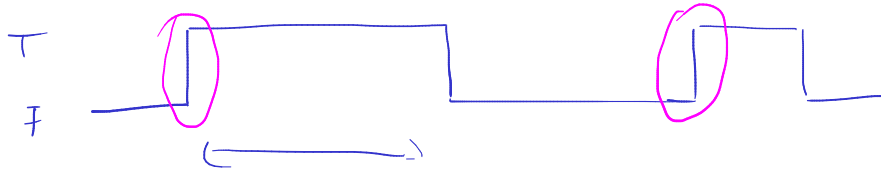


## Embedded System Design and Modeling

## IV. FSM Patterns

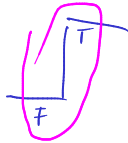
- ▶ **Design patterns:** reusable templates which appear often in applications
- ▶ Patterns
  - ▶ Operating on signal transitions
  - ~~▶ Set Reset~~
  - ▶ Debounce (one-sided, two-sided)
  - ~~▶ Check Synchronize~~

## Operating on signal transitions

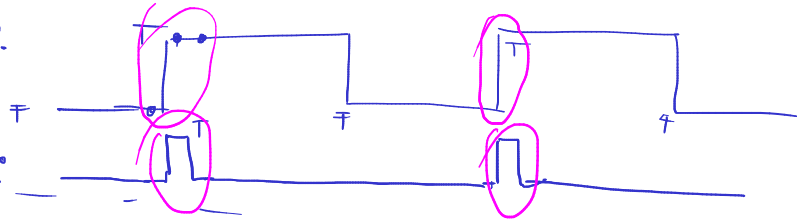


- ▶ For boolean signals/conditions
- ▶ Use when information is in the signals' **fronts** (edge / transition) rather than in its values
- ▶ Solution: detect signal transitions
  - ▶ rising edge
  - ▶ falling edge
  - ▶ both

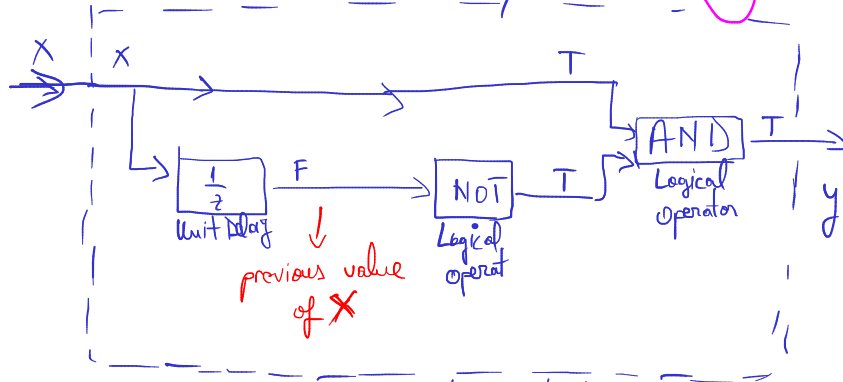
# Detect rising edge



X:

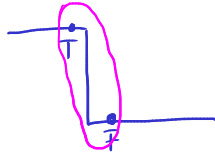


► Draw here: detect rising edge



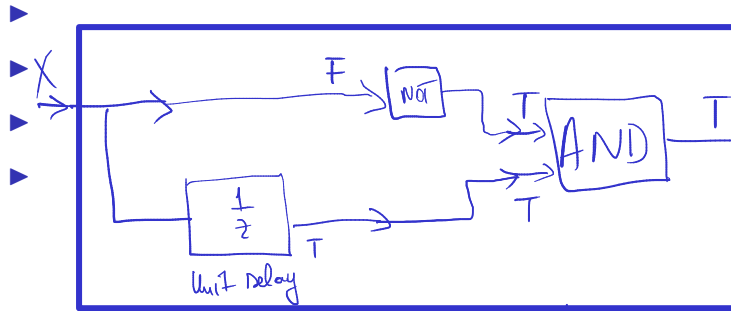
Rising Edge Detector

# Detect falling edge

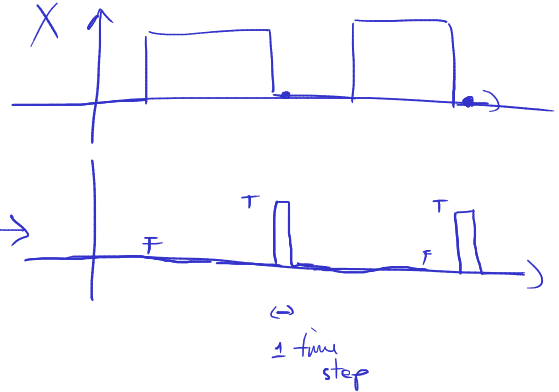


$$X \quad X_{prev}$$
$$\text{if} (X_{prev} == T \text{ AND } X == F)$$

► Draw here: detect falling edge



Falling Edge Detector

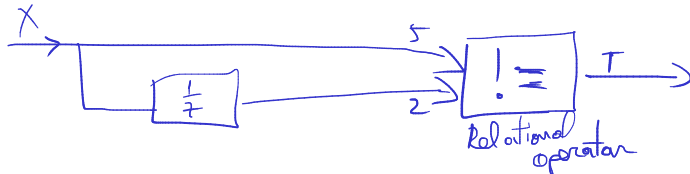
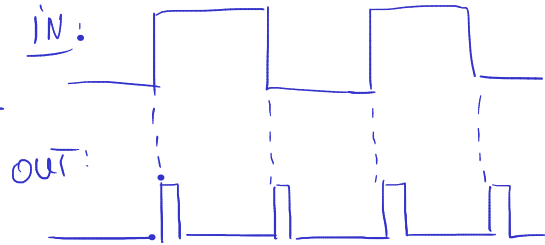
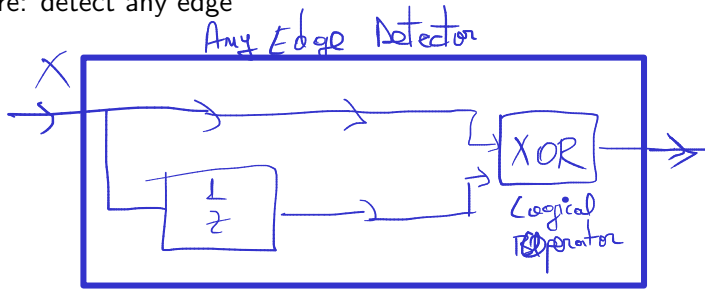


# Detect any edge



F	XOR	F	=	F
F		T	=	T
T		F	=	T
T	XOR	T	=	F

- Draw here: detect any edge



(Next Time)

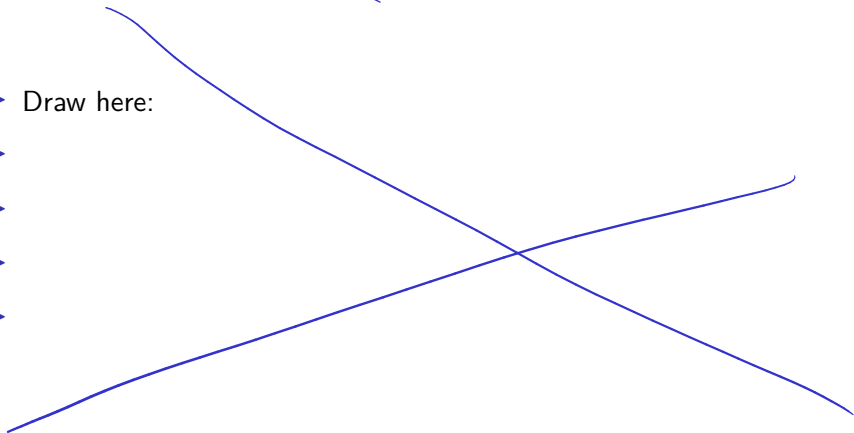
- ▶ Scenario:
  - ▶ two conditions, A and B
  - ▶ when A is true, turn something ON
  - ▶ when B is true, turn it OFF
- ▶ Use when ON-OFF behavior is controlled by two conditions (A and B)



# Set-Reset

(Next Time)

► Draw here:



# Debouncing

- ▶ For boolean signals/conditions
- ▶ Bouncing: real signals look like this:

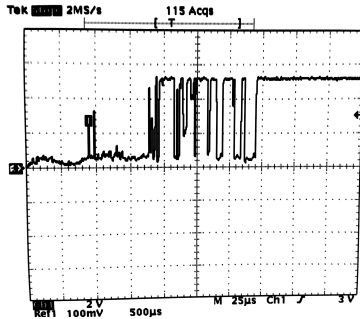
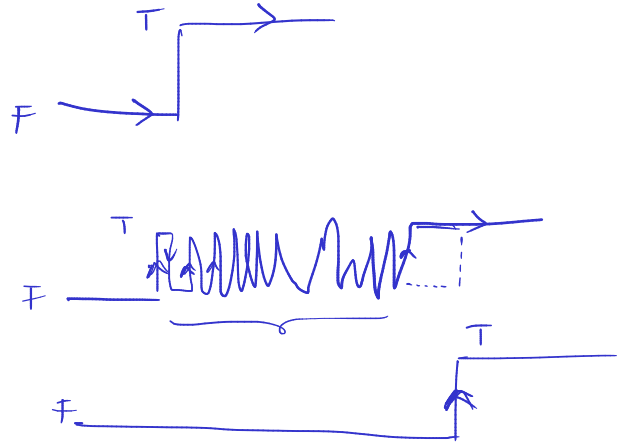


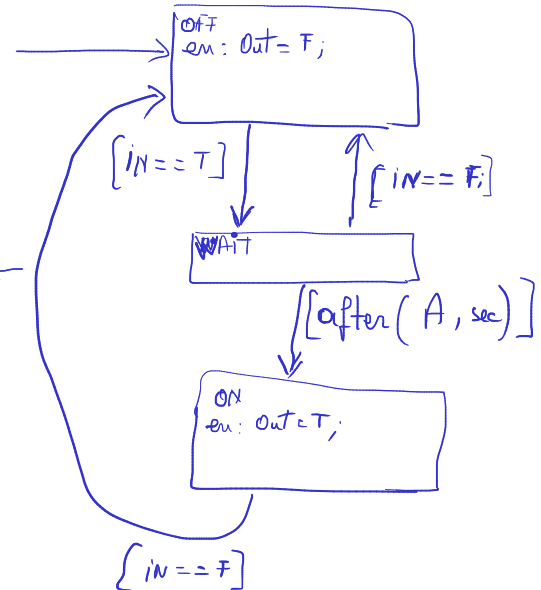
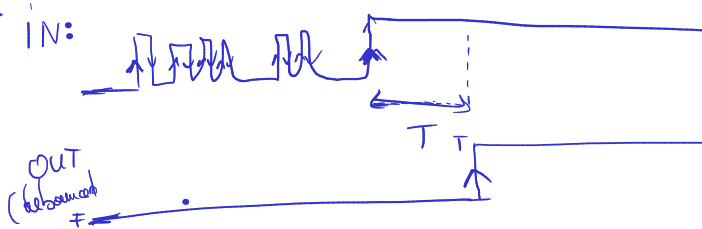
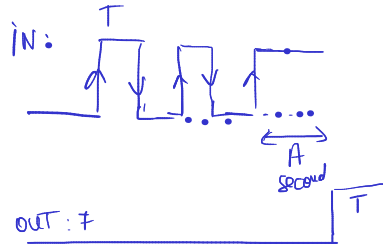
Figure 1: Signal change when pushing a button



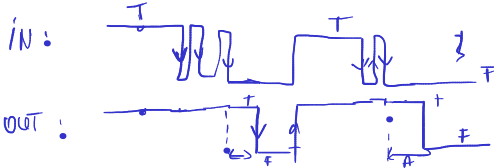
- ▶ Use debouncing to avoid spurious transitions

# Debouncing rising edge

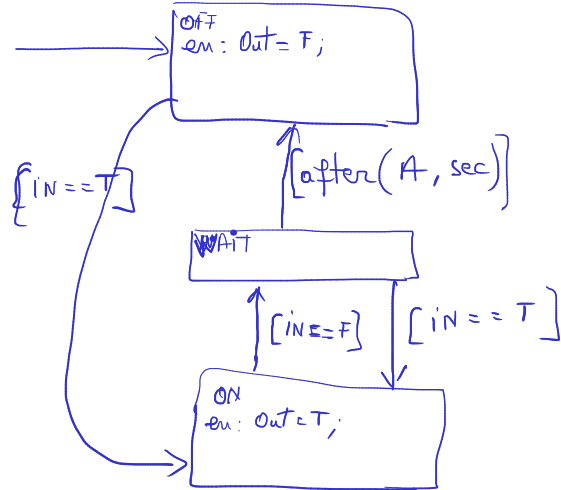
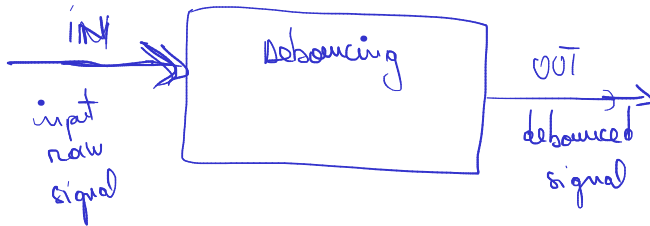
- ▶ Draw here: debounce rising edge



