inst.eecs.berkeley.edu/~cs61c

UC Berkeley CS61C : Machine Structures

Lecture 20 – Synchronous Digital Systems

2007-03-05

Lecturer SOE Dan Garcia

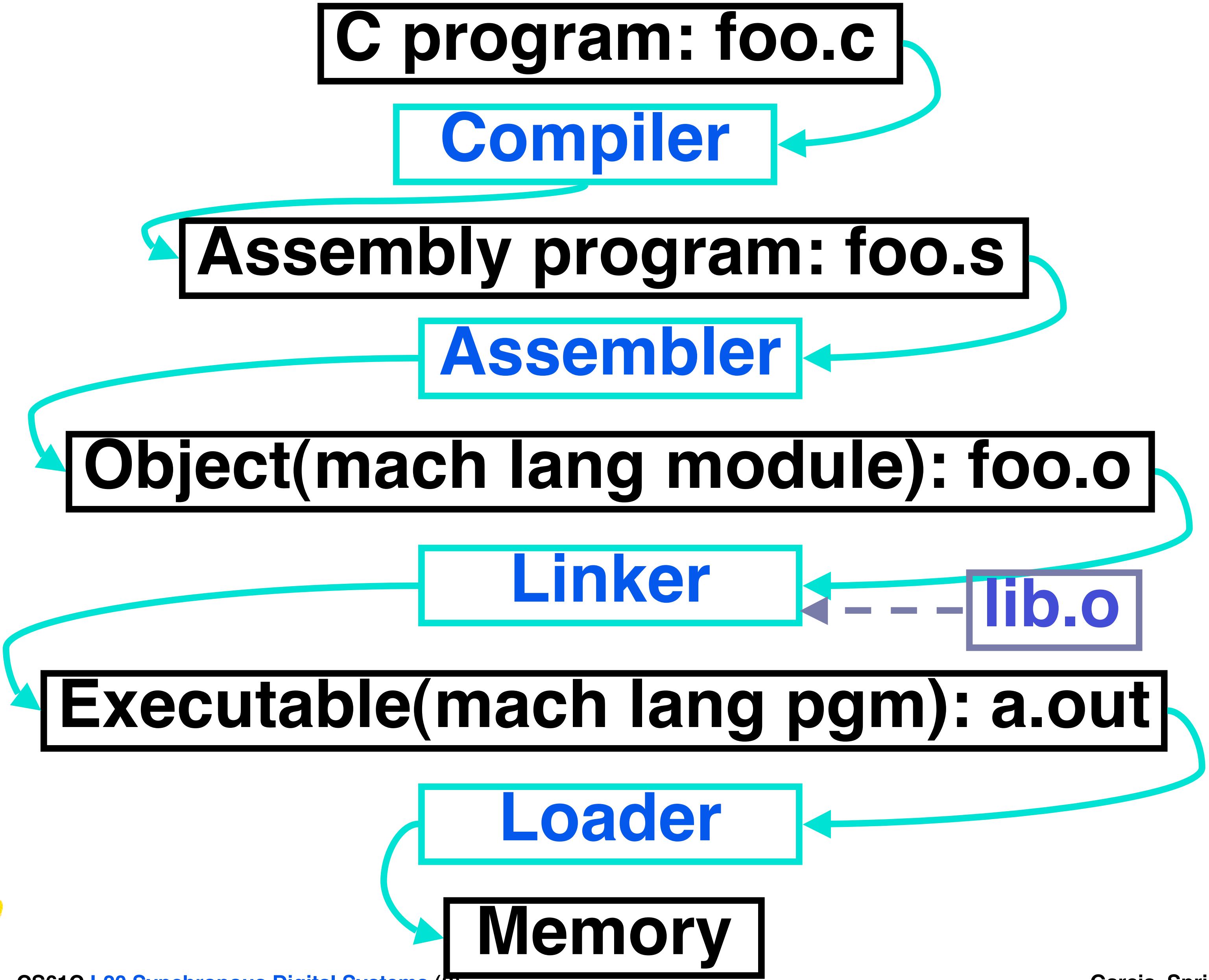
www.cs.berkeley.edu/~ddgarcia

Disk failures 15x specs! ⇒

A recent conference reveals that drives fail in real life MUCH more than data sheets claim, temp has little effect on reliability, and that costly Fibre Channel drives were no more reliable than SATA drives. Fast temp Δ bad.

