

# CSCI 210: Computer Architecture

## Lecture 19: State Elements

Stephen Checkoway

Oberlin College

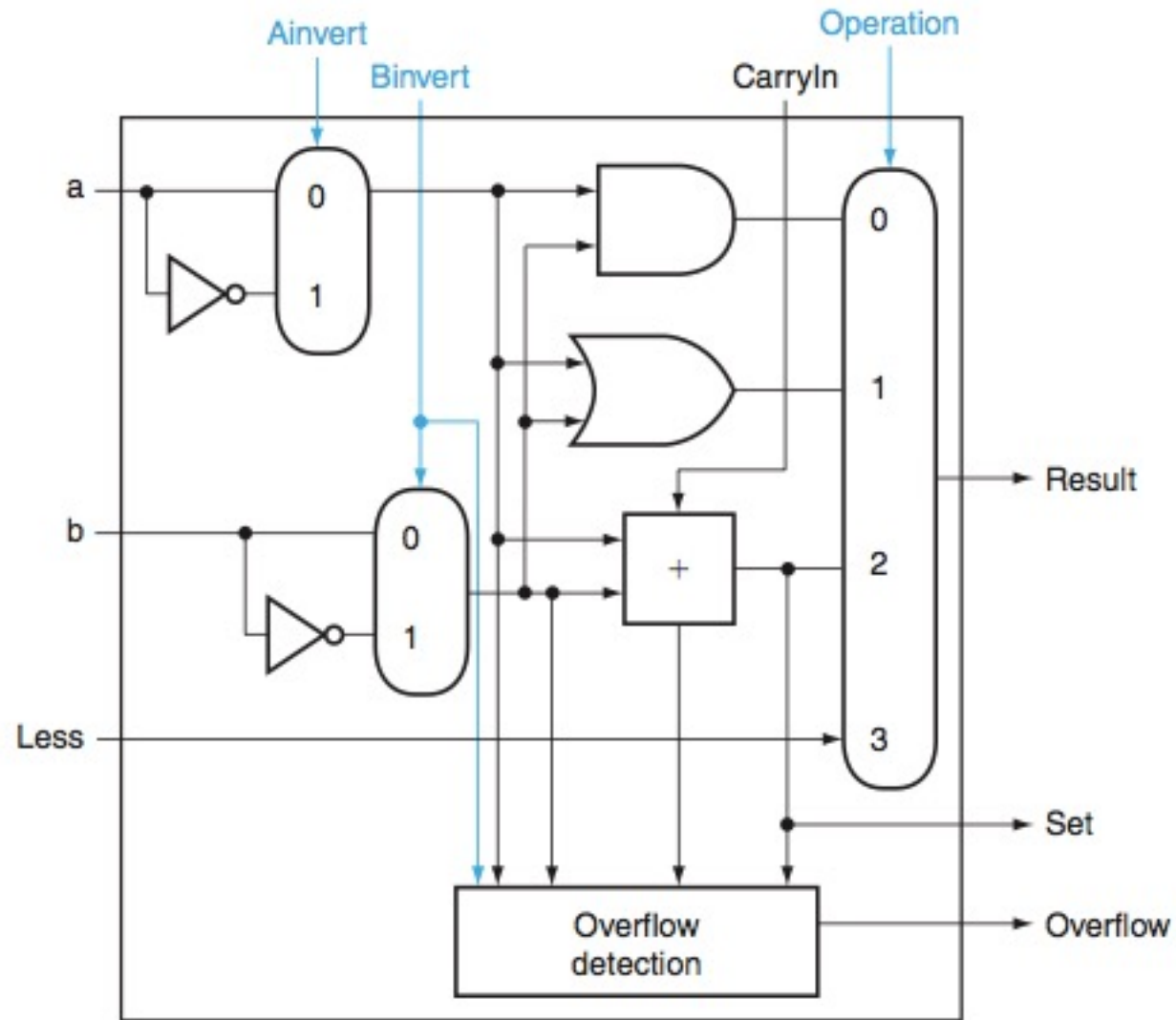
Apr. 11, 2022

Slides from Cynthia Taylor

# Announcements

- Problem Set 6 due Friday
- Lab 5 due Sunday
- Office Hours Tuesday 13:30 – 14:30

# Last Class



# Adding Conditional Branching

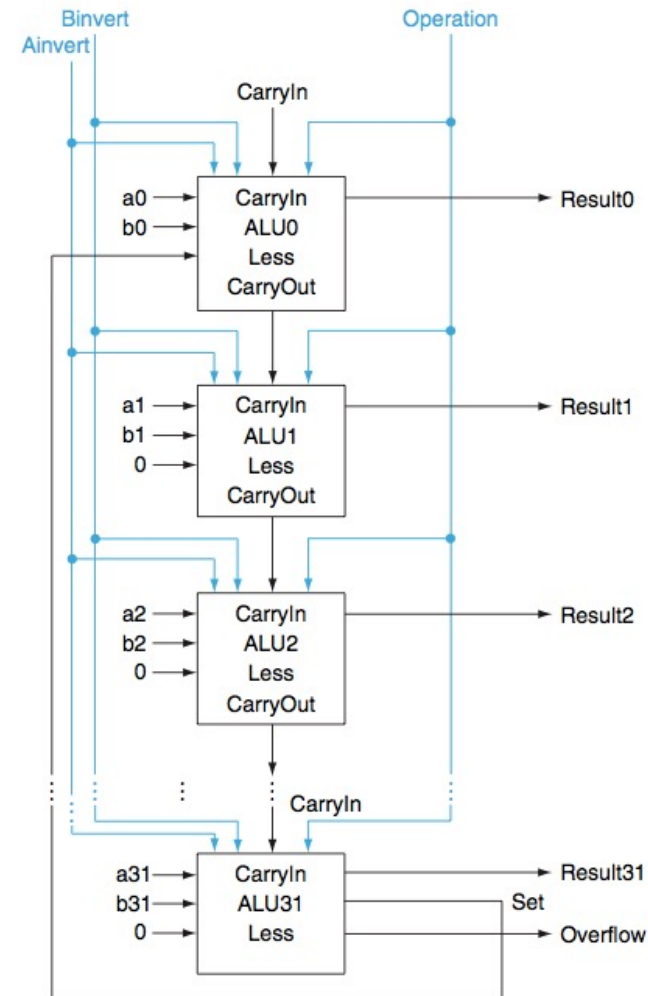
- Want to be able to support beq, bne, etc
- Need to be able to check equality
- If  $a = b$ , then  $a - b = 0$

