

CSCI 210: Computer Architecture

Lecture 18: Clocks and Timing

Stephen Checkoway

Slides from Cynthia Taylor

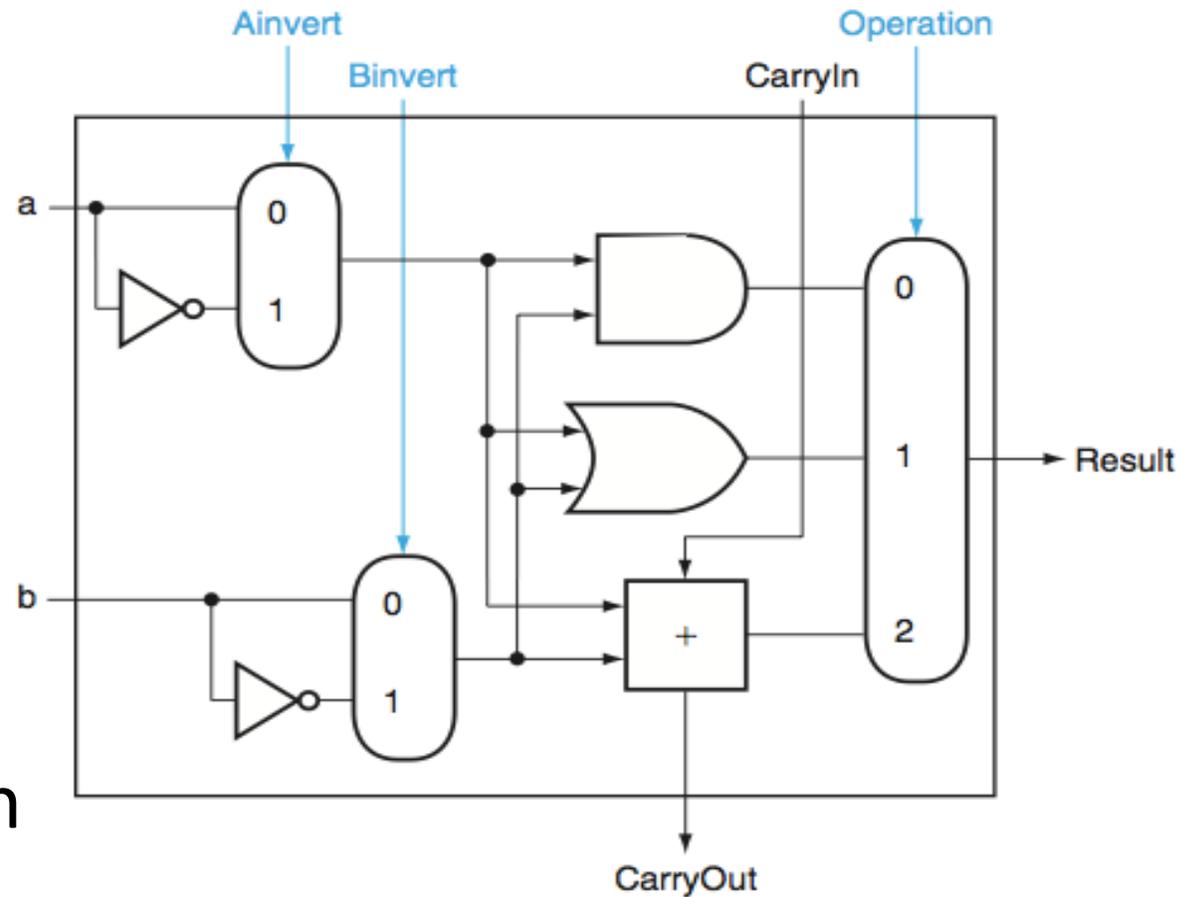
CS History: Lynn Conway



- Computer Scientist, Electrical Engineer and Transgender Activist born in 1938
- At Xerox Park, worked on *multi-project wafers*, which allowed printing multiple circuit designs on a single silicon wafer
- Co-authored *Introduction to VLSI Systems* in 1981, which became the standard textbook in chip design
- In 1983, developed the *Metal Oxide Semiconductor Implementation Service* system, an internet-based system for rapid-prototyping and small-run fabrication of new chip designs
 - Allowed academics and small businesses to develop their own chips for the first time

