovic 15

### Catalog of Gates

- Parasitic delay of common gates  
– In multiples of  $p_{inv}$  ( $\approx 1$ )

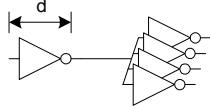
Gate type	Number of inputs				
	1	2	3	4	n
Inverter	1				
NAND		2	3	4	n
NOR		2	3	4	n
Tristate / mux	2	4	6	8	2n
XOR, XNOR		4	6	8	

9/20/2005 VLSI Design I: A. Milenkovic 16



### Example: FO4 Inverter

- Estimate the delay of a fanout-of-4 (FO4) inverter



Logical Effort:  $g =$   
 Electrical Effort:  $h =$   
 Parasitic Delay:  $p =$   
 Stage Delay:  $d =$

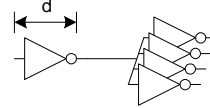
9/20/2005

VLSI Design I: A. Milenkovic

19

### Example: FO4 Inverter

- Estimate the delay of a fanout-of-4 (FO4) inverter



Logical Effort:  $g = 1$   
 Electrical Effort:  $h = 4$   
 Parasitic Delay:  $p = 1$   
 Stage Delay:  $d = 5$

The FO4 delay is about  
 200 ps in a 0.6  $\mu\text{m}$  process  
 60 ps in a 180 nm process  
 1/3 ns in a 1  $\mu\text{m}$  process

9/20/2005

VLSI Design I: A. Milenkovic

20

### Multistage Logic Networks

- Logical effort generalizes to multistage networks
- Path Logical Effort

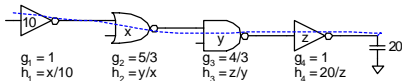
$$G = \prod g_i$$

- Path Electrical Effort

$$H = \frac{C_{\text{out-path}}}{C_{\text{in-path}}}$$

- Path Effort

$$F = \prod f_i = \prod g_i h_i$$



9/20/2005

VLSI Design I: A. Milenkovic

21

### Multistage Logic Networks

- Logical effort generalizes to multistage networks
- Path Logical Effort

$$G = \prod g_i$$

- Path Electrical Effort

$$H = \frac{C_{\text{out-path}}}{C_{\text{in-path}}}$$

- Path Effort

$$F = \prod f_i = \prod g_i h_i$$

- Can we write  $F = GH$ ?

9/20/2005

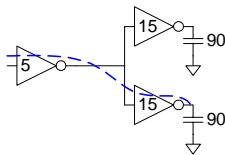
VLSI Design I: A. Milenkovic

22

### Paths that Branch

- No! Consider paths that branch:

$G =$   
 $H =$   
 $GH =$   
 $h_1 =$   
 $h_2 =$   
 $F = GH?$



9/20/2005

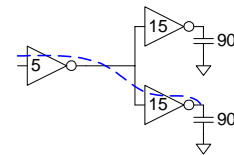
VLSI Design I: A. Milenkovic

23

### Paths that Branch

- No! Consider paths that branch:

$G = 1$   
 $H = 90 / 5 = 18$   
 $GH = 18$   
 $h_1 = (15 + 15) / 5 = 6$   
 $h_2 = 90 / 15 = 6$   
 $F = g_1 g_2 h_1 h_2 = 36 = 2GH$



9/20/2005

VLSI Design I: A. Milenkovic

24

### Branching Effort

- Introduce *branching effort*
  - Accounts for branching between stages in path

$$b = \frac{C_{\text{on path}} + C_{\text{off path}}}{C_{\text{on path}}}$$

$$B = \prod b_i$$

Note:

$$\prod h_i = BH$$

- Now we compute the path effort
  - $F = GBH$

9/20/2005

VLSI Design I: A. Milenkovic

25

### Multistage Delays

- Path Effort Delay  $D_F = \sum f_i$
- Path Parasitic Delay  $P = \sum P_i$
- Path Delay  $D = \sum d_i = D_F + P$

9/20/2005

VLSI Design I: A. Milenkovic

26

### Designing Fast Circuits

$$D = \sum d_i = D_F + P$$

- Delay is smallest when each stage bears same effort

$$\hat{f} = g_i h_i = F^{\frac{1}{N}}$$

- Thus minimum delay of N stage path is

$$D = NF^{\frac{1}{N}} + P$$

- This is a **key** result of logical effort
  - Find fastest possible delay
  - Doesn't require calculating gate sizes

9/20/2005

VLSI Design I: A. Milenkovic

27

### Gate Sizes

- How wide should the gates be for least delay?

$$\hat{f} = gh = g \frac{C_{out}}{C_{in}}$$

$$\Rightarrow C_{in_i} = \frac{g_i C_{out_i}}{\hat{f}}$$

- Working backward, apply capacitance transformation to find input capacitance of each gate given load it drives.
- Check work by verifying input cap spec is met.