# Detect 0 in Multi-bit ALU

- Subtract  $a - b$
- Take output from each 1-bit ALU

We know Result0-31 are 0 if we perform a \_\_\_\_\_ operation on Result0 through Result31, and it outputs \_\_\_\_\_

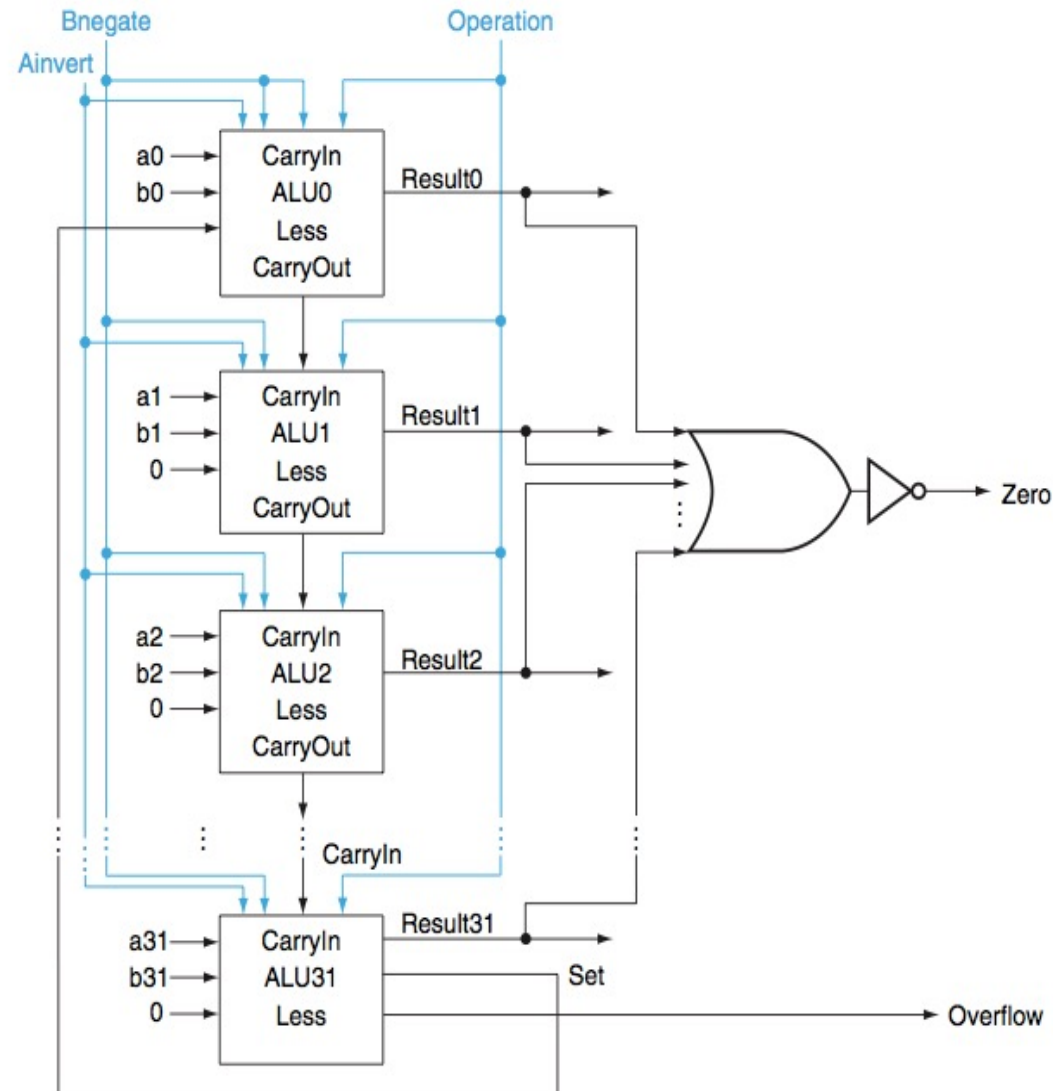
- A. AND, 0
- B. OR, 0
- C. NAND, 1
- D. XOR, 0
- E. None of the above



# Detect 0 in Multi-bit ALU

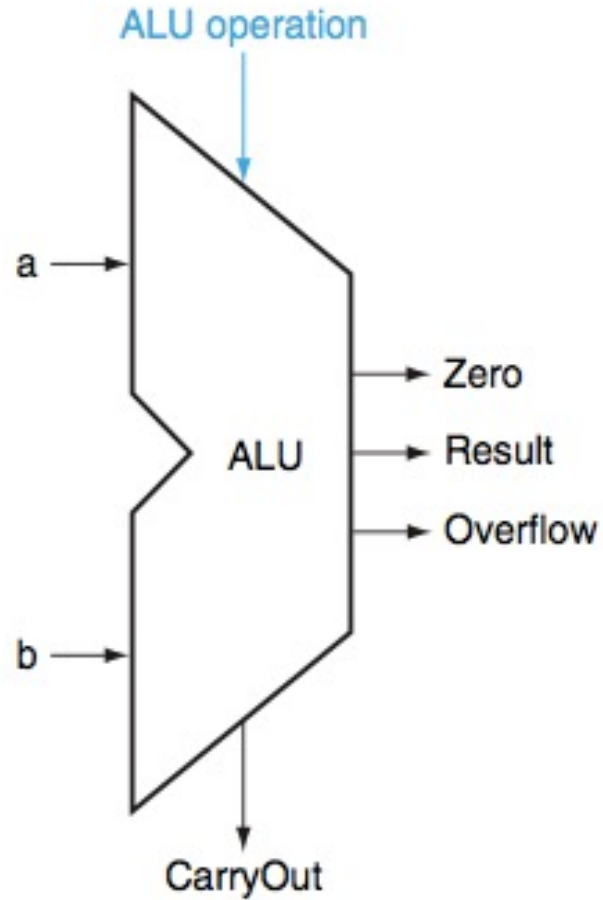
- Subtract  $a - b$
- Take output from each 1-bit ALU
- OR outputs together
  - If any output is 1, result will be 1, else 0
- Negate the result

