


# Introduction to CMOS VLSI Design

## Lecture 1: Circuits & Layout

David Harris



Harvey Mudd College  
Spring 2004

## Outline

- ❑ A Brief History
- ❑ CMOS Gate Design
- ❑ Pass Transistors
- ❑ CMOS Latches & Flip-Flops
- ❑ Standard Cell Layouts
- ❑ Stick Diagrams

1: Circuits & Layout      CMOS VLSI Design      Slide 2

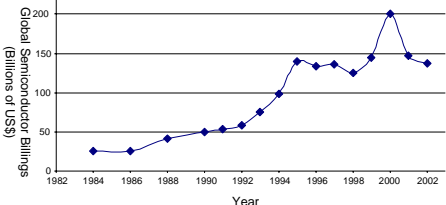
## A Brief History

- ❑ 1958: First integrated circuit
  - Flip-flop using two transistors
  - Built by Jack Kilby at Texas Instruments
- ❑ 2003
  - Intel Pentium 4  $\mu$ processor (55 million transistors)
  - 512 Mbit DRAM (> 0.5 billion transistors)
- ❑ 53% compound annual growth rate over 45 years
  - No other technology has grown so fast so long
- ❑ Driven by miniaturization of transistors
  - Smaller is cheaper, faster, lower in power!
  - Revolutionary effects on society

1: Circuits & Layout      CMOS VLSI Design      Slide 3

## Annual Sales

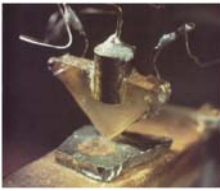
- ❑  $10^{18}$  transistors manufactured in 2003
  - 100 million for every human on the planet



1: Circuits & Layout      CMOS VLSI Design      Slide 4

## Invention of the Transistor

- ❑ Vacuum tubes ruled in first half of 20<sup>th</sup> century
  - Large, expensive, power-hungry, unreliable
- ❑ 1947: first point contact transistor
  - John Bardeen and Walter Brattain at Bell Labs
  - Read *Crystal Fire* by Riordan, Hoddeson



1: Circuits & Layout      CMOS VLSI Design      Slide 5

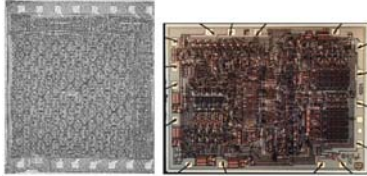
## Transistor Types

- ❑ Bipolar transistors
  - npn or pnp silicon structure
  - Small current into very thin base layer controls large currents between emitter and collector
  - Base currents limit integration density
- ❑ Metal Oxide Semiconductor Field Effect Transistors
  - nMOS and pMOS MOSFETS
  - Voltage applied to insulated gate controls current between source and drain
  - Low power allows very high integration

1: Circuits & Layout      CMOS VLSI Design      Slide 6

## MOS Integrated Circuits

- 1970's processes usually had only nMOS transistors
  - Inexpensive, but consume power while idle



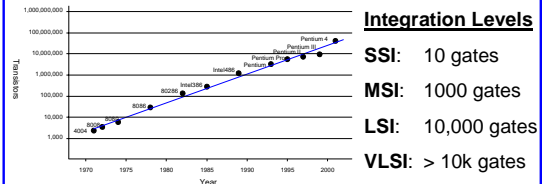
Intel 1101 256-bit SRAM    Intel 4004 4-bit  $\mu$ Proc

- 1980s-present: CMOS processes for low idle power

1: Circuits & Layout    CMOS VLSI Design    Slide 7

## Moore's Law

- 1965: Gordon Moore plotted transistor on each chip
  - Fit straight line on semilog scale
  - Transistor counts have doubled every 26 months



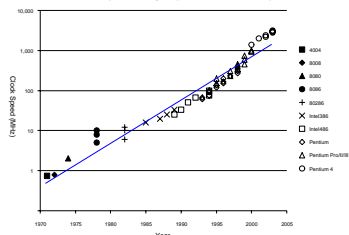
### Integration Levels

- SSI: 10 gates
- MSI: 1000 gates
- LSI: 10,000 gates
- VLSI: > 10k gates

1: Circuits & Layout    CMOS VLSI Design    Slide 8

## Corollaries

- Many other factors grow exponentially
  - Ex: clock frequency, processor performance