Last Class We Implemented A 1-bit CPU with

- And
- Or
- Nor
- Add
- Subtract
- Set less than
- Overflow detection



Adding Conditional Branching

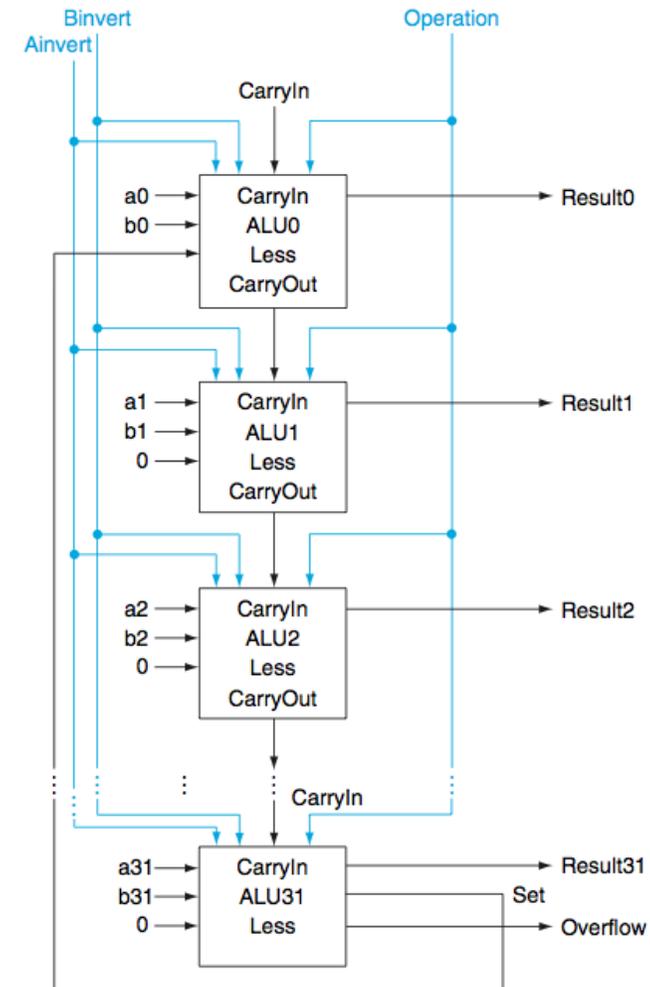
- Want to be able to support beq, bne, etc
- Need to be able to check equality
- If $a = b$, then $a - b = 0$

Detect 0 in Multi-bit ALU

- Subtract $a - b$
- Take output from each 1-bit ALU
 - If they are equal, all outputs should be 0

We know Result0-Result31 are all 0 if we perform a _____ operation on Result0 through Result31, and it outputs _____

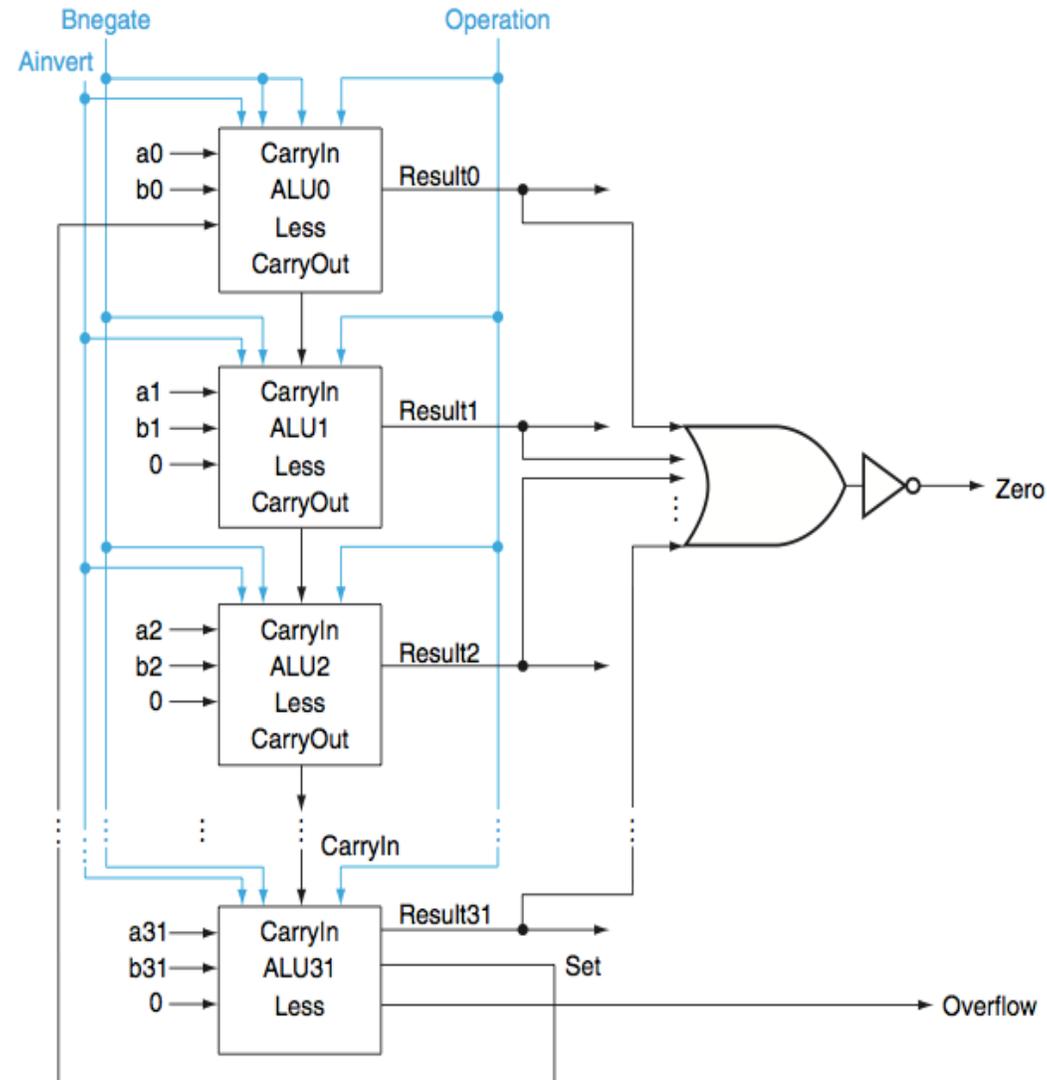
- A. AND, 0
- B. OR, 0
- C. NAND, 1
- D. XOR, 0
- E. None of the above



