CS 61C:
Great Ideas in Computer Architecture
Datapath Building Blocks

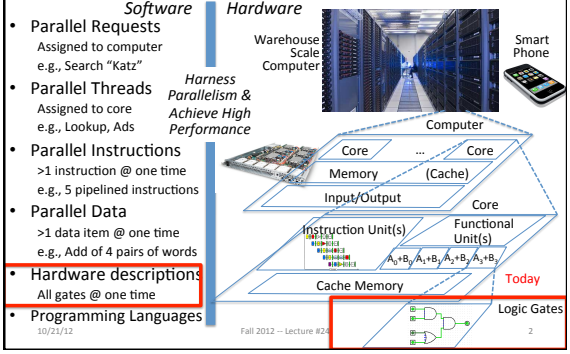
Instructors:
Krste Asanovic, Randy H. Katz
http://inst.eecs.Berkeley.edu/~cs61c/fa12

You are Here!

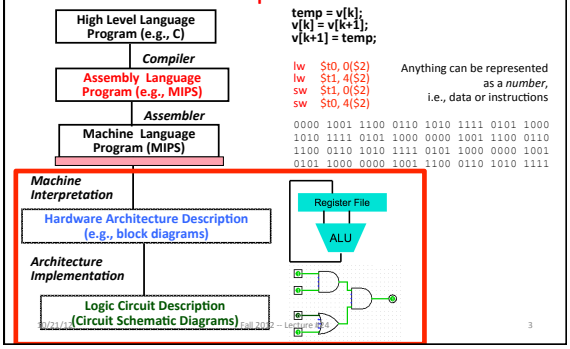
Software | *Hardware*

- Parallel Requests
  Assigned to computer
  e.g., Search "Katz"
- Parallel Threads
  Assigned to core
  e.g., Lookup, Ads
- Parallel Instructions
  >1 instruction @ one time
  e.g., 5 pipelined instructions
- Parallel Data
  >1 data item @ one time
  e.g., Add of 4 pairs of words
- Hardware descriptions
  All gates @ one time
- Programming Languages

Harness Parallelism & Achieve High Performance



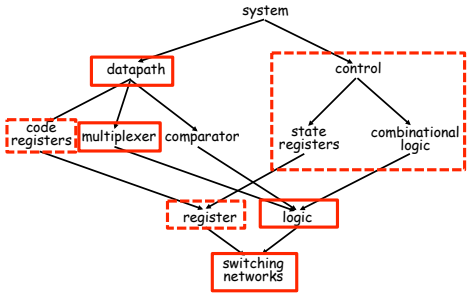
Levels of Representation/ Interpretation



Review

- Truth table can be mapped to gates for combinational logic design
- Boolean algebra allows minimization of gates
- Sequential vs. Combinational Logic
- Unique configurations of a digital system: *State* and *State Machines*

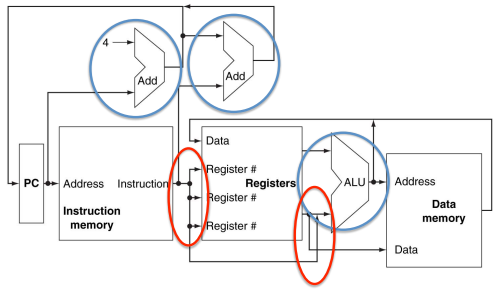
Design Hierarchy



Type of Circuits

- *Synchronous Digital Systems* consist of two basic types of circuits:
  - Combinational Logic (CL) circuits
    - Output is a function of the inputs only, not the history of its execution
    - *E.g., circuits to add A, B (ALUs)*
  - Sequential Logic (SL)
    - Circuits that "remember" or store information
    - aka "State Elements"
    - E.g., memories and registers (Registers)