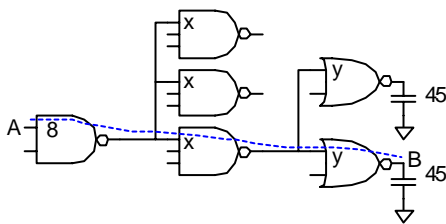
9/20/2005

VLSI Design I: A. Milenkovic

28

### Example: 3-stage path

- Select gate sizes x and y for least delay from A to B

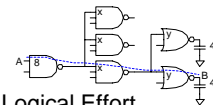


9/20/2005

VLSI Design I: A. Milenkovic

29

### Example: 3-stage path



- Logical Effort  $G =$
- Electrical Effort  $H =$
- Branching Effort  $B =$
- Path Effort  $F =$
- Best Stage Effort  $\hat{f} =$
- Parasitic Delay  $P =$
- Delay  $D =$

9/20/2005

VLSI Design I: A. Milenkovic

30

### Example: 3-stage path

Logical Effort  $G = (4/3) * (5/3) * (5/3) = 100/27$   
 Electrical Effort  $H = 45/8$   
 Branching Effort  $B = 3 * 2 = 6$   
 Path Effort  $F = GBH = 125$   
 Best Stage Effort  $f = \sqrt[3]{F} = 5$   
 Parasitic Delay  $P = 2 + 3 + 2 = 7$   
 Delay  $D = 3 * 5 + 7 = 22 = 4.4 \text{ FO}_4$

9/20/2005

VLSI Design I: A. Milenkovic

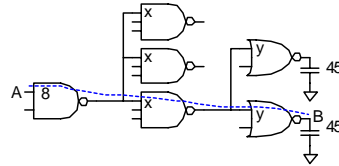
31

### Example: 3-stage path

- Work backward for sizes

$y =$

$x =$



9/20/2005

VLSI Design I: A. Milenkovic

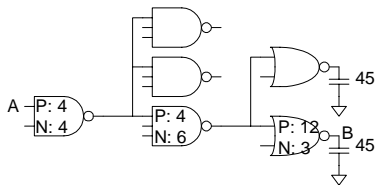
32

### Example: 3-stage path

- Work backward for sizes

$$y = 45 * (5/3) / 5 = 15$$

$$x = (15 * 2) * (5/3) / 5 = 10$$



9/20/2005

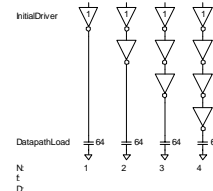
VLSI Design I: A. Milenkovic

33

### Best Number of Stages

- How many stages should a path use?
  - Minimizing number of stages is not always fastest
- Example: drive 64-bit datapath with unit inverter

$D =$



9/20/2005

VLSI Design I: A. Milenkovic

34

### Best Number of Stages

- How many stages should a path use?
  - Minimizing number of stages is not always fastest
- Example: drive 64-bit datapath with unit inverter

$$D = NF^{1/N} + P$$

$$= N(64)^{1/N} + N$$

N	t	D
1	64	65
2	8	18
3	4	15.3
4	2.8	15.3

Fastest

9/20/2005

VLSI Design I: A. Milenkovic

35

### Derivation

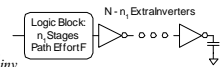
- Consider adding inverters to end of path
  - How many give least delay?

$$D = NF^{1/N} + \sum_{i=1}^{n_1} p_i + (N - n_1) p_{inv}$$

$$\frac{\partial D}{\partial N} = -F^{1/N} \ln F^{1/N} + F^{1/N} + p_{inv} = 0$$

- Define best stage effort  $\rho = F^{1/N}$

$$p_{inv} + \rho(1 - \ln \rho) = 0$$



9/20/2005

VLSI Design I: A. Milenkovic

36

### Best Stage Effort

- $p_{inv} + \rho(1 - \ln \rho) = 0$  has no closed-form solution
- Neglecting parasitics ( $p_{inv} = 0$ ), we find  $\rho = 2.718$  (e)
- For  $p_{inv} = 1$ , solve numerically for  $\rho = 3.59$

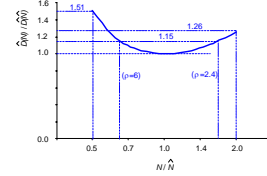
9/20/2005

VLSI Design I: A. Milenkovic

37

### Sensitivity Analysis

- How sensitive is delay to using exactly the best number of stages?



- $2.4 < \rho < 6$  gives delay within 15% of optimal
  - We can be sloppy!
  - I like  $\rho = 4$

9/20/2005

VLSI Design I: A. Milenkovic

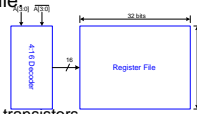
38

### Example, Revised

- Ben Bitdiddle is the memory designer for the Motorola 68W86, an embedded automotive processor. Help Ben design the decoder for a register file.