# Multi-bit ALU with zero check

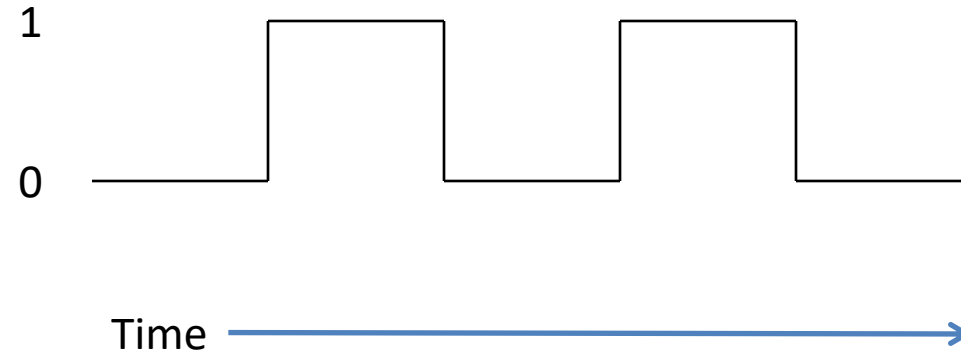




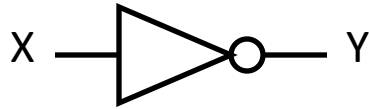
# Symbol for Multi-bit ALU



# Logic Gates and Timing Diagrams



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

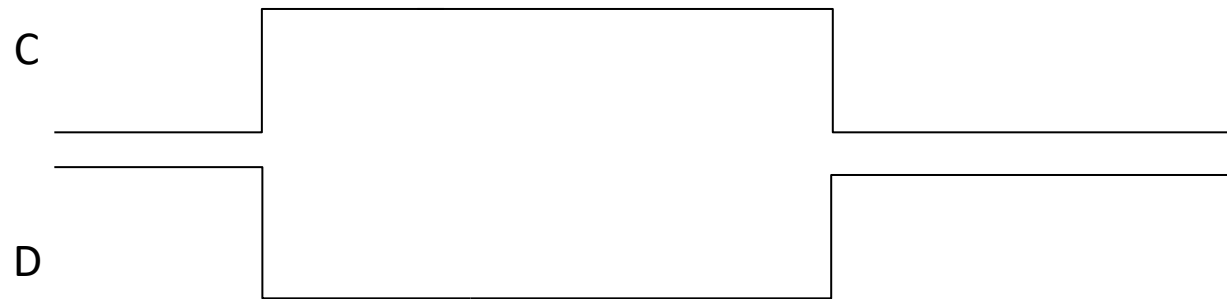
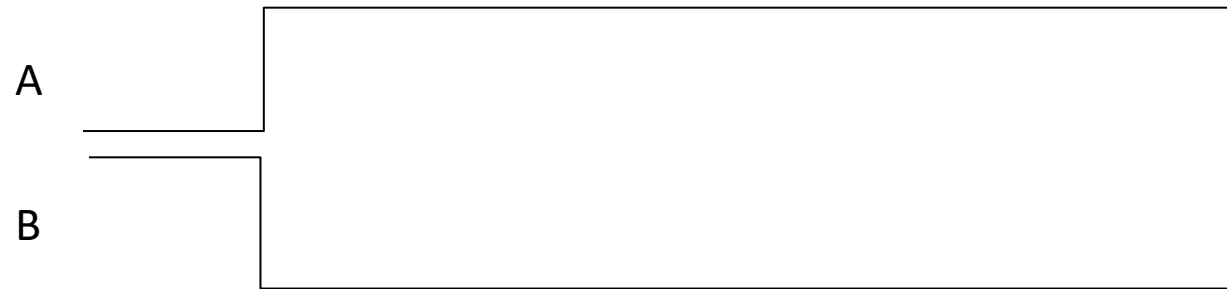


Inverter

X Y

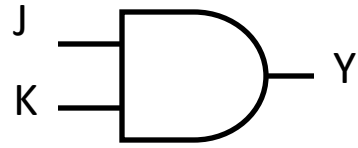
0 1

1 0



E None of the above.