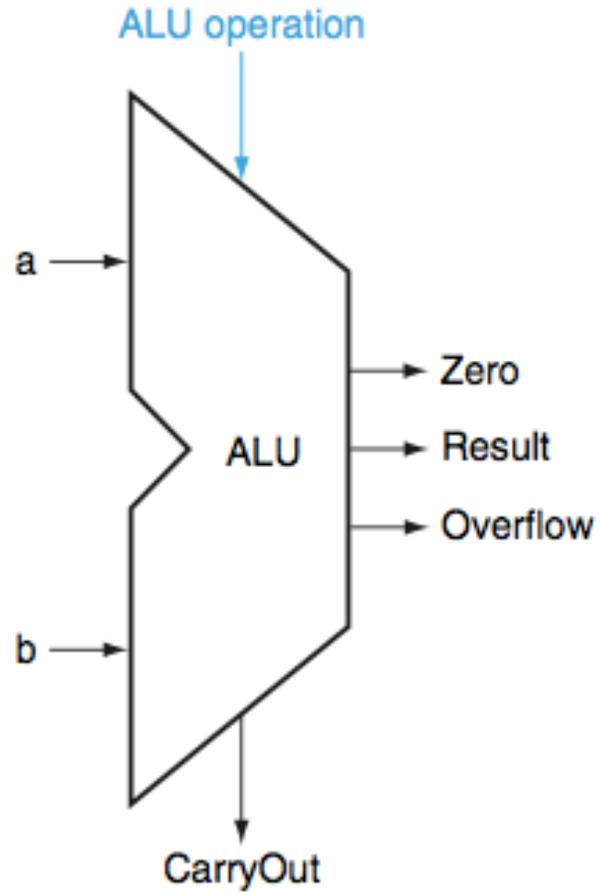
Detect 0 in Multi-bit ALU

- Subtract $a - b$
- Take output from each 1-bit ALU
- OR outputs together
 - If any output is 1, result will be 1, else 0
- Negate the result

32-bit ALU with zero check

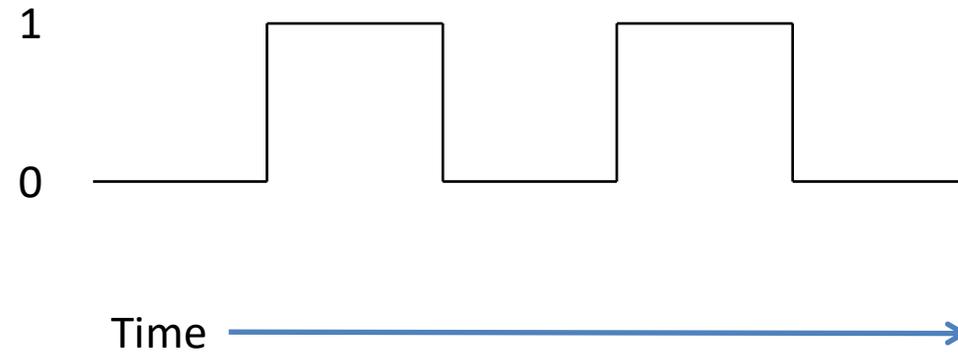


Symbol for Multi-bit ALU

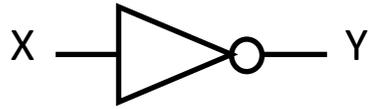


ALU Questions?