Conceptual MIPS Datapath



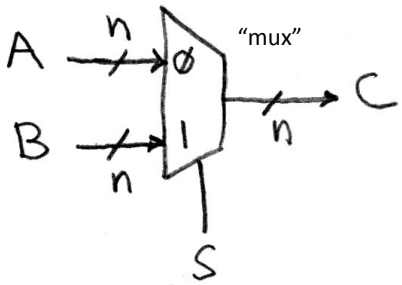
Agenda

- Multiplexer
- ALU Design
- And in Conclusion, …

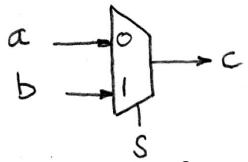
Agenda

- Multiplexer
- ALU Design
- And in Conclusion, …

Data Multiplexer (e.g., 2-to-1 x n-bit-wide)



N Instances of 1-bit-Wide Mux

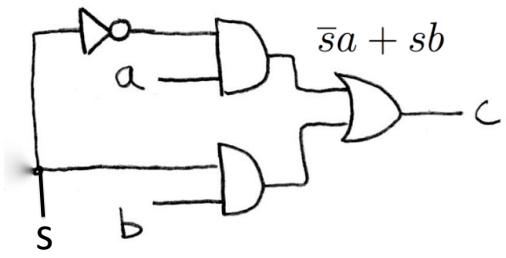


How many rows in TT?

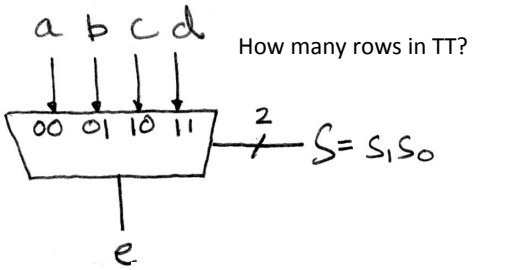
$$
\begin{aligned}
c &= \bar{s}a\bar{b} + \bar{s}ab + s\bar{a}b + sab \\
&= \bar{s}(a\bar{b} + ab) + s(\bar{a}b + ab) \\
&= \bar{s}(a(\bar{b} + b)) + s((\bar{a} + a)b) \\
&= \bar{s}(a(1)) + s((1)b) \\
&= \bar{s}a + sb
\end{aligned}
$$

s	ab	c
0	00	0
0	01	0
0	10	1
1	00	0
1	01	1
1	10	0
1	11	1

How Do We Build a 1-bit-Wide Mux (in Logisim)?

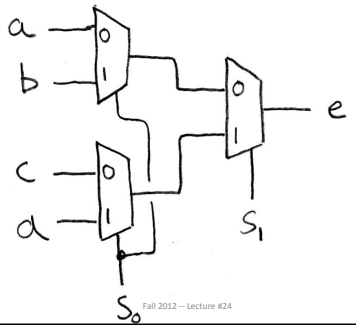


4-to-1 Multiplexer

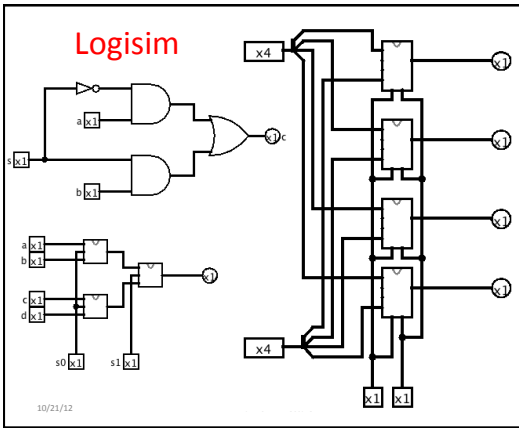


$$e = \overline{s_1}\,\overline{s_0}a + \overline{s_1}s_0b + s_1\overline{s_0}c + s_1s_0d$$

Alternative Hierarchical Approach (in Logisim)



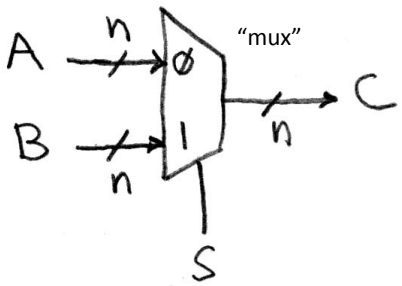
Logisim



Subcircuits

- Subcircuit: Logisim equivalent of procedure or method
  - Every project is a hierarchy of subcircuits

