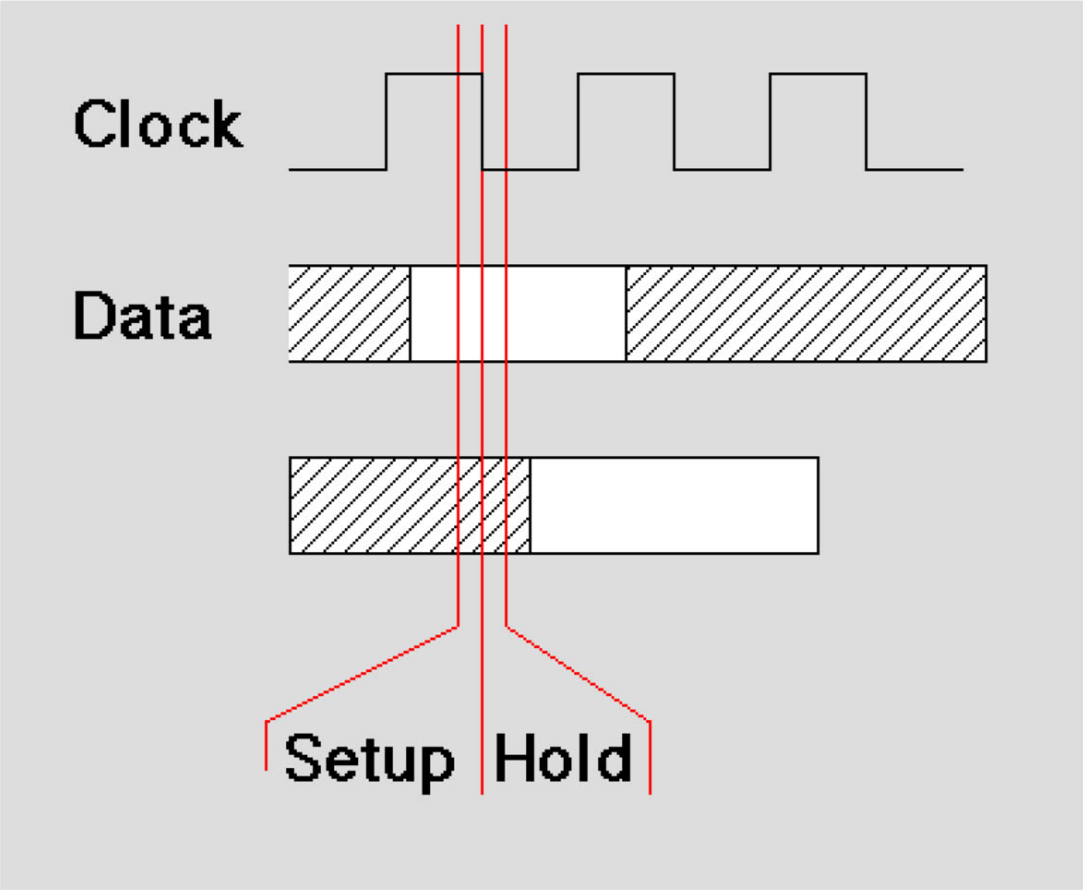
ECE 3060
VLSI and Advanced Digital Design

