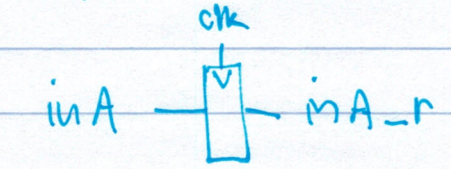
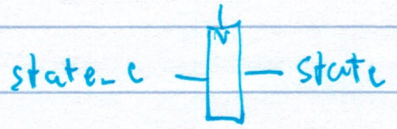
EEC 180

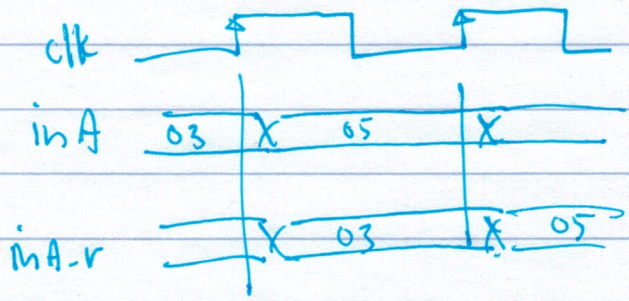
April 27, 2021

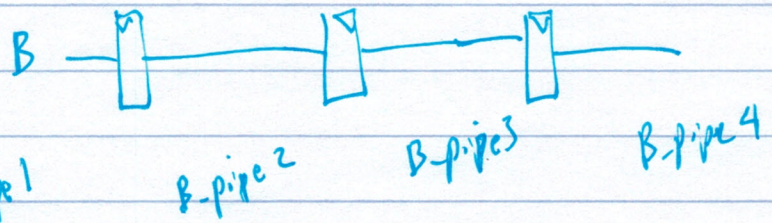
Rules of FFs

1) F we only FFs
2) For comb. logic, use "=" (vgl)
3) For FFs, use "<=" (regs)
4) Add #1 clk-to-Q
5) Avoid logic in FF decl. block
6) clustering
7) 2 clock rules
8) Signal name conventions

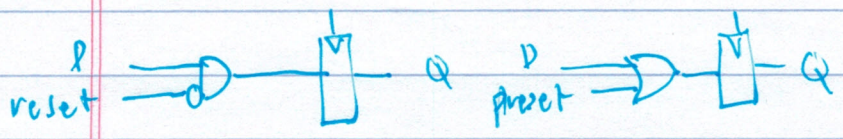


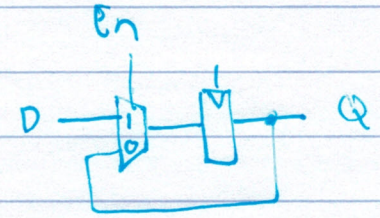






Reset, Preset, Enable

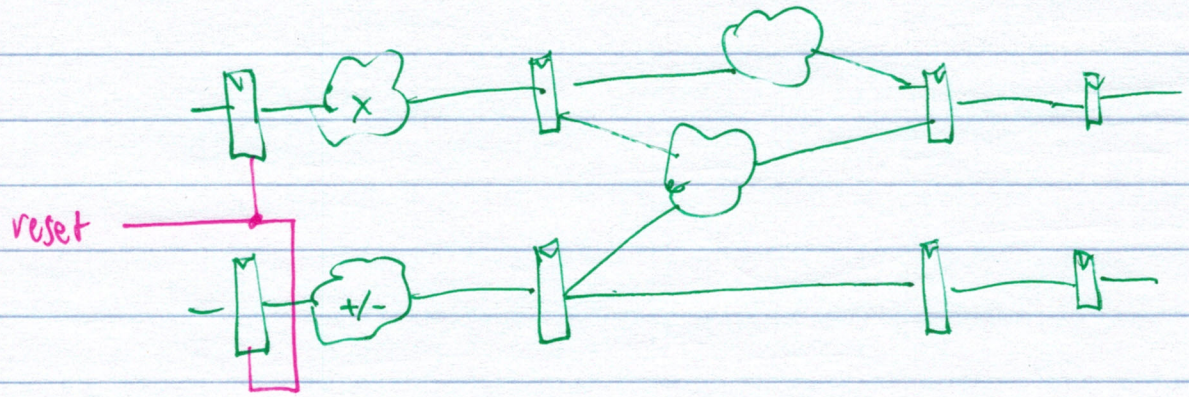




```
always @ (posedge clk) begin
  if (En == 1'b1) begin
    out <= #1 D;
  end
end
```