www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9012066

Review



C program: foo.c

Compiler

Assembly program: foo.s

Assembler

Object(mach lang module): foo.o

Linker

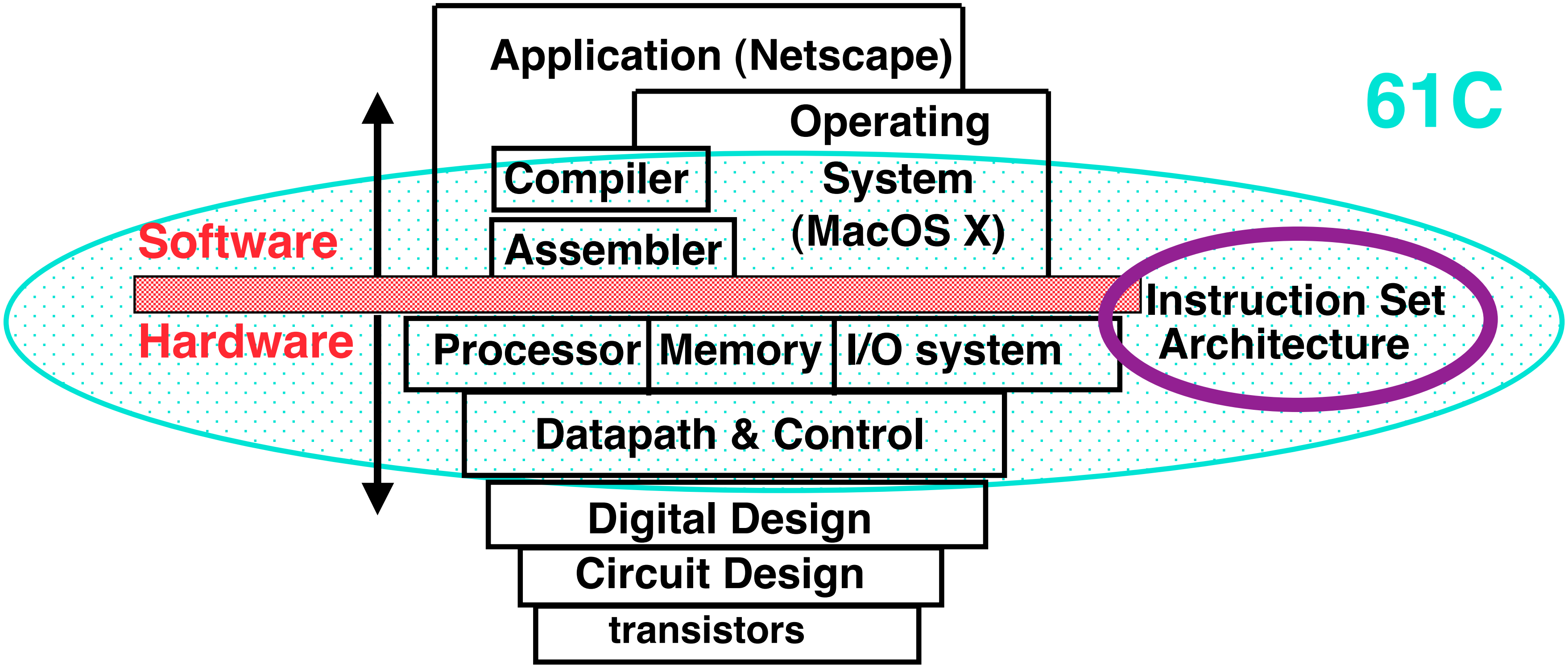
lib.o

Executable(mach lang pgm): a.out

Loader

Memory

What are "Machine Structures"?



Coordination of many *levels of abstraction*

ISA is an important abstraction level: contract between HW & SW

Below the Program

- **High-level language program (in C)**

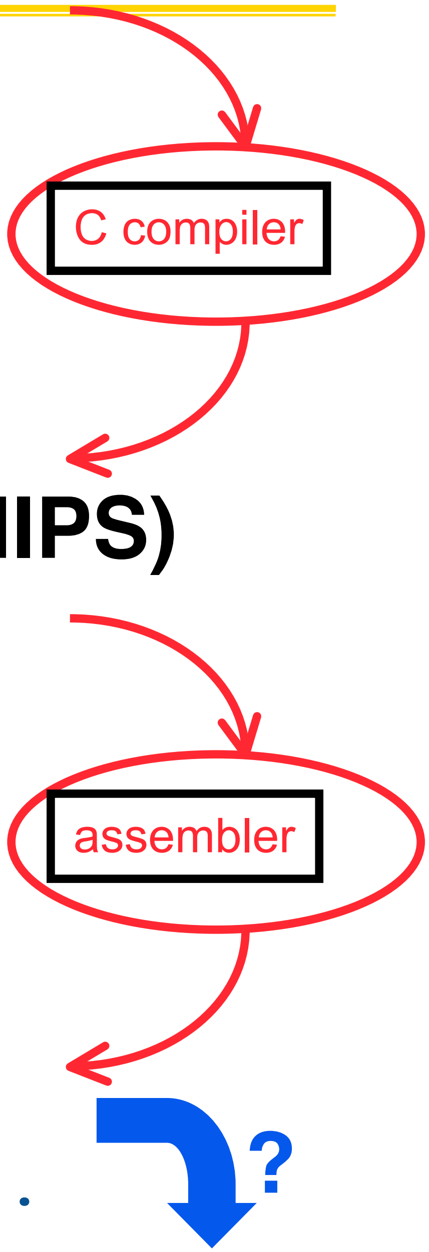
```
swap   int v[], int k){
       int temp;
       temp = v[k];
       v[k] = v[k+1];
       v[k+1] = temp;
}
```