- Decoder specifications:

- 16 word register file
- Each word is 32 bits wide
- Each bit presents load of 3 unit-sized transistors
- True and complementary address inputs  $A[3:0]$
- Each input may drive 10 unit-sized transistors



- Ben needs to decide:

- How many stages to use?
- How large should each gate be?
- How fast can decoder operate?

9/20/2005

VLSI Design I: A. Milenkovic

39

### Number of Stages

- Decoder effort is mainly electrical and branching

Electrical Effort:  $H =$

Branching Effort:  $B =$

- If we neglect logical effort (assume  $G = 1$ )

Path Effort:  $F =$

Number of Stages:  $N =$

9/20/2005

VLSI Design I: A. Milenkovic

40

### Number of Stages

- Decoder effort is mainly electrical and branching

Electrical Effort:  $H = (32 \cdot 3) / 10 = 9.6$

Branching Effort:  $B = 8$

- If we neglect logical effort (assume  $G = 1$ )

Path Effort:  $F = GBH = 76.8$

Number of Stages:  $N = \log_4 F = 3.1$

- Try a 3-stage design

9/20/2005

VLSI Design I: A. Milenkovic

41

### Gate Sizes & Delay

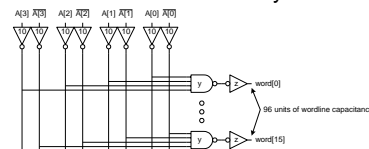
Logical Effort:  $G =$

Path Effort:  $F =$

Stage Effort:  $\hat{f} =$

Path Delay:  $D =$

Gate sizes:  $z =$   $y =$



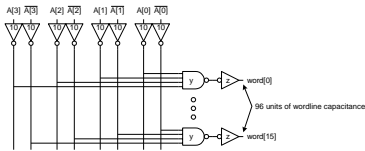
9/20/2005

VLSI Design I: A. Milenkovic

42

### Gate Sizes & Delay

Logical Effort:  $G = 1 * 6/3 * 1 = 2$   
 Path Effort:  $F = GBH = 154$   
 Stage Effort:  $\hat{f} = F^{1/3} = 5.36$   
 Path Delay:  $D = 3\hat{f} + 1 + 4 + 1 = 22.1$   
 Gate sizes:  $z = 96^{1/5.36} = 18$     $y = 18 * 2/5.36 = 6.7$



### Comparison

- Compare many alternatives with a spreadsheet

Design	N	G	P	D
NAND4-INV	2	2	5	29.8
NAND2-NOR2	2	20/9	4	30.1
INV-NAND4-INV	3	2	6	22.1
NAND4-INV-INV-INV	4	2	7	21.1
NAND2-NOR2-INV-INV	4	20/9	6	20.5
NAND2-INV-NAND2-INV	4	16/9	6	19.7
INV-NAND2-INV-NAND2-INV	5	16/9	7	20.4
NAND2-INV-NAND2-INV-INV-INV	6	16/9	8	21.6

### Review of Definitions

Term	Stage	Path
number of stages	1	$N$
logical effort	$g$	$G = \prod g_i$
electrical effort	$h = \frac{C_{in}}{C_{in,0}}$	$H = \frac{C_{out, path}}{C_{in, path}}$
branching effort	$b = \frac{C_{out, path} + C_{out, path}}{C_{in, path}}$	$B = \prod b_i$
effort	$f = gh$	$F = GBH$
effort delay	$f$	$D_F = \sum f_i$
parasitic delay	$p$	$P = \sum p_i$
delay	$d = f + p$	$D = \sum d_i = D_F + P$

### Method of Logical Effort

- 1) Compute path effort  $F = GBH$
- 2) Estimate best number of stages  $N = \log_4 F$
- 3) Sketch path with N stages
- 4) Estimate least delay  $D = NF^{1/3} + P$
- 5) Determine best stage effort  $\hat{f} = F^{1/3}$
- 6) Find gate sizes  $C_{in_i} = \frac{g_i C_{out_i}}{\hat{f}}$

### Limits of Logical Effort

- Chicken and egg problem
  - Need path to compute G
  - But don't know number of stages without G
- Simplistic delay model
  - Neglects input rise time effects
- Interconnect
  - Iteration required in designs with wire
- Maximum speed only
  - Not minimum area/power for constrained delay

### Summary

- Logical effort is useful for thinking of delay in circuits
  - Numeric logical effort characterizes gates
  - NANDs are faster than NORs in CMOS
  - Paths are fastest when effort delays are ~4
  - Path delay is weakly sensitive to stages, sizes
  - But using fewer stages doesn't mean faster paths
  - Delay of path is about  $\log_4 F$  FO4 inverter delays
  - Inverters and NAND2 best for driving large caps
- Provides language for discussing fast circuits
  - But requires practice to master