# Select the correct output for Y



AND

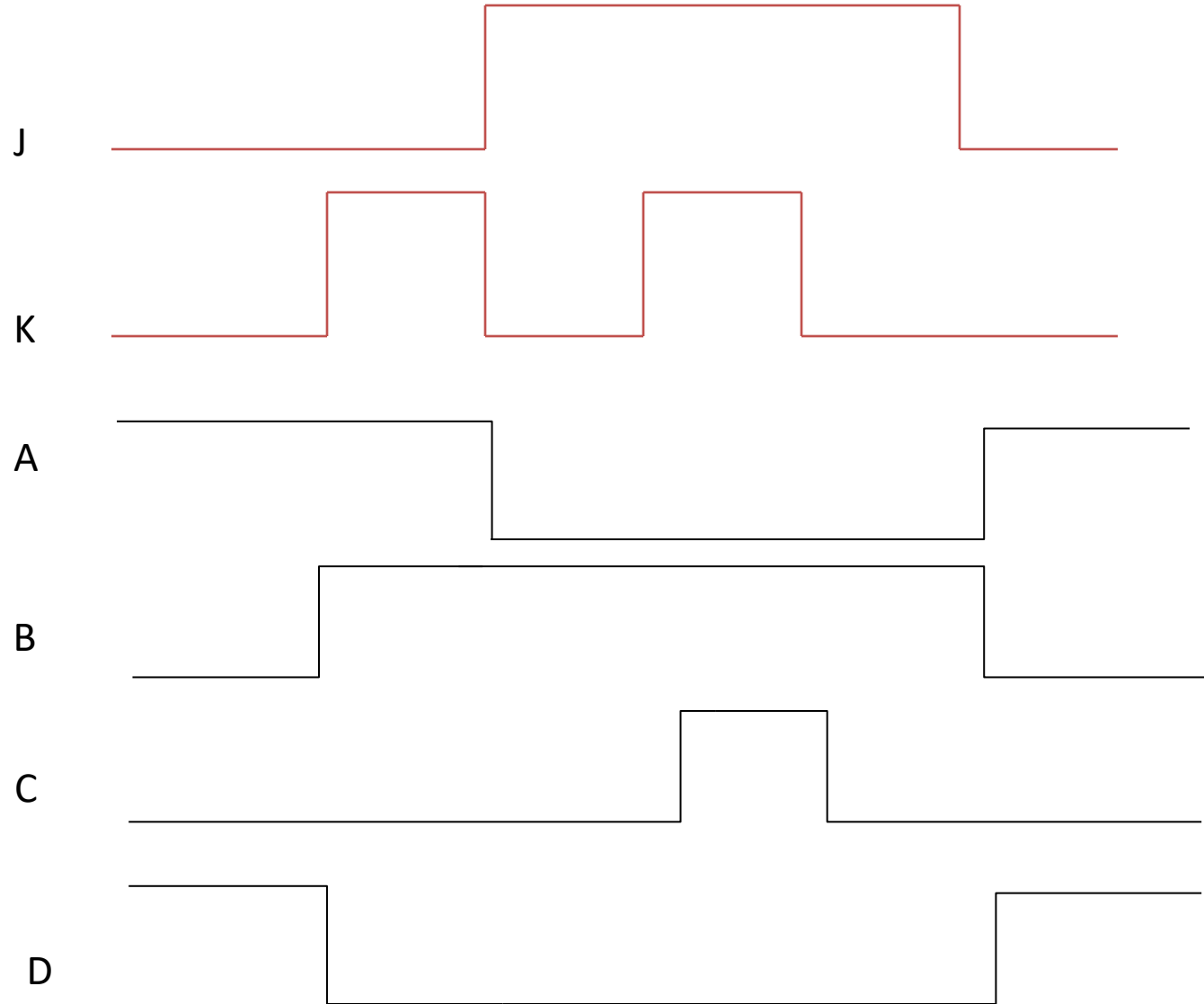
JK Y

00 0

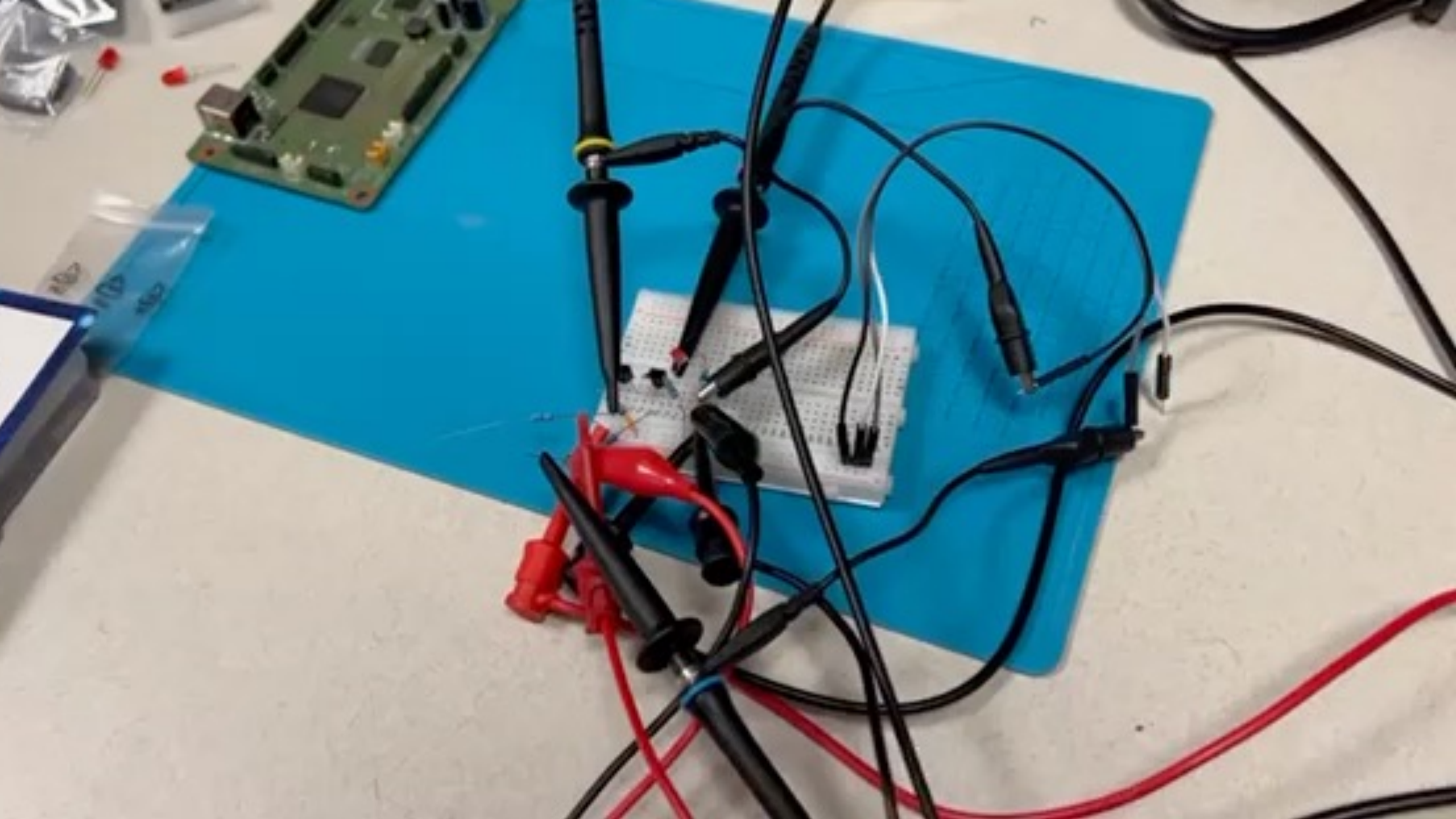
01 0

10 0

11 1

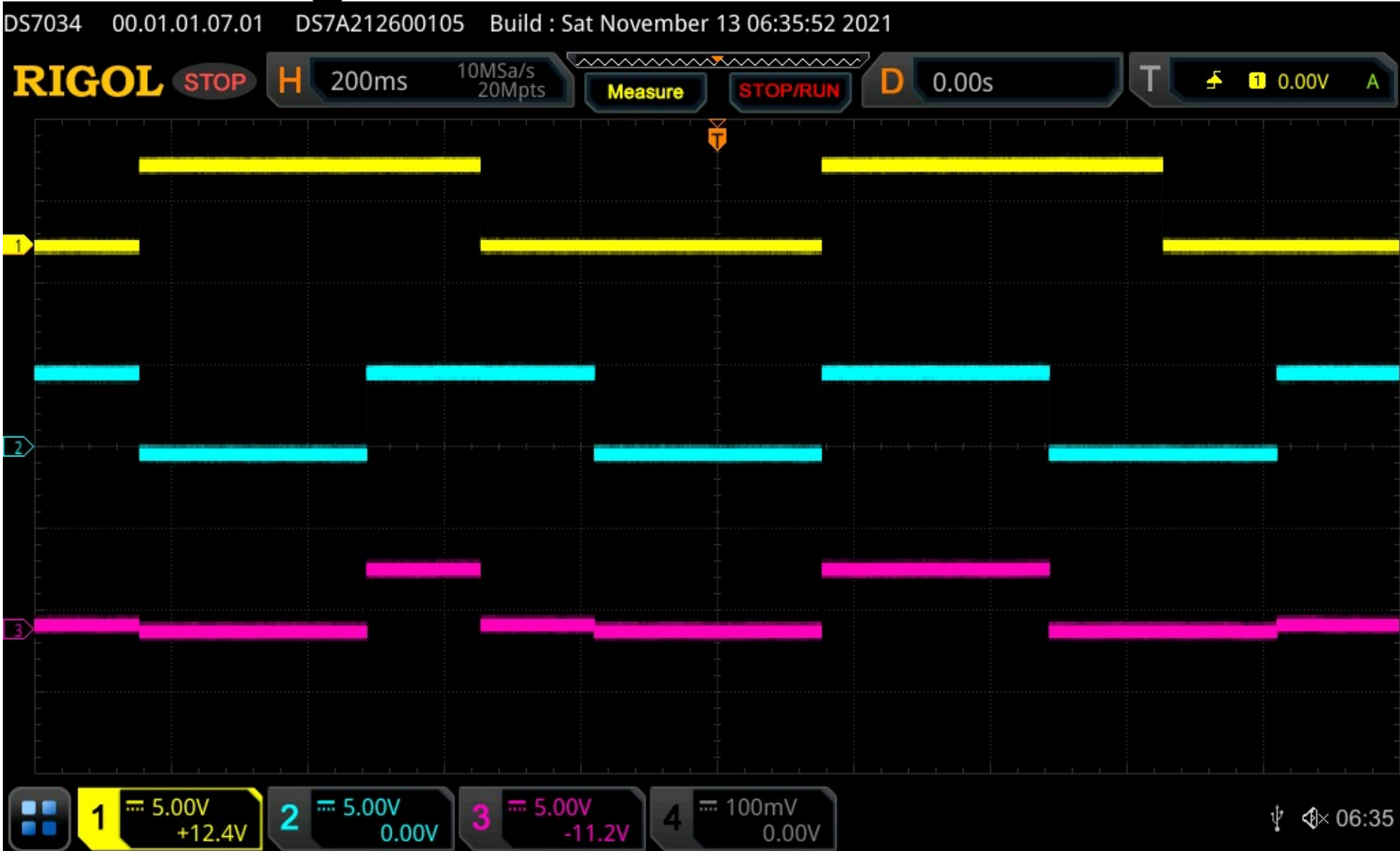


E None of the above

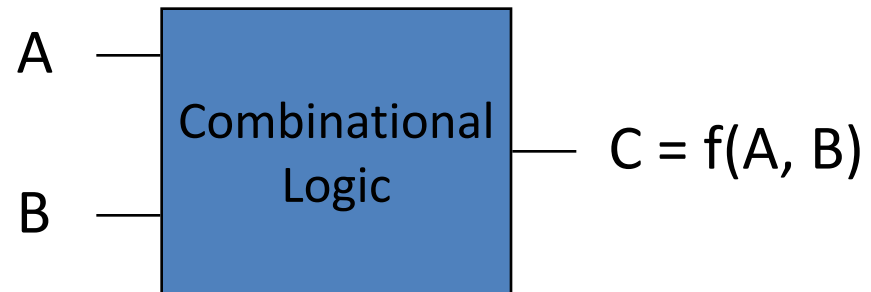