- **Assembly language program (for MIPS)**

```
swap: sll   $2, $5, 2
      add   $2, $4,$2
      lw    $15, 0($2)
      lw    $16, 4($2)
      sw    $16, 0($2)
      sw    $15, 4($2)
      jr    $31
```

- **Machine (object) code (for MIPS)**

```
000000 00000 00101 0001000010000000
000000 00100 00010 0001000000100000 . . .
```



C compiler

assembler

?

Synchronous Digital Systems

The hardware of a processor, such as the MIPS, is an example of a Synchronous Digital System

Synchronous:

- **Means all operations are coordinated by a central clock.**
  - **It keeps the "heartbeat" of the system!**

Digital:

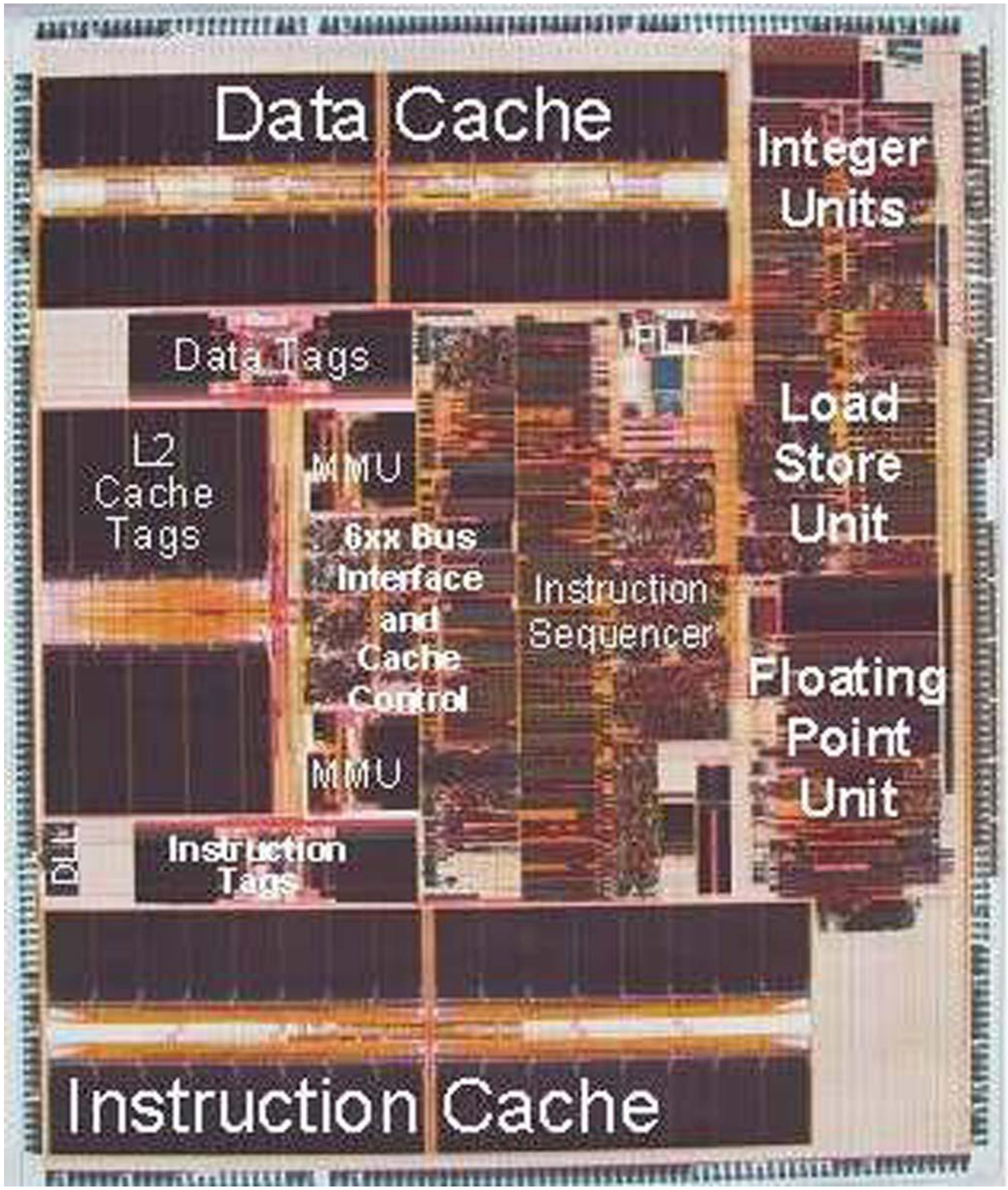
- **Mean all values are represented by discrete values**
- **Electrical signals are treated as 1's and 0's and grouped together to form words.**