Logic Gates and Timing Diagrams



Which of the following most closely maps to Y (the output of the inverter)?

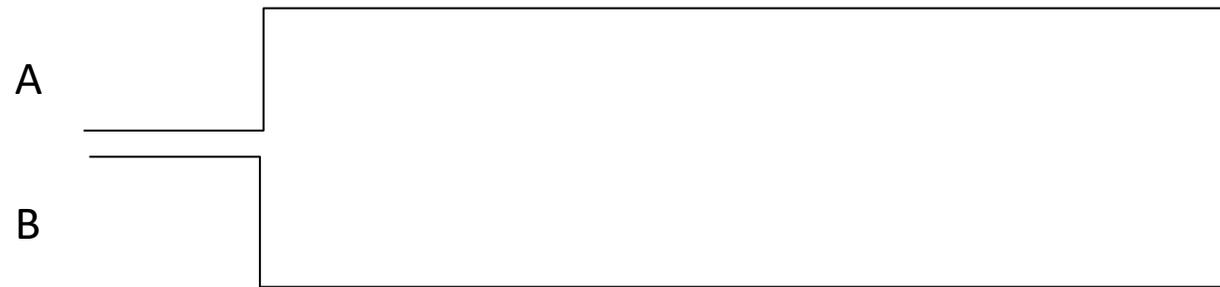


Inverter

X Y

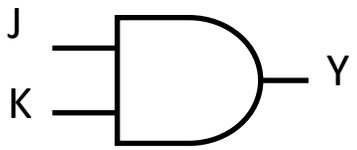
0 1

1 0



E None of the above.