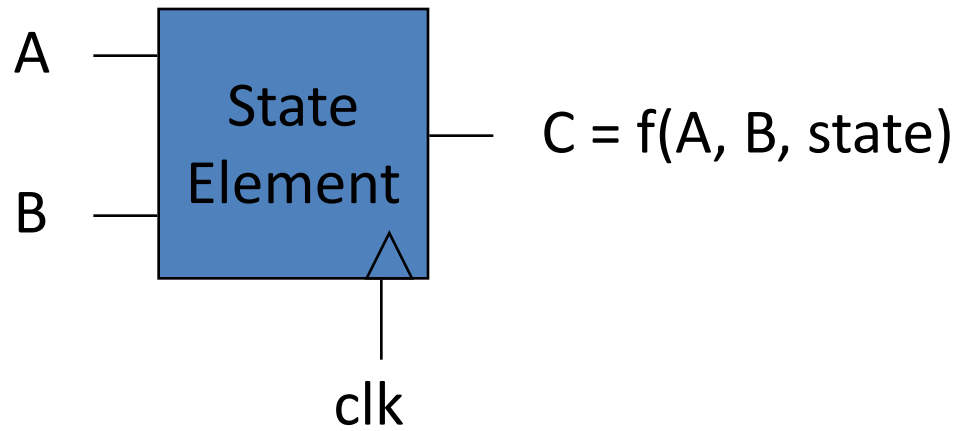


# AND gate waveforms

- Inputs
  - Yellow
  - Blue
- Output
  - Pink



# Two Types of Logic Components

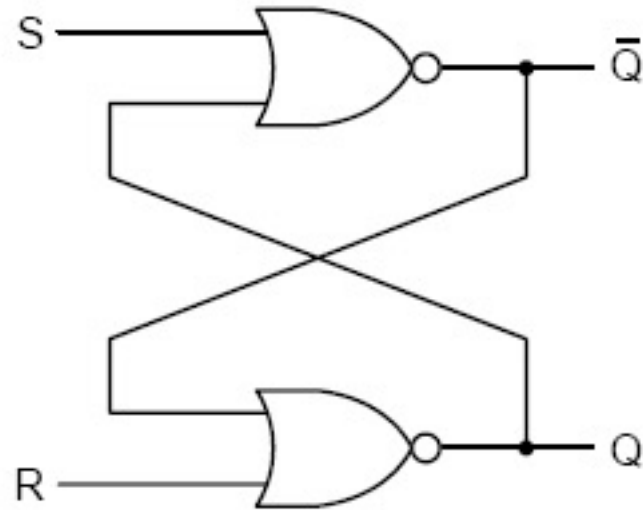


# State Elements

- Output depends on