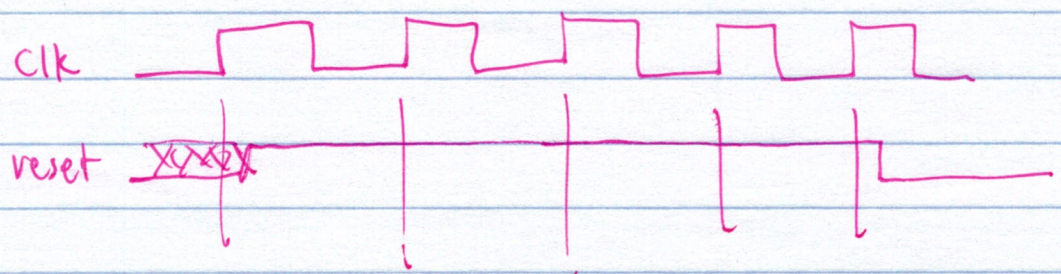
Compare w/ Mistake #4

reset	enable	Action
0	0	nothing
0	1	D FF
1	0	reset or do nothing
1	1	reset, Q=0

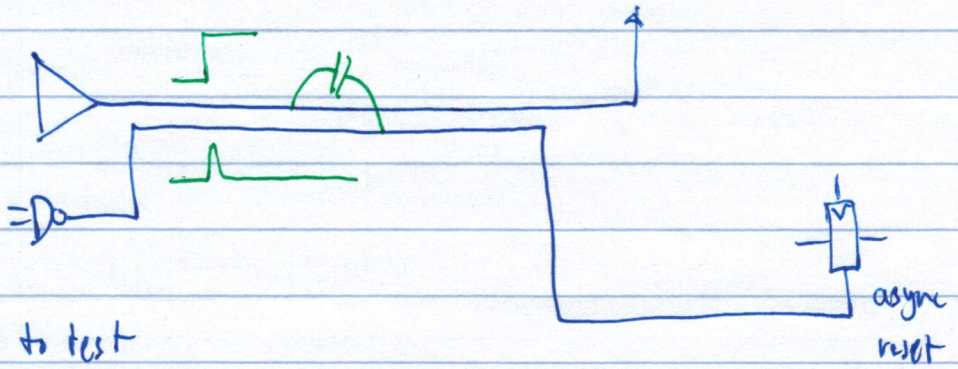


- reset only key registers
- multi-cycle reset

- Min. # of resettable FF

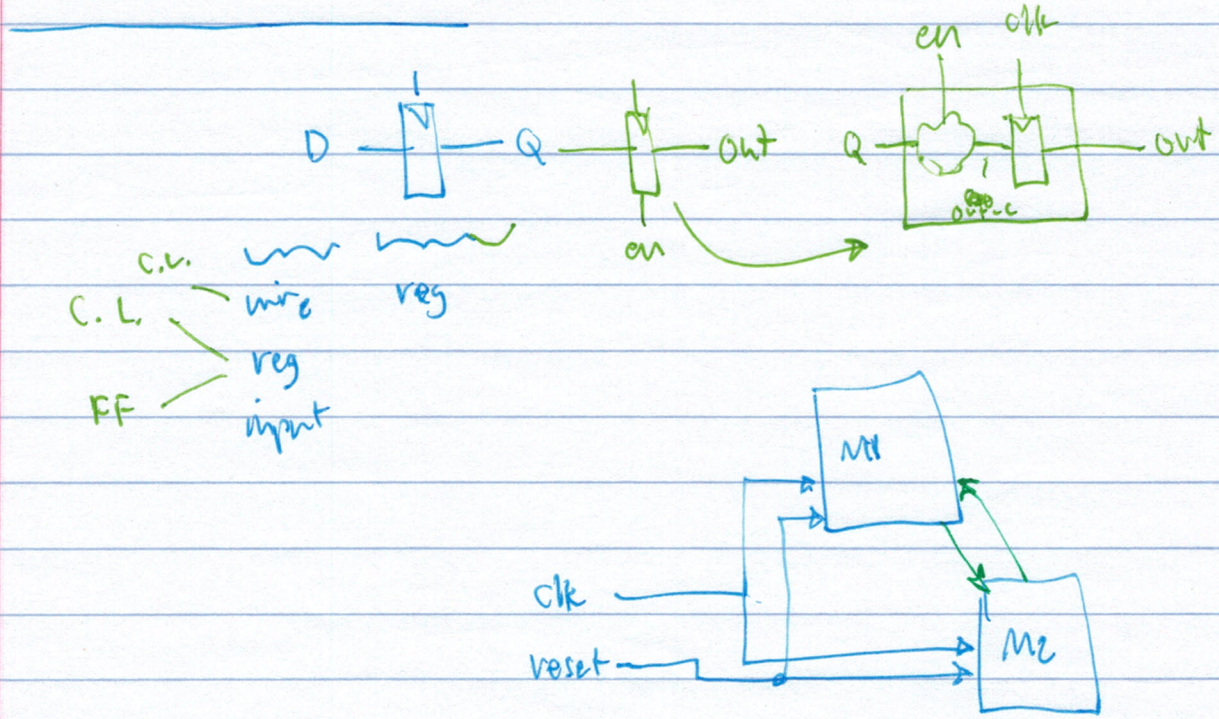


Asynchronous resets
- glitch on reset signal



- difficult to test
- needed in some cases
  - clk generator

Rule #9 : Use only synch reset in RTL



4 things in verilog

1) wire x;
assign x = ~~~;

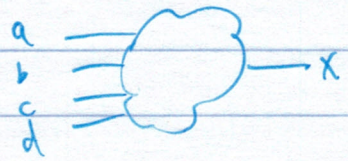
2) reg. y;
always ~~~

end

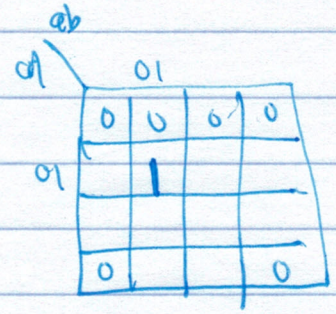
3) reg z;
always @(pos ~~~) ~

~~~

4) module instantiation



a b c d	x
:	0
	0
	0
0 1 0 1	1
	0
:	:
	0



always @(a or b or c or d) begin
out = 1'b0;
if (~~~) ~
out = 1'b1;
end
end
~~~

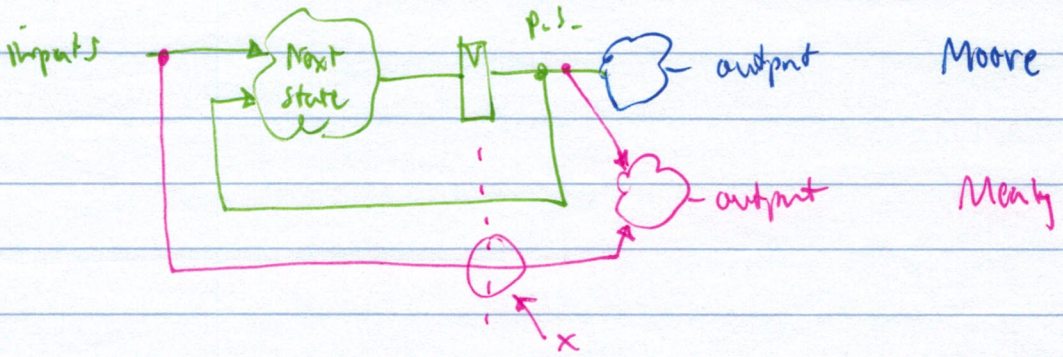
Digital Systems

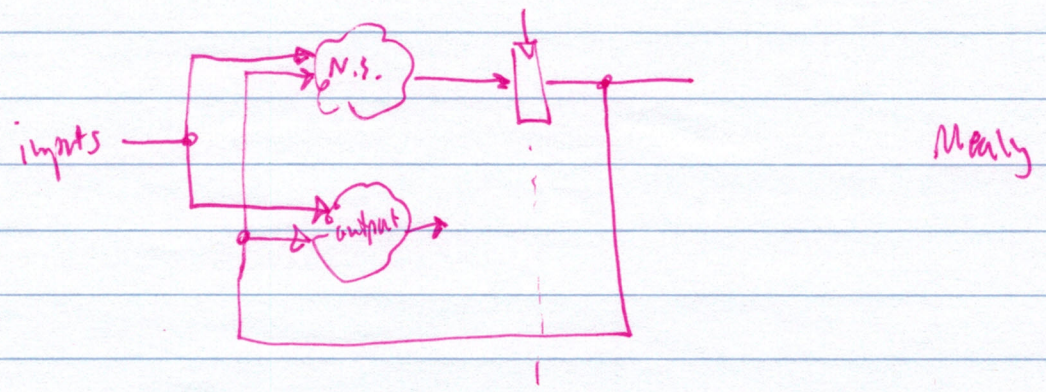
1) Datapath ✓
2) Memory ✓
3) Control

Typically small HW
" large verilog

∴ care little about area
" " " performance

- Verilog is complex
  - many bugs





Design process

Think

regs.

Think

state diagrams

block diagrams

timing diagrams

signal names

Think

⋮

Then

- verilog
- test it

* Clear code

Replace state

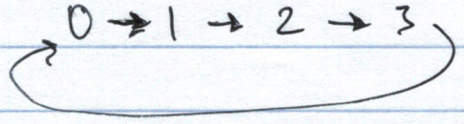
Adding state

I. Counters

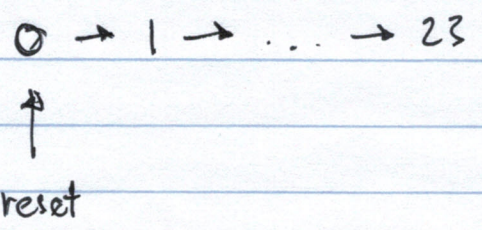
output = state

often increment or decrement

Fixed simple circular counter



- Count Up Counter



Homework 8, 13