Select the correct output for Y



AND

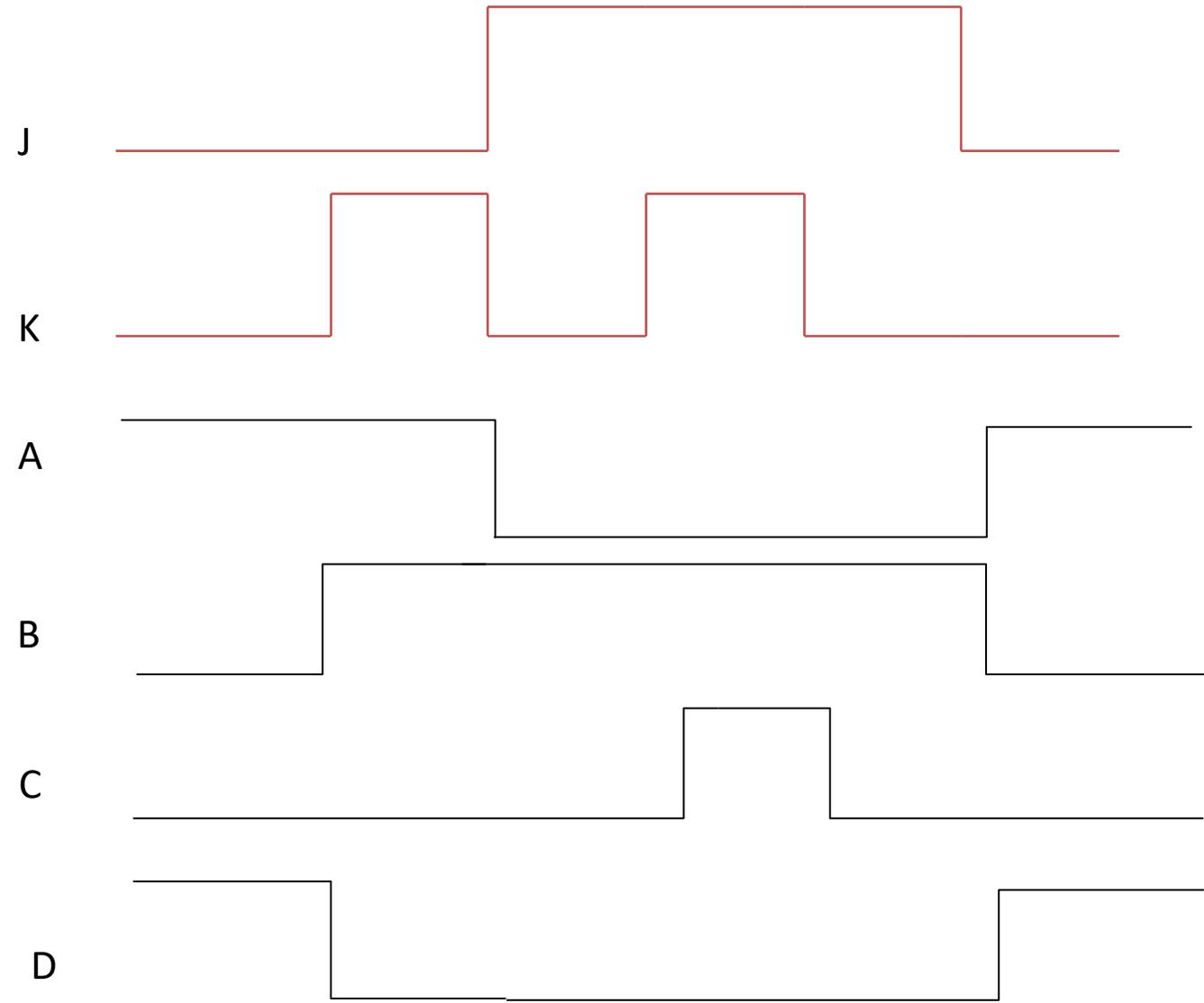
JK Y

00 0

01 0

10 0

11 1



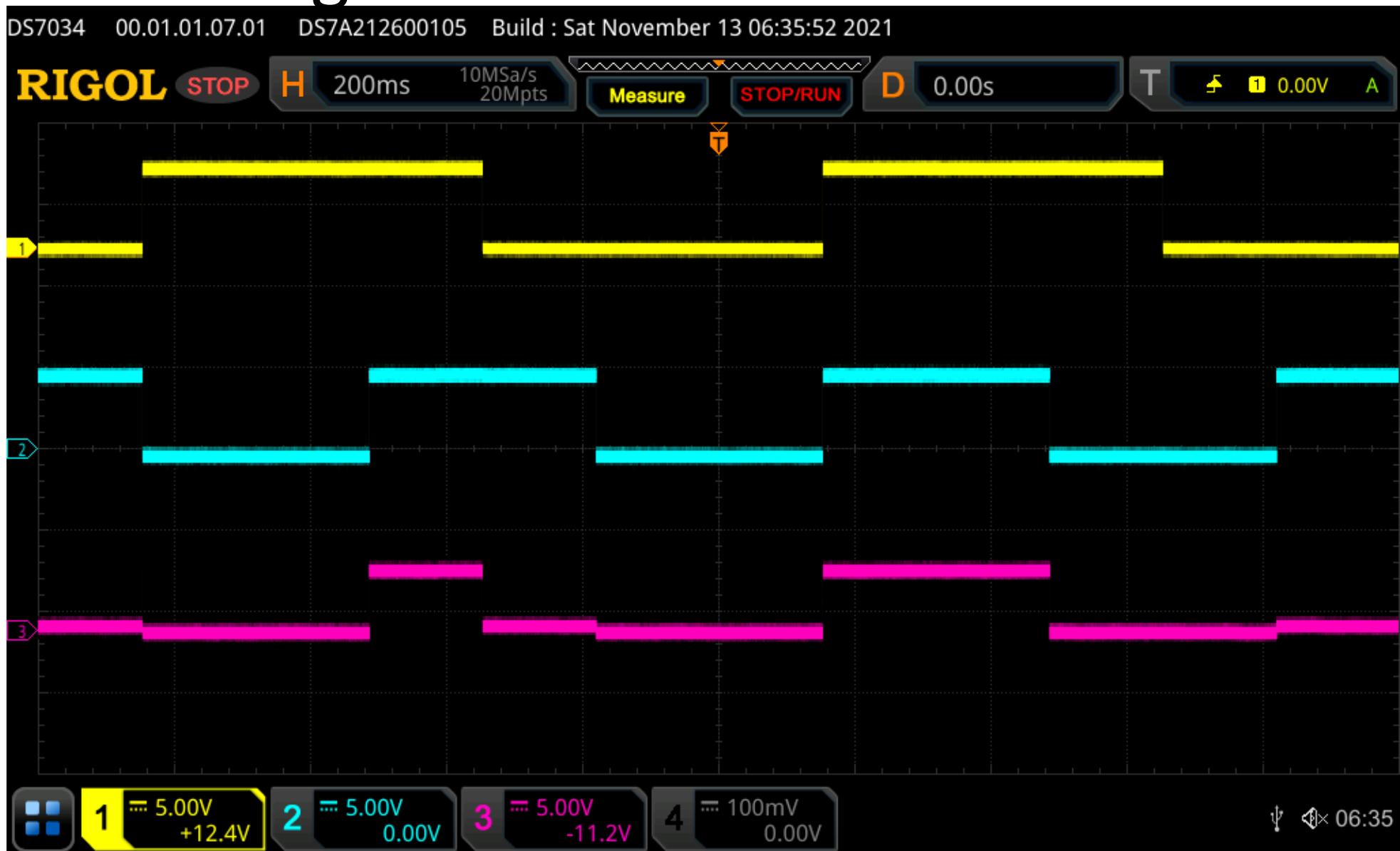
E None of the above

Timing diagrams and voltage measurements

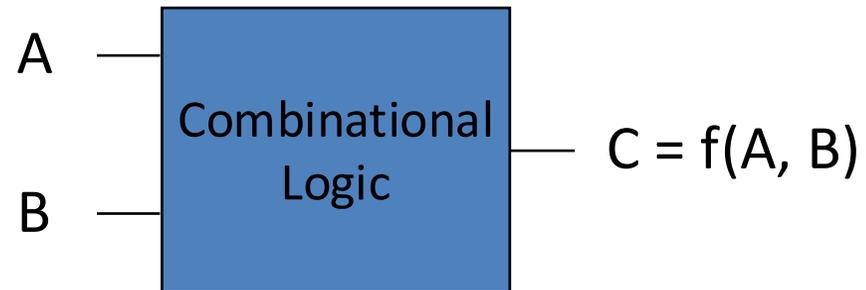
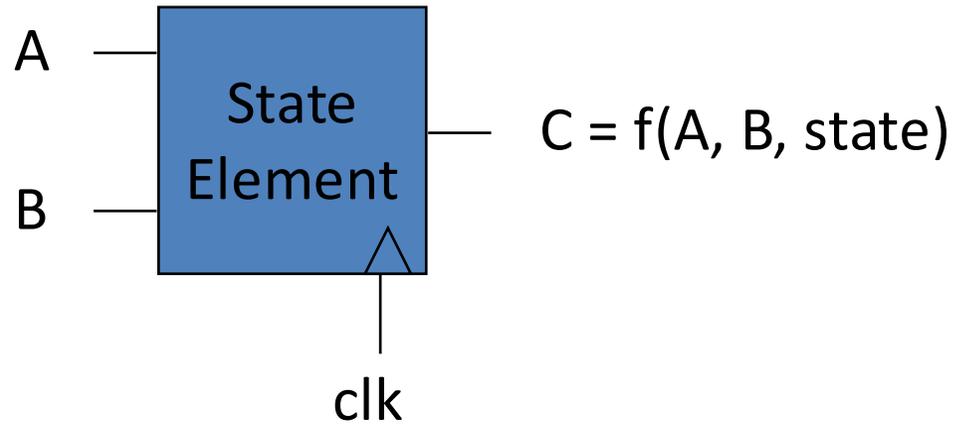
- Timing diagrams show signals changing over time
- We're looking at logical signals (0s and 1s)
- Voltage levels map to 0s and 1s