Logic Design

- Next 4 weeks: we'll study how a modern processor is built; starting with basic elements as building blocks.
- Why study hardware design?
  - Understand capabilities and limitations of hardware in general and processors in particular.
  - What processors can do fast and what they can't do fast (avoid slow things if you want your code to run fast!)
  - Background for more detailed hardware courses (CS 150, CS 152)
  - There is just so much you can do with processors. At some point you may need to design your own custom hardware.

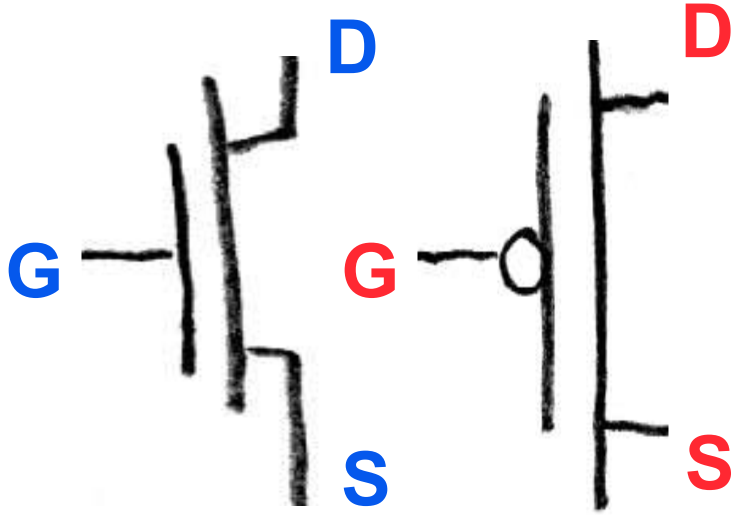
PowerPC Die Photograph

Let's look closer…

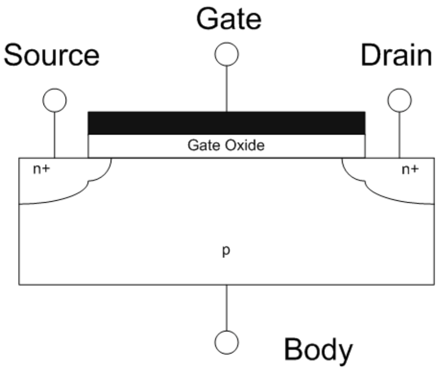


Transistors 101

- MOSFET
  - Metal-Oxide-Semiconductor Field-Effect Transistor
  - Come in two types:
    - n-type NMOSFET
    - p-type PMOSFET
- For n-type (p-type opposite)
  - If voltage not enough between G & S, transistor turns “off” (cut-off) and Drain-Source NOT connected
  - If the G & S voltage is high enough, transistor turns “on” (saturation) and Drain-Source ARE connected



n-type p-type

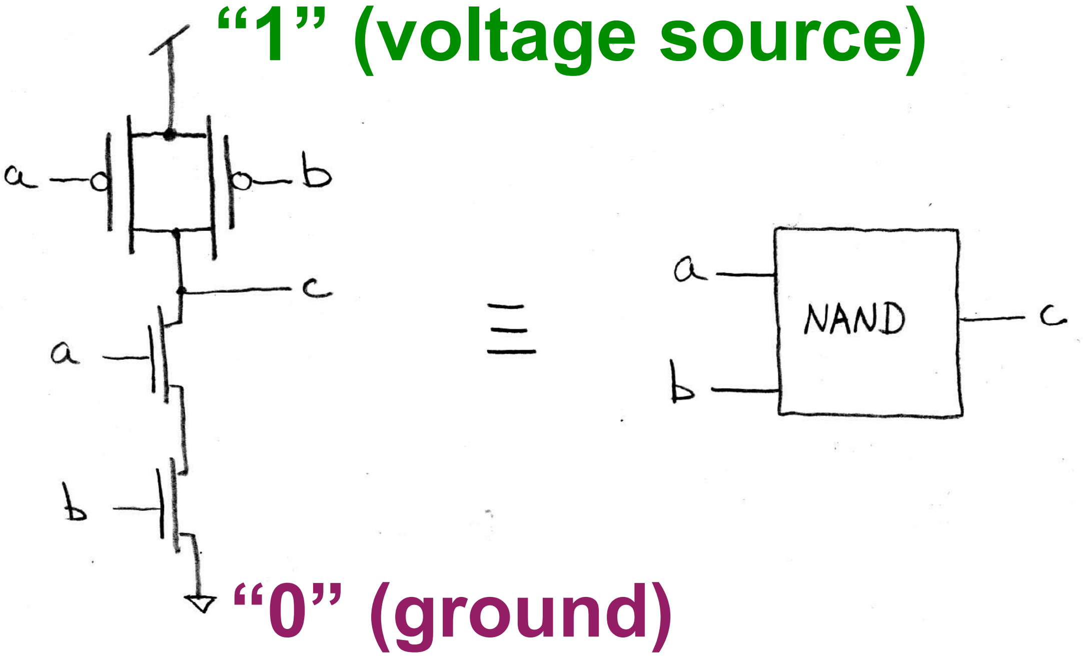


Side view

www.wikipedia.org/wiki/Mosfet

Transistor Circuit Rep. vs. Block diagram

- Chips is composed of nothing but transistors and wires.
- Small groups of transistors form useful building blocks.



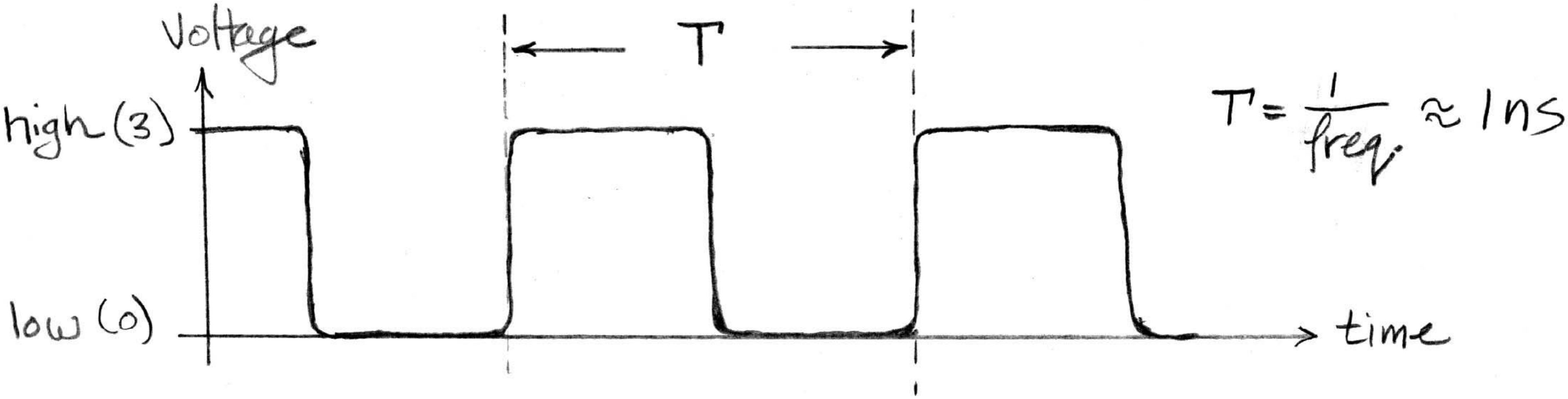
“1” (voltage source)

“0” (ground)

