

**Lecture 21 – State Elements :
 Circuits that Remember**



2006-10-16

Lecturer SOE Dan Garcia

www.cs.berkeley.edu/~ddgarcia

One Laptop per Child =>

The OLPC project has been making news recently with its \$100 Linux laptop intended for every child on earth. Libya just purchased 1.2 million of them (powered by a wind-up crank). Cool!



Review

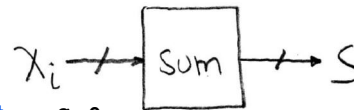
- ISA is very important abstraction layer
 - Contract between HW and SW
- Clocks control pulse of our circuits
- Voltages are analog, quantized to 0/1
- Circuit delays are fact of life
- Two types of circuits:
 - Stateless Combinational Logic (&,!,~)
 - State circuits (e.g., registers)

Uses for State Elements

1. As a place to store values for some indeterminate amount of time:
 - Register files (like \$1-\$31 on the MIPS)
 - Memory (caches, and main memory)
2. Help control the flow of information between combinational logic blocks.
 - State elements are used to hold up the movement of information at the inputs to combinational logic blocks and allow for orderly passage.

Accumulator Example

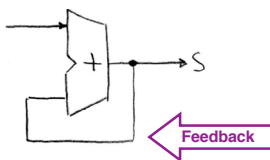
Why do we need to control the flow of information?



Want: $S=0;$
 for $(i=0; i<n; i++)$
 $S = S + X_i$

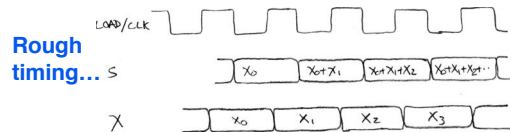
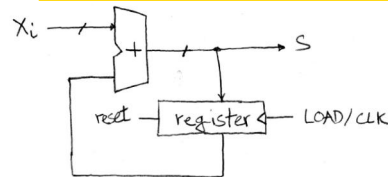
Assume:
 • Each X value is applied in succession, one per cycle.
 • After n cycles the sum is present on S.

First try...Does this work?

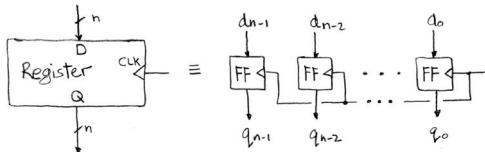


Nope!
 Reason #1... What is there to control the next iteration of the 'for' loop?
 Reason #2... How do we say: 's=0'?

Second try...How about this?



Register Details...What's inside?



- n instances of a "Flip-Flop"
- Flip-flop name because the output flips and flops between 0,1
- D is "data", Q is "output"
- Also called "d-type Flip-Flop"



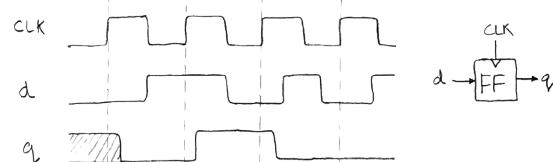
CS61C L21 State Elements - Circuits that Remember (7)

Garcia, Fall 2006 © UCB

What's the timing of a Flip-flop? (1/2)

- Edge-triggered d-type flip-flop
 - This one is "positive edge-triggered"
- "On the rising edge of the clock, the input d is sampled and transferred to the output. At all other times, the input d is ignored."

• Example waveforms:

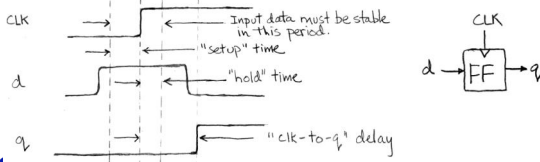


CS61C L21 State Elements - Circuits that Remember (8)

Garcia, Fall 2006 © UCB

What's the timing of a Flip-flop? (2/2)

- Edge-triggered d-type flip-flop
 - This one is "positive edge-triggered"
- "On the rising edge of the clock, the input d is sampled and transferred to the output. At all other times, the input d is ignored."
- Example waveforms (more detail):



CS61C L21 State Elements - Circuits that Remember (9)

Garcia, Fall 2006 © UCB

Administrivia -

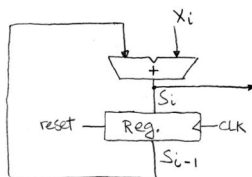
- Midterm tonight!!!
 - 7-10pm HERE (2050 VLSB)
 - DO NOT PANIC (remember clobbering)
 - You can get a 0 on this midterm and still get the highest overall grade in the class!



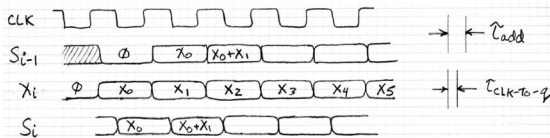
CS61C L21 State Elements - Circuits that Remember (10)

Garcia, Fall 2006 © UCB

Accumulator Revisited (proper timing 1/2)



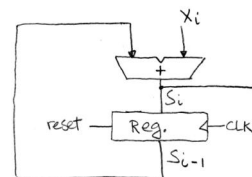
- Reset input to register is used to force it to all zeros (takes priority over D input).
- S_i holds the result of the i^{th} -1 iteration.
- Analyze circuit timing starting at the output of the register.



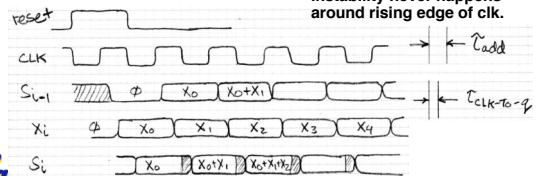
CS61C L21 State Elements - Circuits that Remember (11)

Garcia, Fall 2006 © UCB

Accumulator Revisited (proper timing 2/2)



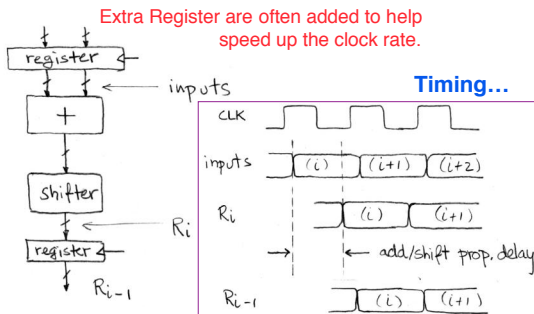
- reset signal shown.
- Also, in practice X might not arrive at the adder at the same time as S_{i-1}
- S_i temporarily is wrong, but register always captures correct value.
- In good circuits, instability never happens around rising edge of clk.



CS61C L21 State Elements - Circuits that Remember (12)

Garcia, Fall 2006 © UCB

Pipelining to improve performance (1/2)



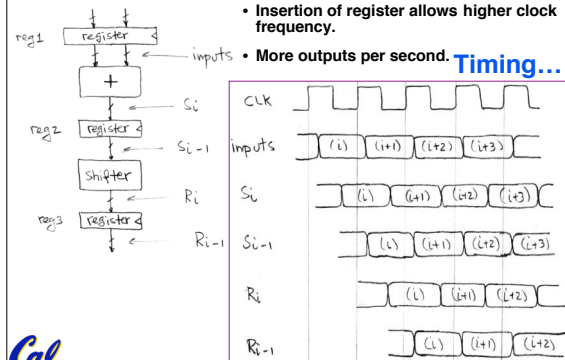
Note: delay of 1 clock cycle from input to output.
Clock period limited by propagation delay of adder/shifter.



CS61C L21 State Elements - Circuits that Remember (13)

Garcia, Fall 2006 © UCB

Pipelining to improve performance (2/2)

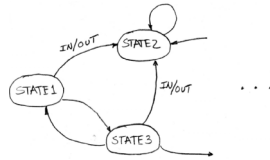


CS61C L21 State Elements - Circuits that Remember (14)

Garcia, Fall 2006 © UCB

Finite State Machines (FSM) Introduction

- You have seen FSMs in other classes.
- Same basic idea.
- The function can be represented with a "state transition diagram".
- With combinational logic and registers, any FSM can be implemented in hardware.