- Therefore, we can measure the voltage over time with a tool like an oscilloscope to see this with real hardware

AND gate waveforms

- Inputs
 - Yellow
 - Blue
- Output
 - Pink



Two Types of Logic Components



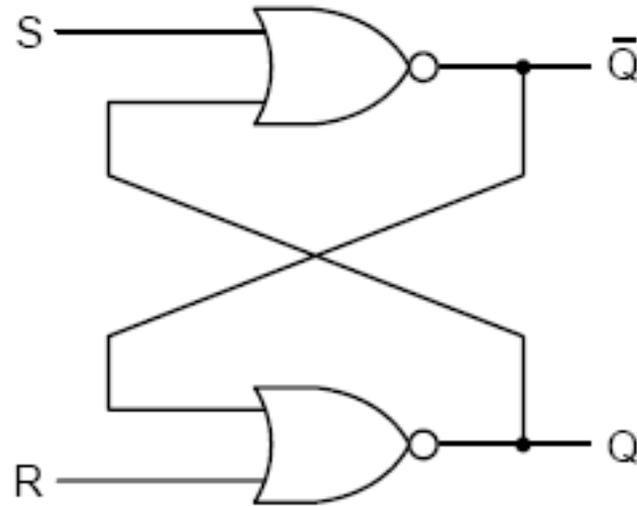
Combinational logic

- Circuits whose outputs are pure functions of its inputs
- Logical gates (AND, OR, NOT, XOR, etc.)
- Decoders, multiplexers, adders, ALUs