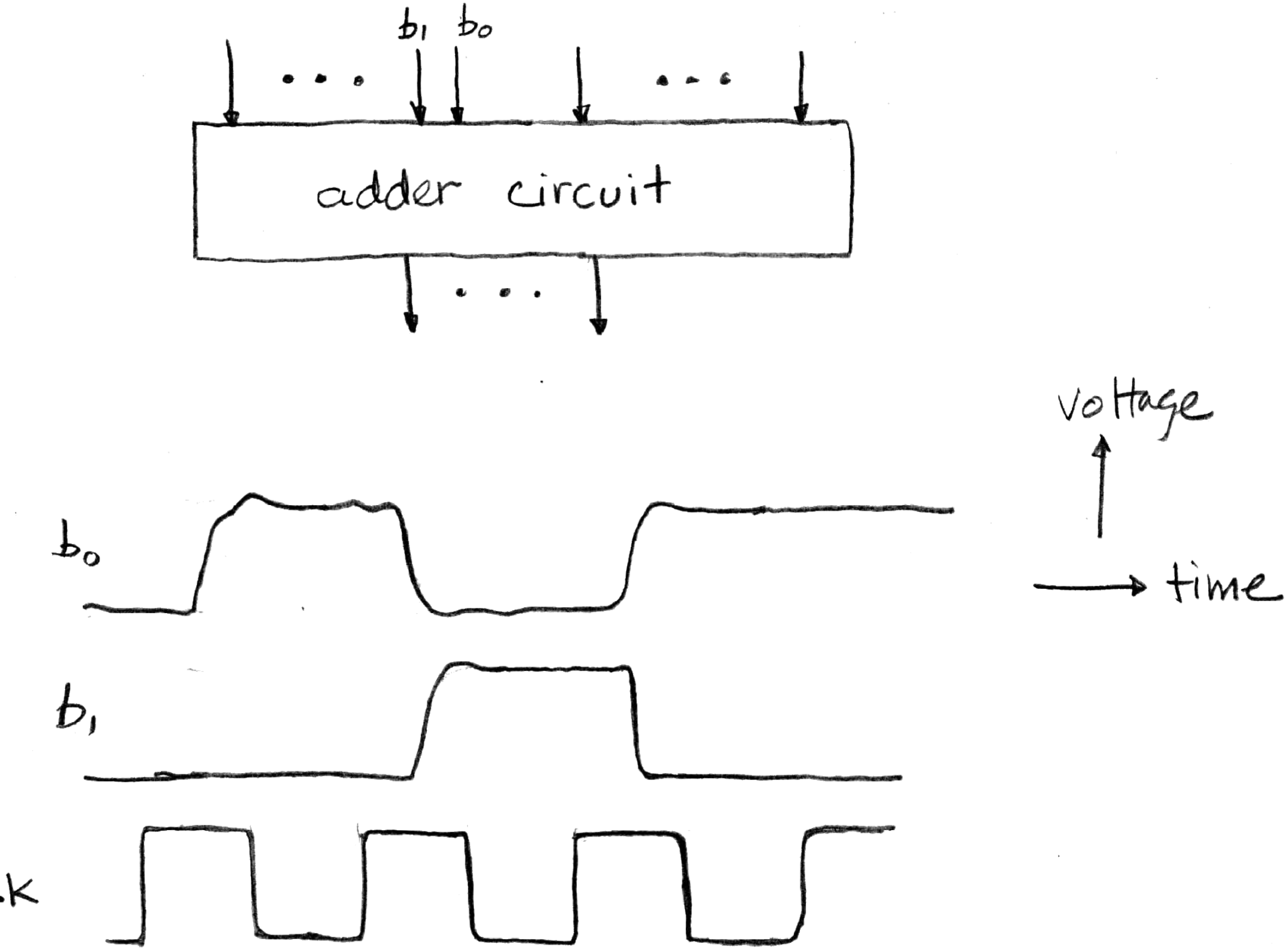
a	b	c
0	0	1
0	1	1
1	0	1
1	1	0

- Block are organized in a hierarchy to build higher-level blocks: ex: adders.