N-bit-wide Data Multiplexer (in Logisim + tunnel)



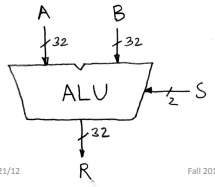
Administrivia

Agenda

- Multiplexer
- ALU Design
- And, in Conclusion, …

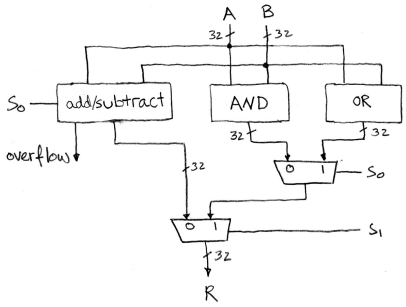
Arithmetic and Logic Unit

- Most processors contain a special logic block called "Arithmetic and Logic Unit" (ALU)
- We'll show you an easy one that does ADD, SUB, bitwise AND, bitwise OR



when S=00, R=A+B
when S=01, R=A-B
when S=10, R=A AND B
when S=11, R=A OR B

Simple ALU



Adder/Subtractor: One-bit adder Least Significant Bit

	a_3	a_2	a_1	a_0
+	b_3	b_2	b_1	b_0
	s_3	s_2	s_1	s_0

a_0	b_0	s_0	c_1
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$s_0 = a_0 \text{ XOR } b_0$$
$$c_1 = a_0 \text{ AND } b_0$$

Adder/Subtractor: One-bit adder (1/2)

…

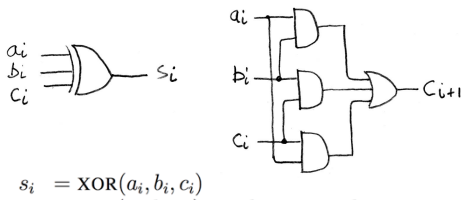
	a_3	a_2	a_1	a_0
+	b_3	b_2	b_1	b_0
	s_3	s_2	s_1	s_0

a_i	b_i	c_i	s_i	c_{i+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

s_i =
c_{i+1} =

Adder/Subtractor: One-bit Adder (2/2)

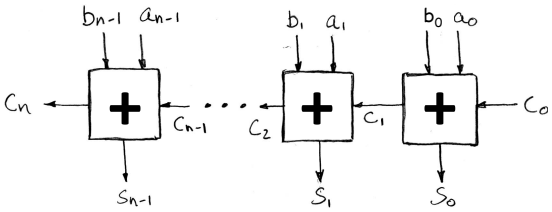
…



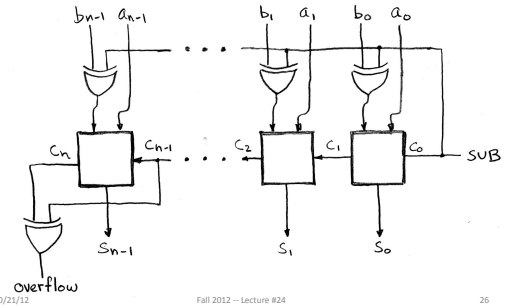
$s_i = \text{XOR}(a_i, b_i, c_i)$
$c_{i+1} = \text{MAJ}(a_i, b_i, c_i) = a_i b_i + a_i c_i + b_i c_i$

N x 1-bit Adders ⇒ 1 N-bit Adder

Connect Carry Out i-1 to Carry in i:



Twos Complement Adder/Subtractor



Critical Path

- When setting clock period in synchronous systems, must allow for worst case
- Path through combinational logic that is worst case called "critical path"
  - Can be estimated by number of "gate delays": Number of gates must go through in worst case
- Idea: Doesn't matter if speedup other paths if don't improve the critical path
- What might critical path of ALU?

And, in Conclusion, …

- Use muxes to select among input
  - S input bits selects 2^S inputs
  - Each input can be n-bits wide, indep of S
- Can implement muxes hierarchically
- Arithmetic circuits are a kind of combinational logic