input, AND a value saved inside the element
- Have *memory*

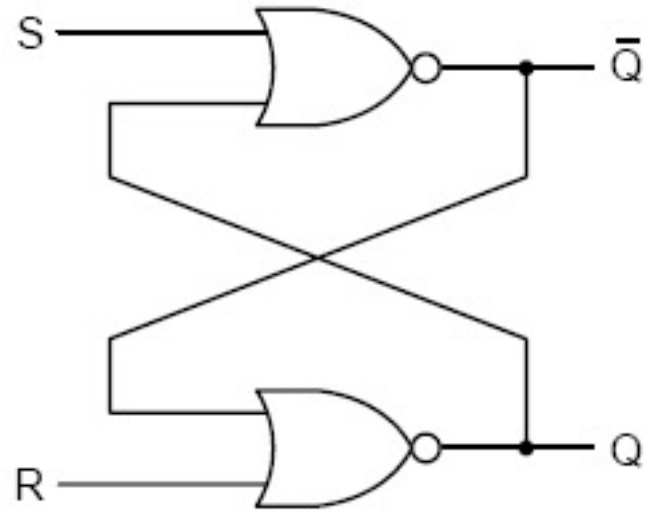


# Set-Reset (S-R) Latch



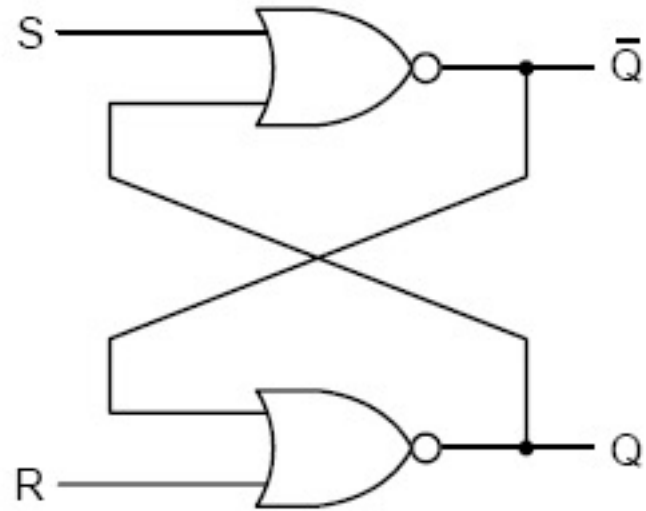
- Output depends on S, R, AND previous value of Q
- Stores 1 bit of state