1: Circuits & Layout    CMOS VLSI Design    Slide 9

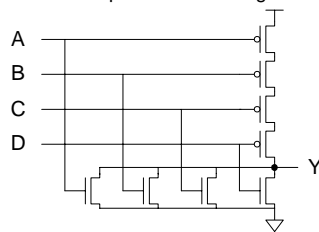
## CMOS Gate Design

- Activity:
  - Sketch a 4-input CMOS NAND gate

1: Circuits & Layout    CMOS VLSI Design    Slide 10

## CMOS Gate Design

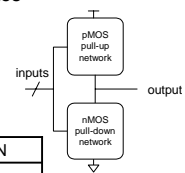
- Activity:
  - Sketch a 4-input CMOS NOR gate



1: Circuits & Layout    CMOS VLSI Design    Slide 11

## Complementary CMOS

- Complementary CMOS logic gates
  - nMOS pull-down network
  - pMOS pull-up network
  - a.k.a. static CMOS



	Pull-up OFF	Pull-up ON
Pull-down OFF	Z (float)	1
Pull-down ON	0	X (crowbar)

1: Circuits & Layout    CMOS VLSI Design    Slide 12

## Series and Parallel

- ❑ nMOS: 1 = ON
- ❑ pMOS: 0 = ON
- ❑ *Series*: both must be ON
- ❑ *Parallel*: either can be ON

1: Circuits & Layout      CMOS VLSI Design      Slide 13

## Conduction Complement

- ❑ Complementary CMOS gates always produce 0 or 1
- ❑ Ex: NAND gate
  - Series nMOS: Y=0 when both inputs are 1
  - Thus Y=1 when either input is 0
  - Requires parallel pMOS
- ❑ Rule of *Conduction Complements*
  - Pull-up network is dual of pull-down
  - Parallel -> series, series -> parallel

1: Circuits & Layout      CMOS VLSI Design      Slide 14

## Compound Gates

- ❑ *Compound gates* can do any inverting function
- ❑ Ex:  $Y = \overline{A}B + C\overline{D}$  (AND-AND-OR-INVERT, AOI22)

1: Circuits & Layout      CMOS VLSI Design      Slide 15

## Example: O3AI

- ❑  $Y = \overline{(A+B+C)D}$

1: Circuits & Layout      CMOS VLSI Design      Slide 16

## Example: O3AI

- ❑  $Y = \overline{(A+B+C)D}$

1: Circuits & Layout      CMOS VLSI Design      Slide 17



## Signal Strength

- ❑ *Strength* of signal
  - How close it approximates ideal voltage source
- ❑  $V_{DD}$  and GND rails are strongest 1 and 0
- ❑ nMOS pass strong 0
  - But degraded or weak 1
- ❑ pMOS pass strong 1
  - But degraded or weak 0
- ❑ Thus nMOS are best for pull-down network

1: Circuits & Layout      CMOS VLSI Design      Slide 18

## Pass Transistors



- Transistors can be used as switches


1: Circuits & Layout
CMOS VLSI Design
Slide 19

## Pass Transistors

- Transistors can be used as switches


$g = 0$



Input  $g = 1$  Output

$0 \rightarrow$  strong 0


$g = 1$



$g = 1$

$1 \rightarrow$  degraded 1


$g = 0$



Input  $g = 0$  Output

$0 \rightarrow$  degraded 0

$g = 1$



$g = 0$

$\rightarrow$  strong 1

1: Circuits & Layout
CMOS VLSI Design
Slide 20





## Transmission Gates

- Pass transistors produce degraded outputs
- Transmission gates* pass both 0 and 1 well


1: Circuits & Layout
CMOS VLSI Design
Slide 21

## Transmission Gates

- Pass transistors produce degraded outputs
- Transmission gates* pass both 0 and 1 well