CS61C L21 State Elements - Circuits that Remember (15)

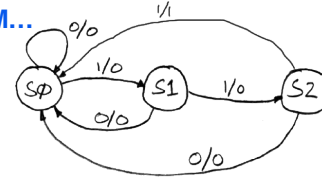
Garcia, Fall 2006 © UCB

Finite State Machine Example: 3 ones...

FSM to detect the occurrence of 3 consecutive 1's in the input.



Draw the FSM...



Assume state transitions are controlled by the clock:
on each clock cycle the machine checks the inputs and moves to a new state and produces a new output...

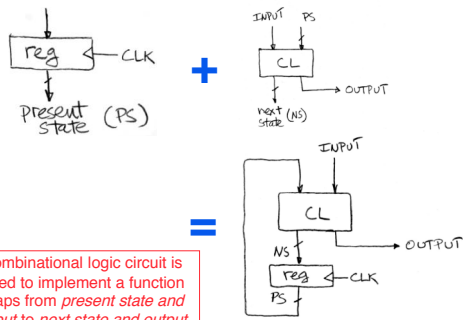


CS61C L21 State Elements - Circuits that Remember (16)

Garcia, Fall 2006 © UCB

Hardware Implementation of FSM

... Therefore a register is needed to hold the a representation of which state the machine is in. Use a unique bit pattern for each state.



Combinational logic circuit is used to implement a function maps from present state and input to next state and output.

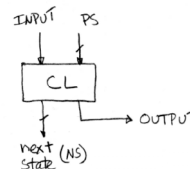


CS61C L21 State Elements - Circuits that Remember (17)

Garcia, Fall 2006 © UCB

Hardware for FSM: Combinational Logic

Next lecture we will discuss the detailed implementation, but for now can look at its functional specification, truth table form.



Truth table...

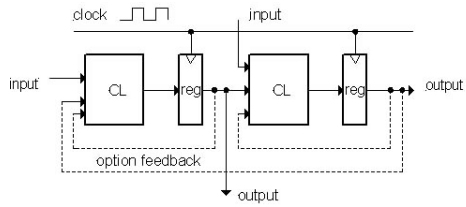
PS	Input	NS	Output
00	0	00	0
00	1	01	0
01	0	00	0
01	1	10	0
10	0	00	0
10	1	00	1



CS61C L21 State Elements - Circuits that Remember (18)

Garcia, Fall 2006 © UCB

General Model for Synchronous Systems



- Collection of CL blocks separated by registers.
- Registers may be back-to-back and CL blocks may be back-to-back.
- Feedback is optional.
- Clock signal(s) connects only to clock input of registers.



CS61C L21 State Elements - Circuits that Remember (19)

Garcia, Fall 2006 © UCB

Peer Instruction

- A. HW feedback akin to SW recursion
- B. We can implement a D-Q flipflop as simple CL (And, Or, Not gates)
- C. You can build a FSM to signal when an equal number of 0s and 1s has appeared in the input.

	ABC
1:	FFF
2:	FTF
3:	FTT
4:	FTT
5:	FTT
6:	FTT
7:	FTT
8:	FTT



CS61C L21 State Elements - Circuits that Remember (20)

Garcia, Fall 2006 © UCB

“And In conclusion...”

- State elements are used to:
 - Build memories
 - Control the flow of information between other state elements and combinational logic
- D-flip-flops used to build registers
- Clocks tell us when D-flip-flops change
 - Setup and Hold times important
- We pipeline long-delay CL for faster clock
- Finite State Machines extremely useful



You'll see them again (150,152) & 164

CS61C L21 State Elements - Circuits that Remember (22)

Garcia, Fall 2006 © UCB