State Elements

- Output depends on input, AND a value saved inside the element
- State elements have *memory*

Set-Reset (S-R) Latch

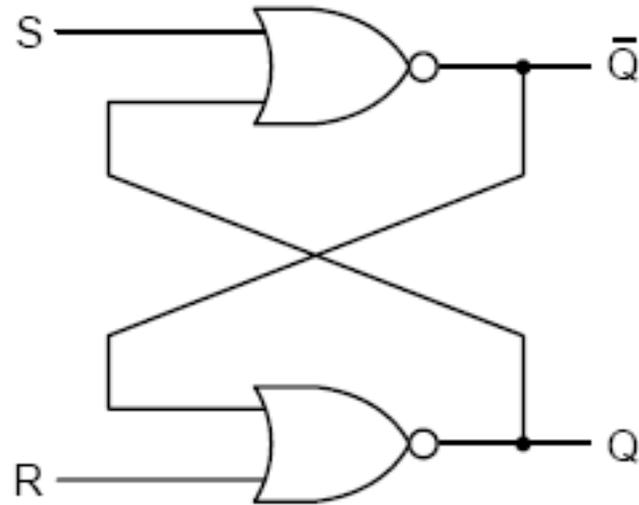


- Output (Q) depends on S, R, AND previous value of Q
- Stores 1 bit of state
- Output Q can change over time: changing inputs S or R can cause Q to change which causes the inputs to the NOR gates to change which can cause Q to change and so on

State element's state

- The *state* of an S-R latch is the value stored in the latch
- The state changes over time depending on the inputs and the current state
- For an S-R latch, the next state Q_{next} is derived from the inputs S and R and the current state Q_{curr}
- The next few questions will ask you what Q_{next} is for given S and R inputs; options are 0, 1, Q_{curr} , $\overline{Q_{\text{curr}}}$, or none of the above if something else occurs

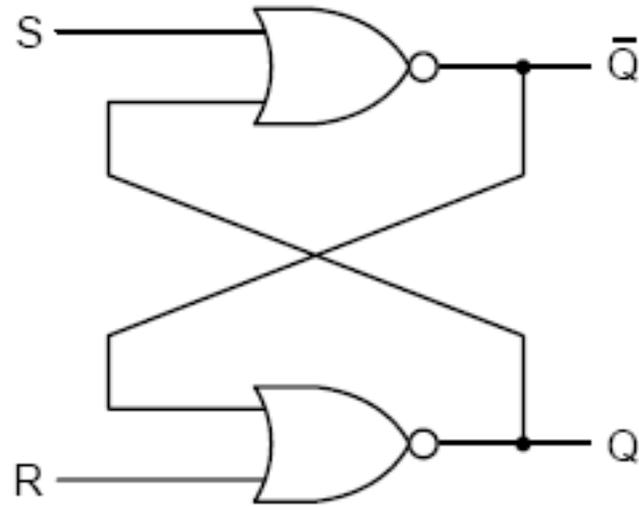
S-R Latch: Given $S = 1$, $R = 0$, what is Q_{next} ?



Consider $Q_{\text{curr}} = 0$
and $Q_{\text{curr}} = 1$

	Q_{next}
A	0
B	1
C	Q_{curr}
D	$\overline{Q_{\text{curr}}}$
E	None of the above

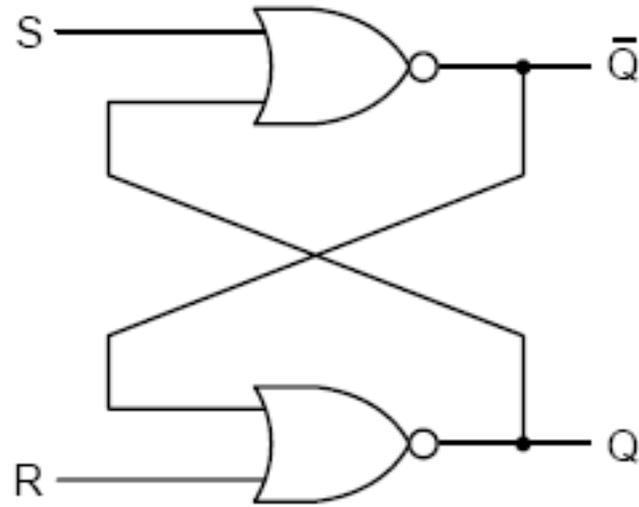
S-R Latch: Given $S = 0$, $R = 1$, what is Q_{next} ?