Sequential VLSI System Design

Setup and Hold Time



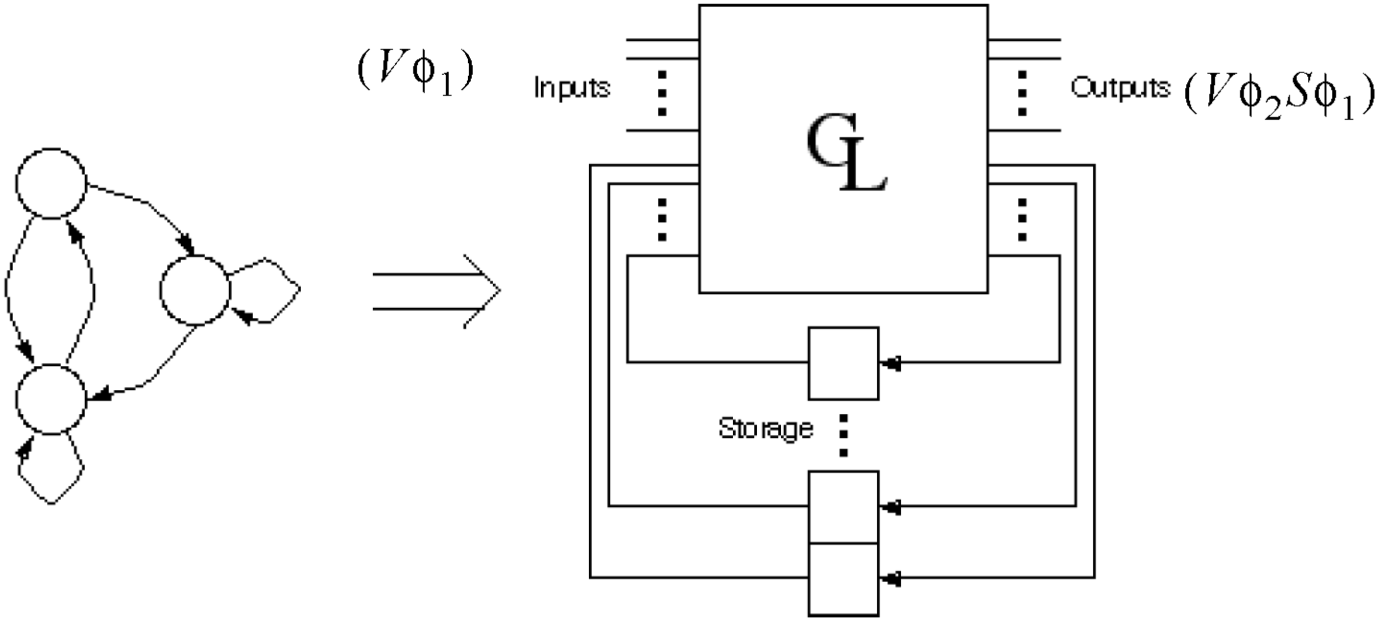
Sequential Control

Functionality

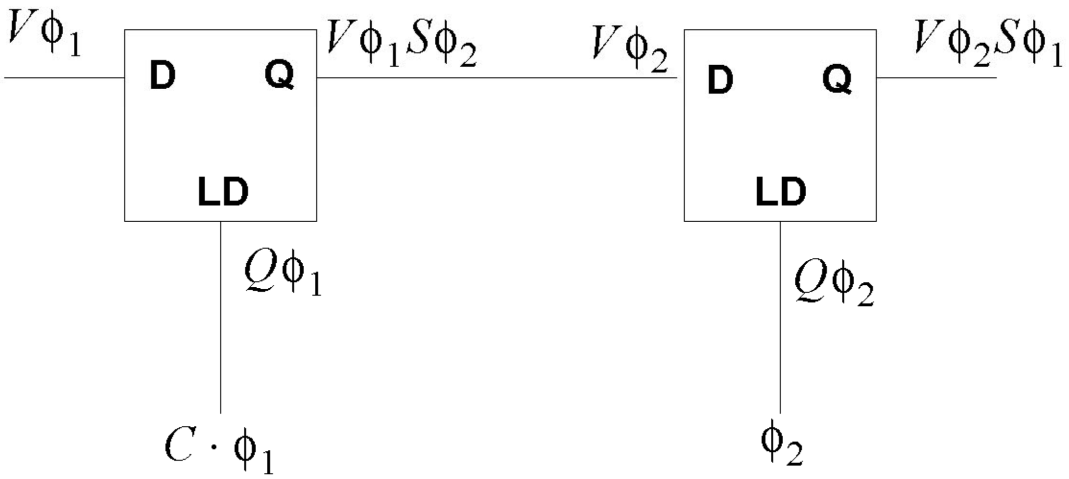
Generates Control Signals to Datapath (Registers and ALU)