$g = 0, gb = 1$



Input  $g = 1, gb = 0$  Output

$0 \rightarrow$  strong 0

$g = 1, gb = 0$



$g = 1, gb = 0$

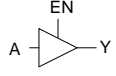
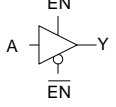
$1 \rightarrow$  strong 1

1: Circuits & Layout
CMOS VLSI Design
Slide 22

## Tristates

- Tristate buffer* produces Z when not enabled

EN	A	Y
0	0	
0	1	
1	0	
1	1	

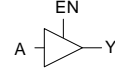
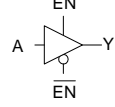



1: Circuits & Layout
CMOS VLSI Design
Slide 23

## Tristates

- Tristate buffer* produces Z when not enabled

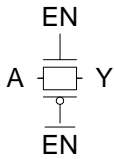
EN	A	Y
0	0	Z
0	1	Z
1	0	0
1	1	1

1: Circuits & Layout
CMOS VLSI Design
Slide 24

## Nonrestoring Tristate

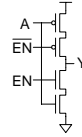
- Transmission gate acts as tristate buffer
  - Only two transistors
  - But *nonrestoring*
    - Noise on A is passed on to Y



1: Circuits & Layout CMOS VLSI Design Slide 25

## Tristate Inverter

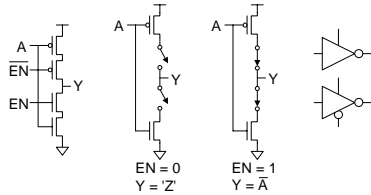
- Tristate inverter produces restored output
  - Violates conduction complement rule
  - Because we want a Z output



1: Circuits & Layout CMOS VLSI Design Slide 26

## Tristate Inverter

- Tristate inverter produces restored output
  - Violates conduction complement rule
  - Because we want a Z output

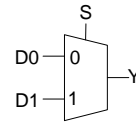


1: Circuits & Layout CMOS VLSI Design Slide 27

## Multiplexers

- 2:1 multiplexer chooses between two inputs

S	D1	D0	Y
0	X	0	
0	X	1	
1	0	X	
1	1	X	

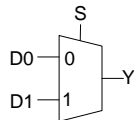


1: Circuits & Layout CMOS VLSI Design Slide 28

## Multiplexers

- 2:1 multiplexer chooses between two inputs

S	D1	D0	Y
0	X	0	0
0	X	1	1
1	0	X	0
1	1	X	1



1: Circuits & Layout CMOS VLSI Design Slide 29

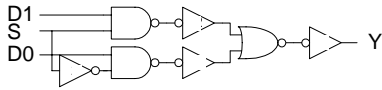
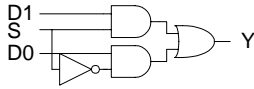
## Gate-Level Mux Design

- $Y = SD_1 + \bar{S}D_0$  (too many transistors)
- How many transistors are needed?

1: Circuits & Layout CMOS VLSI Design Slide 30

## Gate-Level Mux Design

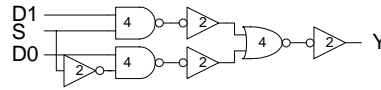
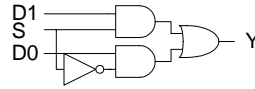
- ❑  $Y = SD_1 + \bar{S}D_0$  (too many transistors)
- ❑ How many transistors are needed?



1: Circuits & Layout CMOS VLSI Design Slide 31

## Gate-Level Mux Design

- ❑  $Y = SD_1 + \bar{S}D_0$  (too many transistors)
- ❑ How many transistors are needed? 20



1: Circuits & Layout CMOS VLSI Design Slide 32

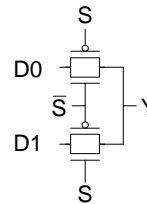
## Transmission Gate Mux

- ❑ Nonrestoring mux uses two transmission gates

1: Circuits & Layout CMOS VLSI Design Slide 33

## Transmission Gate Mux

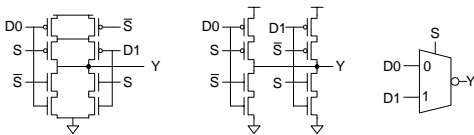
- ❑ Nonrestoring mux uses two transmission gates
  - Only 4 transistors