Consider $Q_{\text{curr}} = 0$
and $Q_{\text{curr}} = 1$

	Q_{next}
A	0
B	1
C	Q_{curr}
D	$\overline{Q_{\text{curr}}}$
E	None of the above

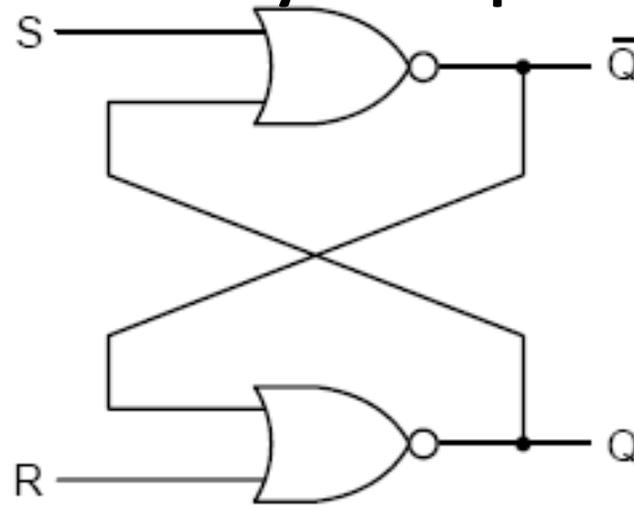
S-R Latch: Given $S = 0$, $R = 0$, what is Q_{next} ?



Consider $Q_{\text{curr}} = 0$
and $Q_{\text{curr}} = 1$

	Q_{next}
A	0
B	1
C	Q_{curr}
D	$\overline{Q_{\text{curr}}}$
E	None of the above

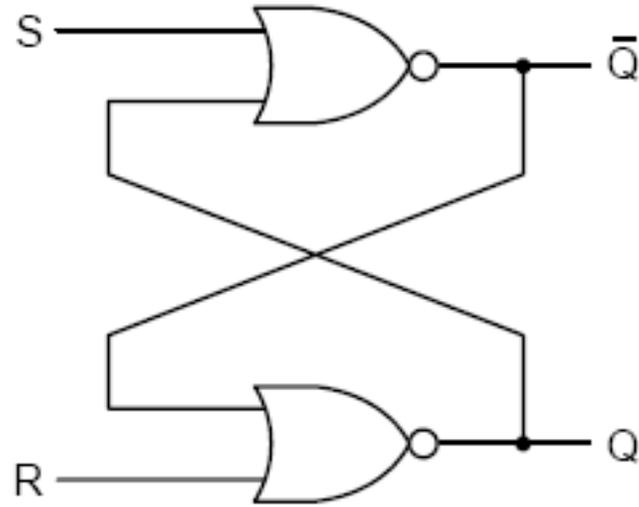
S-R Latch: Given $S = 1$, $R = 1$, what are Q_{next} and the secondary output $\overline{Q_{\text{next}}}$?



Consider $Q_{\text{curr}} = 0$
and $Q_{\text{curr}} = 1$

	Q_{next}	$\overline{Q_{\text{next}}}$
A	0	1
B	1	0
C	Q_{curr}	$\overline{Q_{\text{curr}}}$
D	$\overline{Q_{\text{curr}}}$	Q_{curr}
E	None of the above	

S-R Latch



- Set ($S=1, R=0$): $Q_{\text{next}} = 1$
- Reset ($S=0, R=1$): $Q_{\text{next}} = 0$
- Neither ($S=0, R=0$): $Q_{\text{next}} = Q_{\text{curr}}$

Terminology

- The S-R latch is a **bistable multivibrator**
 - **Bistable**: two stable states—set $Q = 1, \bar{Q} = 0$ and reset $Q = 0, \bar{Q} = 1$
 - Monostable: one stable state, one unstable state; the circuit returns to the stable state after a short time in the unstable state
 - Astable: two unstable states and the circuit switches between them
 - **Multivibrator**: a digital circuit that uses feedback
 - The name comes from the first such circuit that produced a square wave which has many harmonics, hence *multivibrateur*

Clock



- Oscillates between 1 and 0 at a set rate
- Used with elements that have memory
- Next time: we'll add a clock input to the S-R latch to make a clocked S-R latch