# S-R Latch: $S = 1, R = 0$



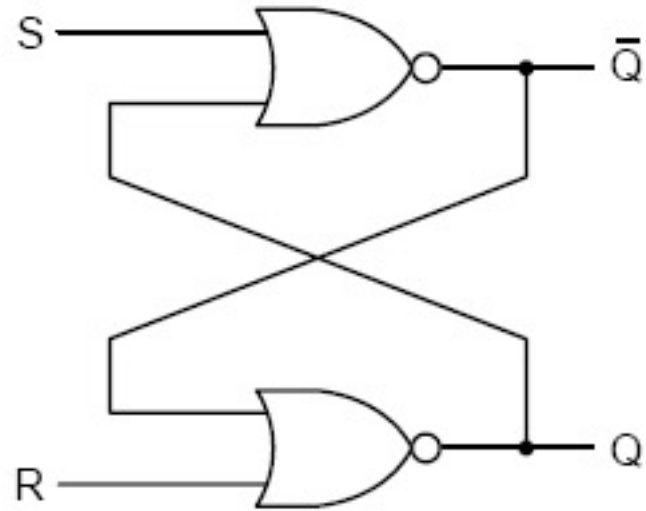
	Q
A	0
B	1
C	Q from before
D	$\bar{Q}$ from before
E	None of the above

# S-R Latch: $S = 0, R = 1$



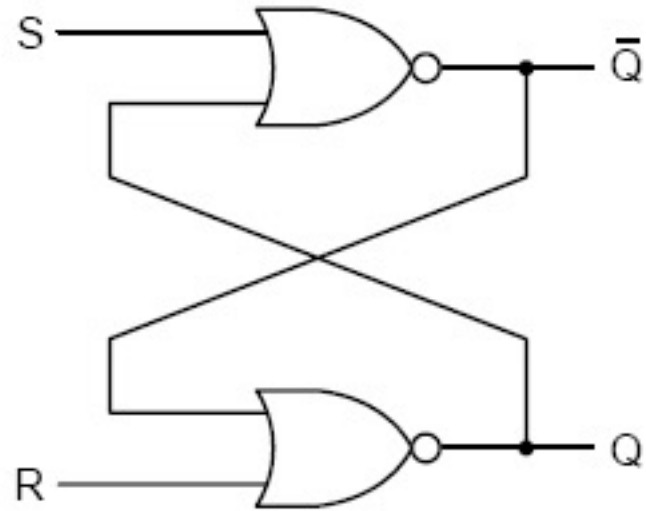
	Q
A	0
B	1
C	Q from before
D	$\bar{Q}$ from before
E	None of the above

# S-R Latch: $S = 0, R = 0$



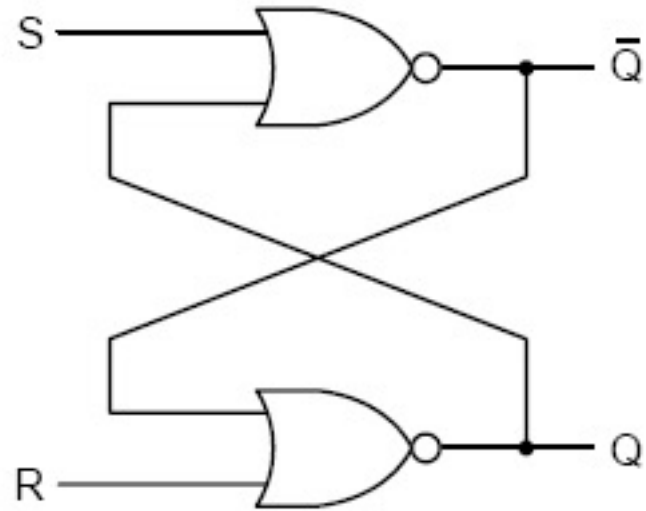
	Q
A	0
B	1
C	Q from before
D	$\bar{Q}$ from before
E	None of the above

# S-R Latch: $S = 1, R = 1$



	Q	$\bar{Q}$
A	0	1
B	1	0
C	Q from before	$\bar{Q}$ from before
D	$\bar{Q}$ from before	Q from before
E	None of the above	