1: Circuits & Layout CMOS VLSI Design Slide 34

## Inverting Mux

- ❑ Inverting multiplexer
  - Use compound AOI22
  - Or pair of tristate inverters
  - Essentially the same thing
- ❑ Noninverting multiplexer adds an inverter



1: Circuits & Layout CMOS VLSI Design Slide 35

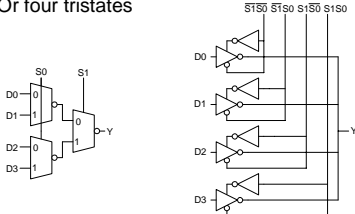
## 4:1 Multiplexer

- ❑ 4:1 mux chooses one of 4 inputs using two selects

1: Circuits & Layout CMOS VLSI Design Slide 36

## 4:1 Multiplexer

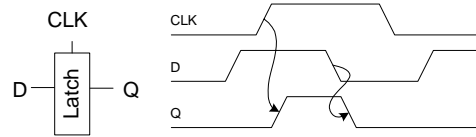
- 4:1 mux chooses one of 4 inputs using two selects
  - Two levels of 2:1 muxes
  - Or four tristates



1: Circuits & Layout CMOS VLSI Design Slide 37

## D Latch

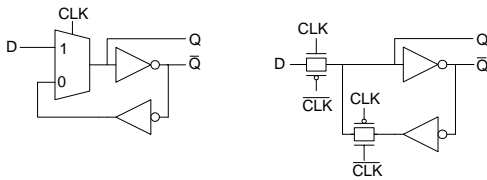
- When CLK = 1, latch is *transparent*
  - D flows through to Q like a buffer
- When CLK = 0, the latch is *opaque*
  - Q holds its old value independent of D
- a.k.a. *transparent latch* or *level-sensitive latch*



1: Circuits & Layout CMOS VLSI Design Slide 38

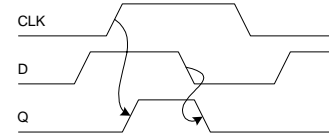
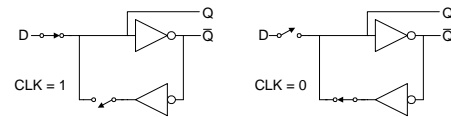
## D Latch Design

- Multiplexer chooses D or old Q



1: Circuits & Layout CMOS VLSI Design Slide 39

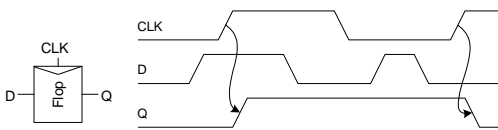
## D Latch Operation



1: Circuits & Layout CMOS VLSI Design Slide 40

## D Flip-flop

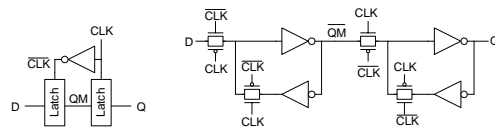
- When CLK rises, D is copied to Q
- At all other times, Q holds its value
- a.k.a. *positive edge-triggered flip-flop*, *master-slave flip-flop*



1: Circuits & Layout CMOS VLSI Design Slide 41

## D Flip-flop Design

- Built from master and slave D latches



1: Circuits & Layout CMOS VLSI Design Slide 42

### D Flip-flop Operation

1: Circuits & Layout      CMOS VLSI Design      Slide 43

### Race Condition

- ❑ Back-to-back flops can malfunction from clock skew
  - Second flip-flop fires late
  - Sees first flip-flop change and captures its result
  - Called *hold-time failure* or *race condition*

1: Circuits & Layout      CMOS VLSI Design      Slide 44

### Nonoverlapping Clocks

- ❑ Nonoverlapping clocks can prevent races
  - As long as nonoverlap exceeds clock skew
- ❑ You can use them if you like for safe design
  - Industry manages skew more carefully instead

1: Circuits & Layout      CMOS VLSI Design      Slide 45