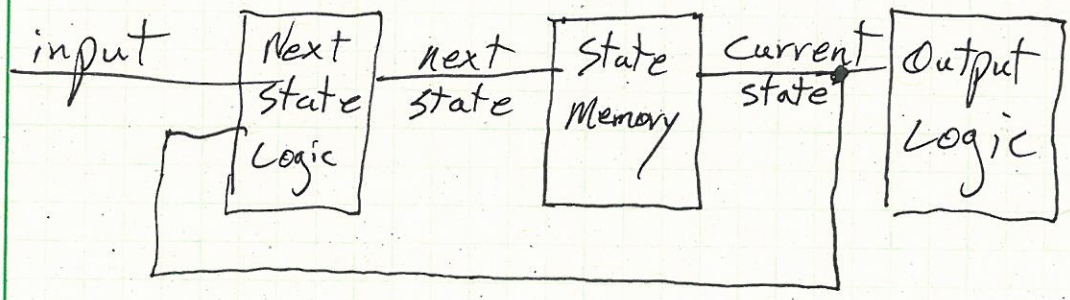
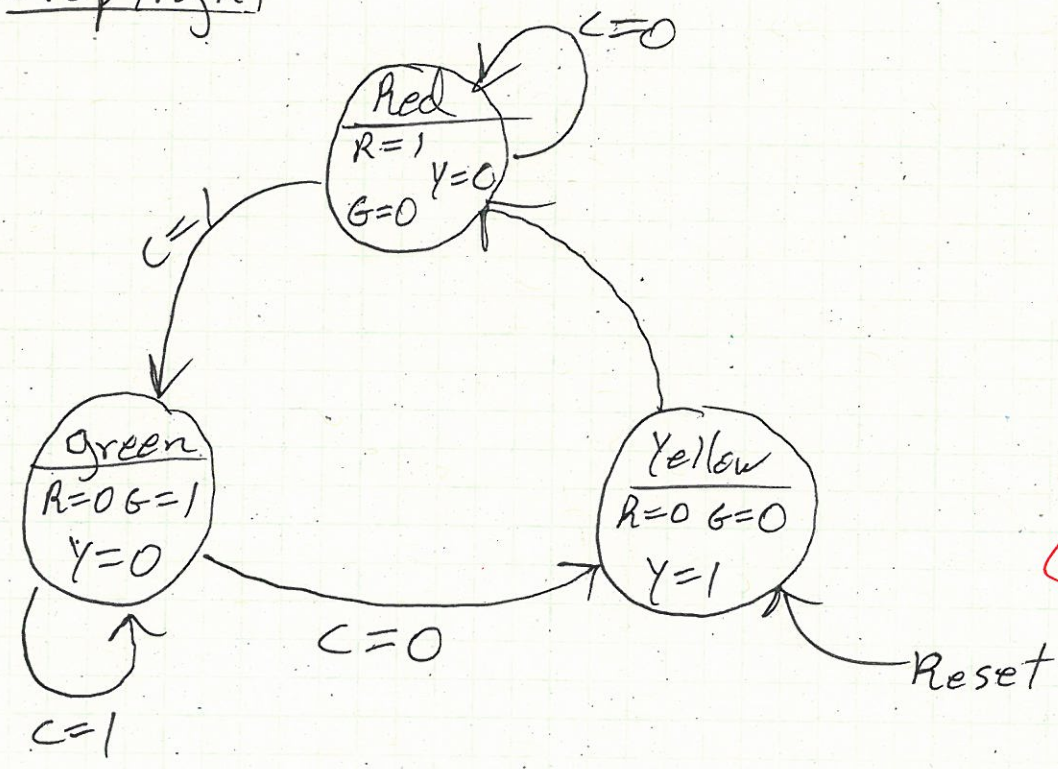
Overview:

- State machines
- testbenches
- VHDL, etc
- HW#3 & Prelab #3 Due LSN22

Parts of a State Machine:



Stoplight



(why Reset to yellow?) - Don't want to go straight to Red

Enumerated data type

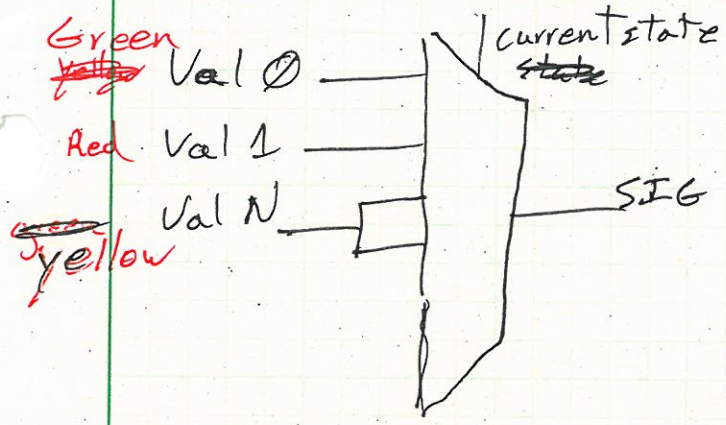
```
type stoplight_state is (red, green, yellow);
signal nextstate, currentstate : stoplight_state;

inside a process:

Case currentstate is
    when red =>
        if c=1 then
            nextstate <= green;
        else
            nextstate <= red;
        end if;
    when green =>
        if c=1 then
            nextstate <= green;
        else
            nextstate <= yellow;
        end if;

    when yellow =>
        nextstate <= Red;
    when others
        nextstate <= yellow;
end case;
```

stop here




```
Case Currentstate is
    when green =>
        SIG <= Val1;
    when red   =>
        SIG <= Val0;
    when others =>
        SIG <= ValN;
end case;
```

Use when others
Whenever you don't want to enumerate all possible conditions.
Also so we don't implicitely create memory we don't want!
- So close all cases

Testbenches:

- assert:
- report:
- severity

3 testbench commands

given:
A <= '0';
B <= '0';
C <= '0';
wait for 10 ns

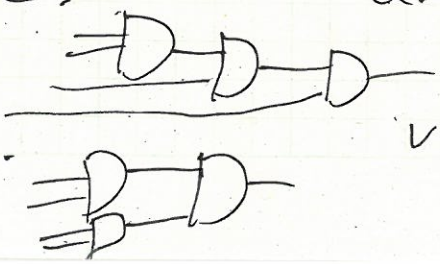
```
assert F='0' report "F=" & std_logic'image(F)
```

↑ expected value
won't report if it doesn't get an error

severity (warning, error, note, failures);

Be careful w/ use: What could be an issue?

- Propogation Delay



Component Declaration

How many of you have investigated the top of your TB?

in architecture →

```
component componentName
    port ( signame : mode sig type );
        :
        );
end component;
```

Down below Begin:

```
Label : componentname portmap (
        port1 => signal1,
        port2 => signal2 ....);
```

Usually uut (unit under test)

Can also do this;
- The quick way!

```
Label : componentname portmap (sig1, sig2, ... );
```

★ Make sure you put #s in Portmap correctly!

Show Wakerly-5-3 VHDC Example in Handouts folder p. 19
- for component Declaration examples.

Create similar VHDL code w/ component Declarations for this;