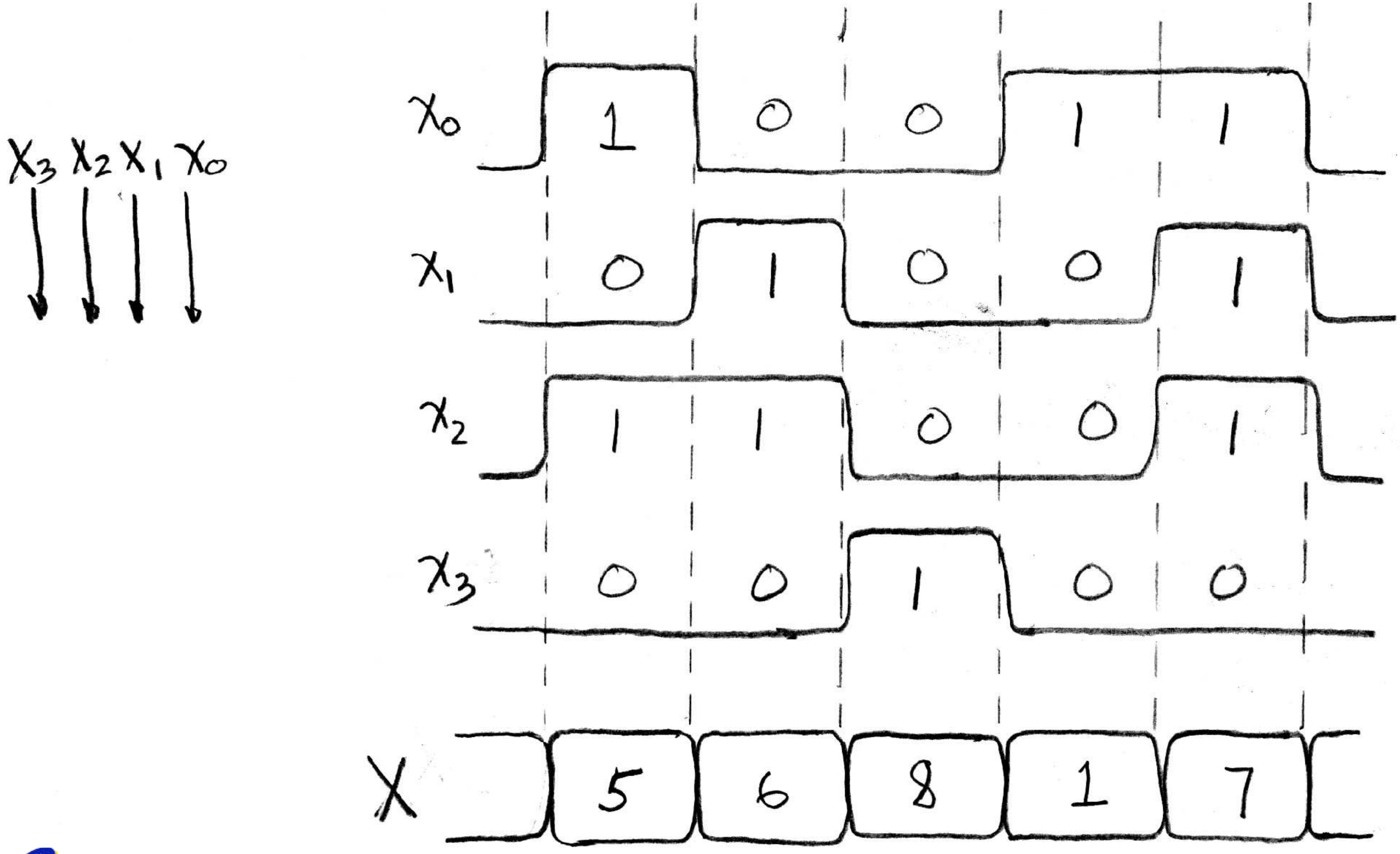
The Clock Signal



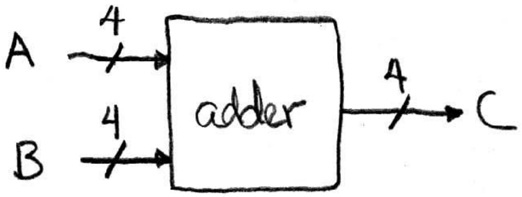
Signals and Waveforms



Signals and Waveforms: Grouping

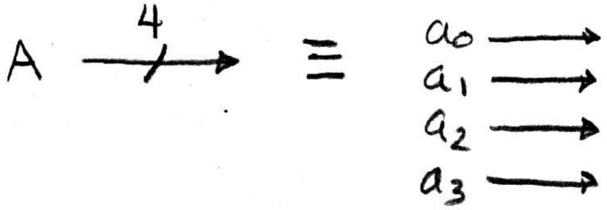


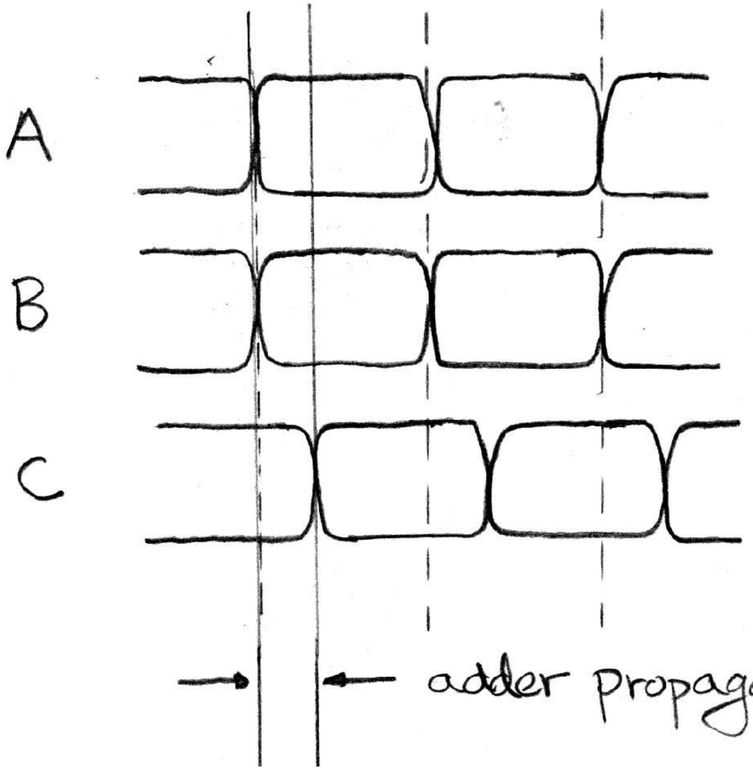
Signals and Waveforms: Circuit Delay



$A = [a_3, a_2, a_1, a_0]$

$B = [b_3, b_2, b_1, b_0]$



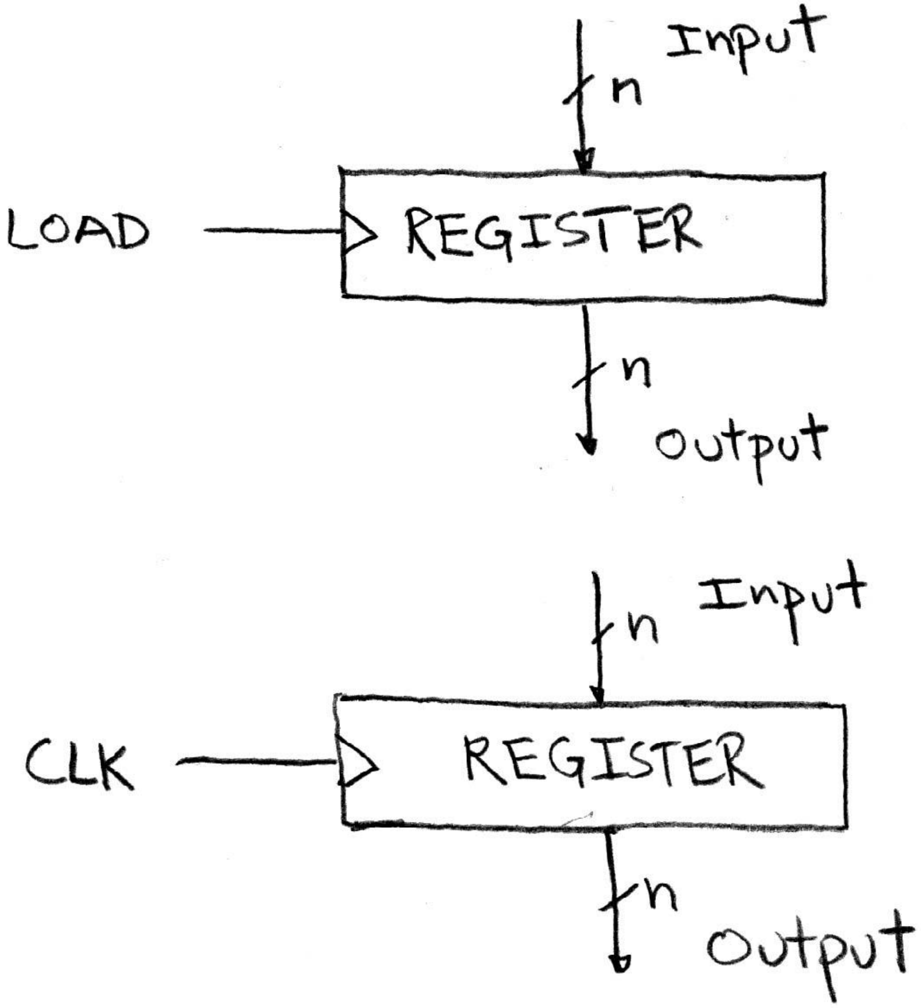


adder propagation delay

Type of Circuits

- Synchronous Digital Systems are made up of two basic types of circuits:
- Combinational Logic (CL) circuits
  - Our previous adder circuit is an example.
  - Output is a function of the inputs only.
  - Similar to a pure function in mathematics, y = f(x). (No way to store information from one invocation to the next. No side effects)
- State Elements: circuits that store information.

Circuits with STATE (e.g., register)



Peer Instruction

A. SW can peek at HW (past ISA abstraction boundary) for optimizations

B. SW can depend on particular HW implementation of ISA

C. Timing diagrams serve as a critical debugging tool in the EE toolkit

	ABC
0:	FFF
1:	FFT
2:	FTF
3:	FTT
4:	TFF
5:	TFT
6:	TTF
7:	TTT

And in conclusion…

- ISA is very important abstraction layer
  - Contract between HW and SW
- Clocks control pulse of our circuits
- Voltages are analog, quantized to 0/1
- Circuit delays are fact of life
- Two types of circuits:
  - Stateless Combinational Logic (&,l,~)
  - State circuits (e.g., registers)