# S-R Latch



- Set:  $Q_t = 1$
- Reset:  $Q_t = 0$
- Otherwise,  $Q_t = Q_{t-1}$

# 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 had many harmonics, hence *multivibrateur*

# Clock



- Oscillates between 1 and 0 at a set rate
- Used with elements that have memory



# Clocked SR Latch

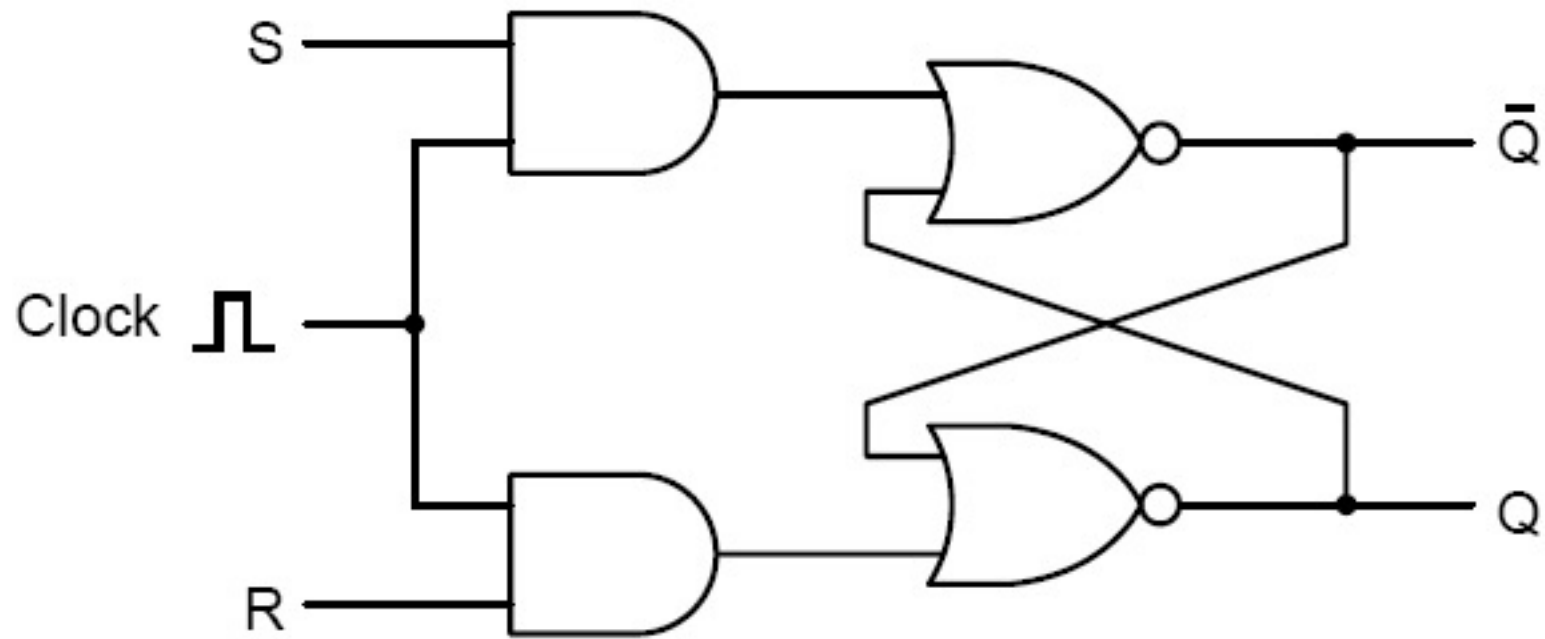


Figure 3-23. A clocked SR latch.

- Only changes state when the clock is asserted

Given S, R and Clock, Q will be:

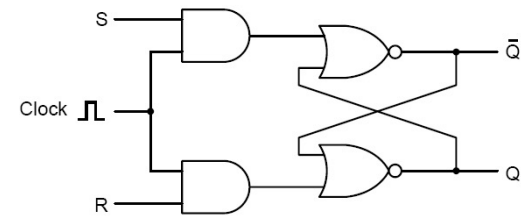
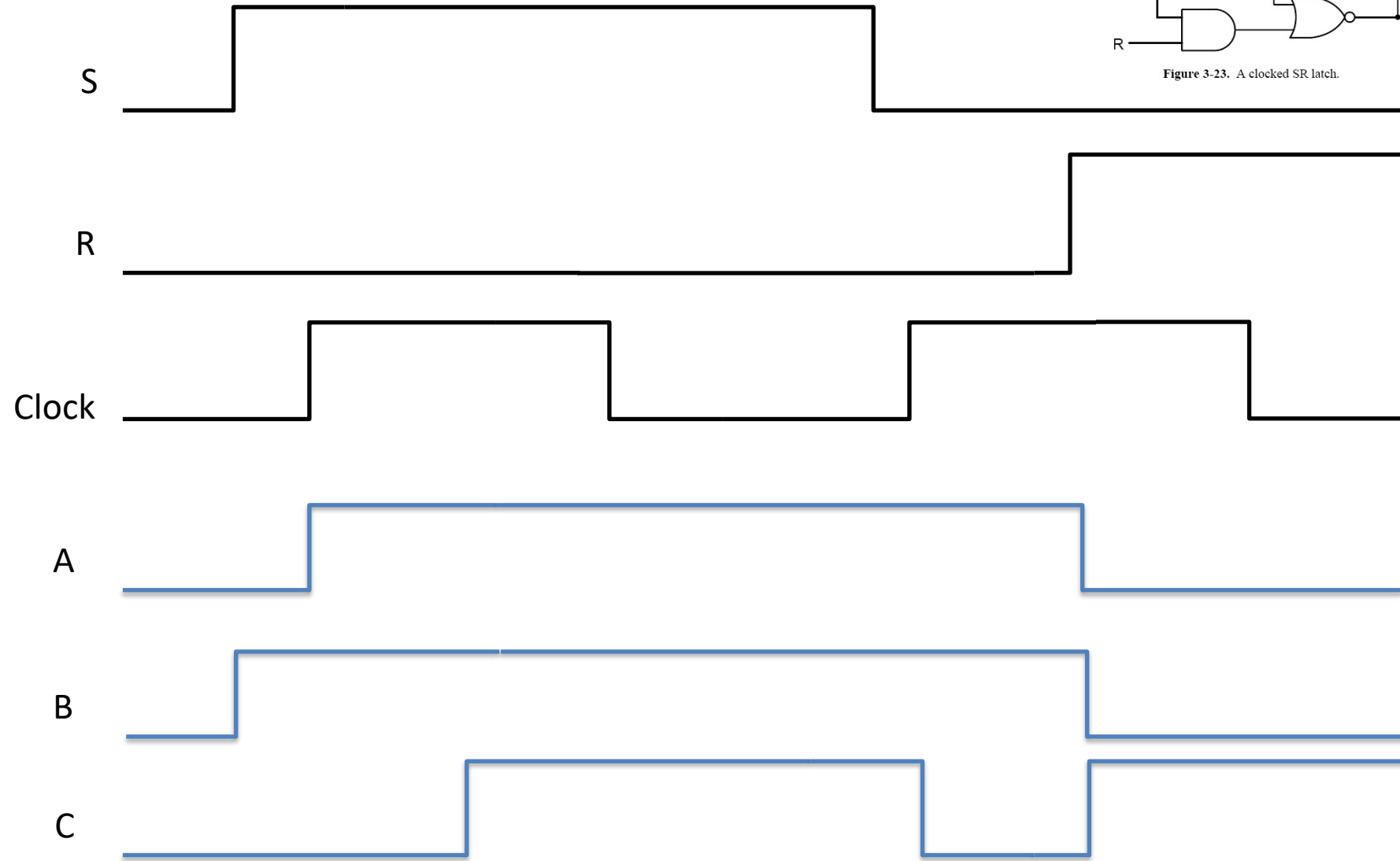


Figure 3-23. A clocked SR latch.



D. None of the above

# Reading

- Next lecture: Clocks, Latches and Flip flops  
– 3.7
- Problem Set 6 due Friday
- Lab 5 due Sunday