Designed as a Finite State Machine

Cycles through States as a Function of Input



Register Clocking



C must be (at least) $S\phi_1$ so that the LD signal on the ϕ_1 latch is $Q\phi_1$. During clock periods where C is high, the register is loaded

Assertions on inputs are the weakest possible

Assertions on outputs are the strongest possible

Asynchronous Inputs

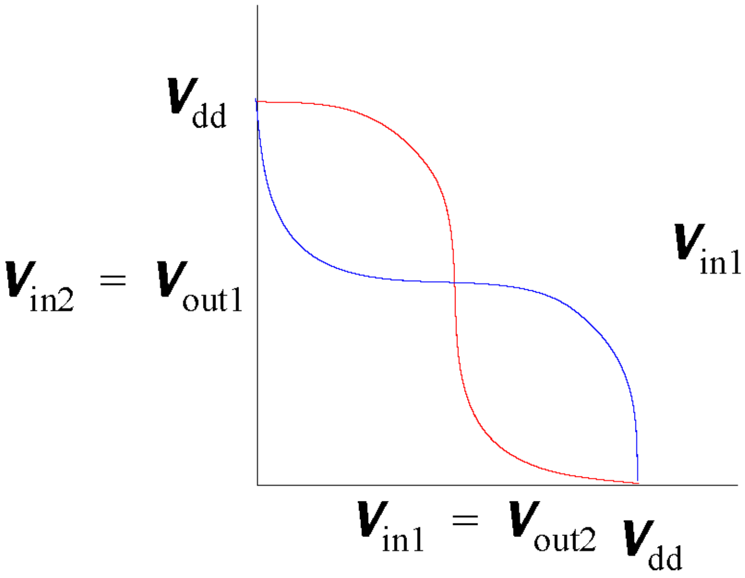
Question: What happens when we cannot make a timing assertion on an input because it is generated asynchronously to our system clock(s)?

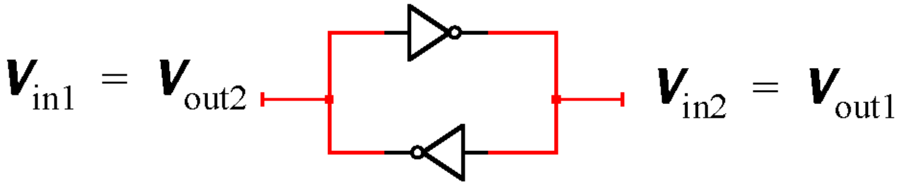
Answer: This can cause unpredictable system behavior.

The reason is a phenomena known as metastability, which we can never completely get rid of, but which we can make very unlikely.

Inverter Loop Characteristic

- Consider the $V_{out}(V_{in})$ characteristic of an inverter loop (latch) cell

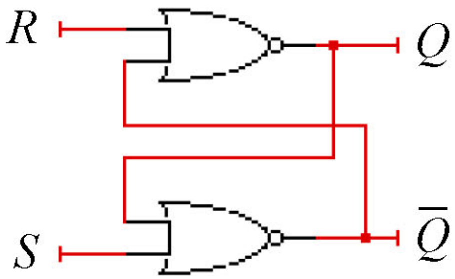




- There are two stable points and one metastable point
- How can we reach the metastable point?

SR Latch Oscillation

Consider the transition on which S and R both go to zero simultaneously



Depending on the delay, the outputs will oscillate

In both cases, the outputs will eventually resolve one way or the other

In both cases, we do not know which way the output resolves, or how long it will take.

If this happens to a state variable, kaflooie.

Synchronization Failure

Within a system, signals can be generated with a known, computable relationship to a clock. If we do the timing analysis carefully, signals will always meet setup and hold times and latches/flip-flops can never become metastable.

The problem is that systems communicate with the outside world. Buttons get pushed.
I/O devices operate on their own clocks. Sensors get tripped.

These external signals are used as inputs to state machines. If they change at precisely the wrong time (i.e. cause a setup/hold time violation) a synchronization failure occurs.