# Debouncing falling edge



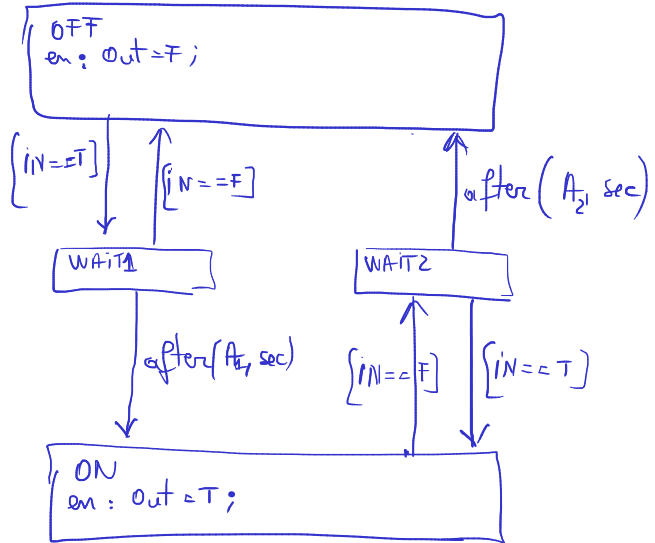
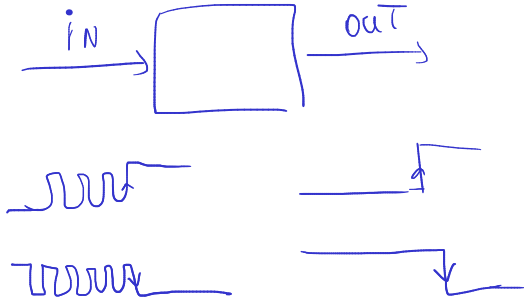
► Draw here: debounce falling edge



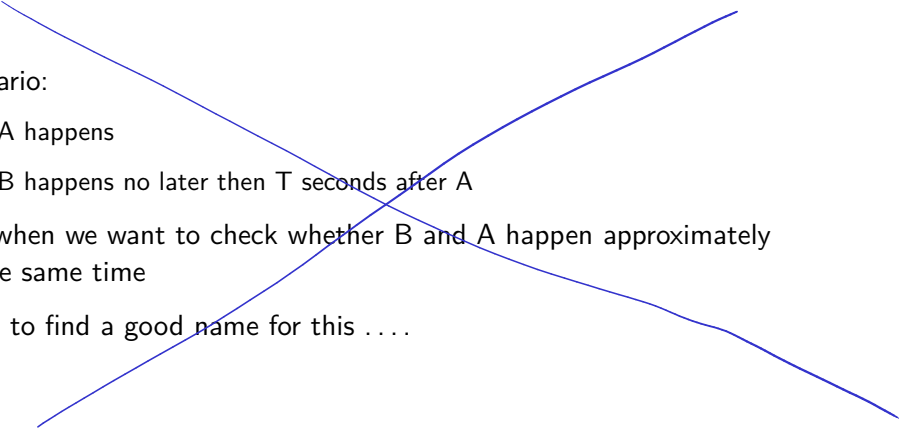
# Debouncing both edges

- Draw here: debounce both edges

- 
- 
- 
- 

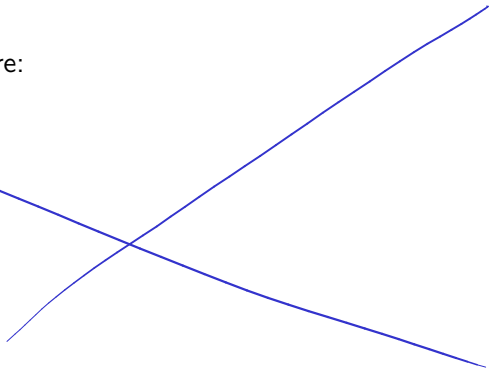


# Check-Synchronize

- 
- ▶ Scenario:
    - ▶ A happens
    - ▶ B happens no later than  $T$  seconds after A
  - ▶ Use when we want to check whether B and A happen approximately at the same time
  - ▶ Need to find a good name for this ....

# Check-Synchronize

► Draw here:



# Check-Synchronize

► Draw here:

