Introduction to CMOS VLSI Design (E158)

Lecture 19: Input/Output Challenges

David Harris

Harvey Mudd College

David_Harris@hmc.edu

Based on EE271 developed by Mark Horowitz, Stanford University

Overview

Reading

W&E 5.6 - I/O Pads

Introduction

So far we have been talking about how to build logic on a chip. But no matter how complex the chip, there will always be a need to get information onto and off of the chip. In many high performance chips, you need to stream large amounts of data onto and off the chip, requiring high-bandwidth I/O. There are many special issues that you need to think about when designing / using I/O drivers, that arise from their ‘unique’ mechanical and electrical constraints. I/O circuitry has to physically connect to some wire on the board, and that means there has to been a geometric scaling of the wire size to make this connection. In addition the I/O needs to protect the internal circuitry from the ‘harsh’ electrical environment on the board. Finally while it might be possible to maintain a good clock environment on the chip, you will need to eventually talk with some systems that run off a different clock. In this situation you will need to synchronize the external signal to your clock.

I/O Plan

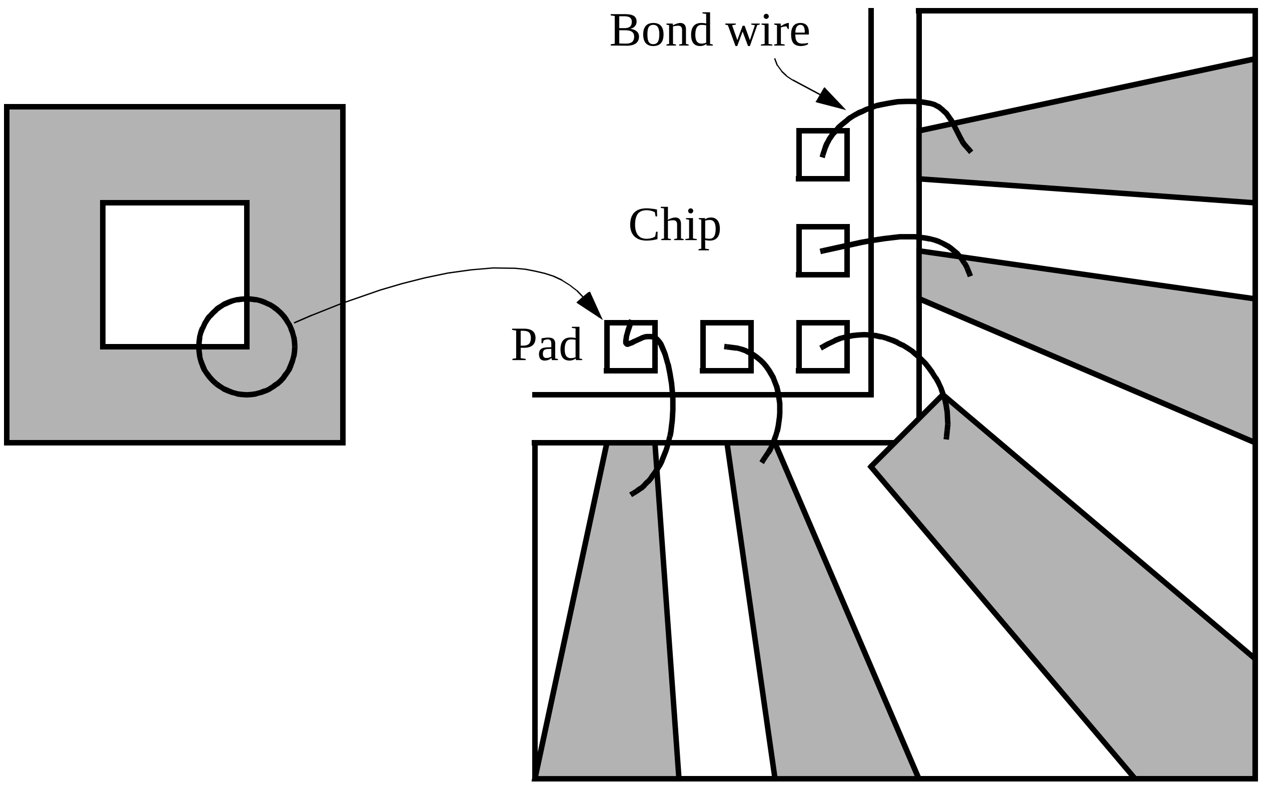
Almost all chips use the same basic method to connect the internal circuits to the outside world.

- Have a (sometimes multiple) ring of large metal squares surround the chip. These squares are called pads.
- Place the chip in a cavity that is surrounded by conductors. This cavity is part of the package of the chip
- Weld small (25μ diameter) wires to the metal pads and the wires in the package to connect them
- Solder the package onto the printed circuit board (PCB)
- This limits I/Os to the perimeter of the chip

    Chip size can be limited by the number of I/O required

    Pads and spacing have to be large (200μ pitch)

I/O Plan



Other I/O Technology

IBM has had a different way to attach chips to packages

- Called C4
- Place pads anywhere[1] on the surface of the die. Then coat them with a special multilayer solder ball.
- Flip the chip upside down (solder balls down) and reflow solder the chip to the package
- Solder is done blind

    Somewhat self aligning. Solder surface tension pulls the chip in place

- Area connection. Can have more pins
- Other companies are doing it too.

1. Well not anywhere. You need to be careful about mechanical stress caused by differential thermal expansion between the silicon and the package

Packages

The bond wires only connect the chips to the package. Still need to connect the packages to the board. Still have a large number of pins.

Two approaches

- PGA - Pin grid arrays moving to BGA (ball grid array)

  Create a 2-D array of pins under the package

  .1” centers (some packages have 0.05” offset centers)

  Use through-hole in the board to connect package to board

  State of the Art 600 pins for PGA, 1000 pins for BGAs

- Quad Flat Packs

  Fine lead pitch (.02 -.03” pitch)

  Surface mount technology

  Pins on 4 edges of the package

Problem with I/O

- High Capacitance

  Pads are large, and pins are larger. Since capacitance is proportional to size, the capacitance loading of the pins is large. Once the pin is connected to a board trace, the capacitance grows even larger. The standard board load chips are tested with is 50pF.

- Unknown voltages

  On chip we are sure that the voltages on the wires will be between the power supply rails. But on the pins of the chip you can't make that guarantee. You an not sure what the input signals will look like. They might be from a TTL gate and go from 0 - 3V, or they might be from a ringing transmission line and go below ground or above VDD

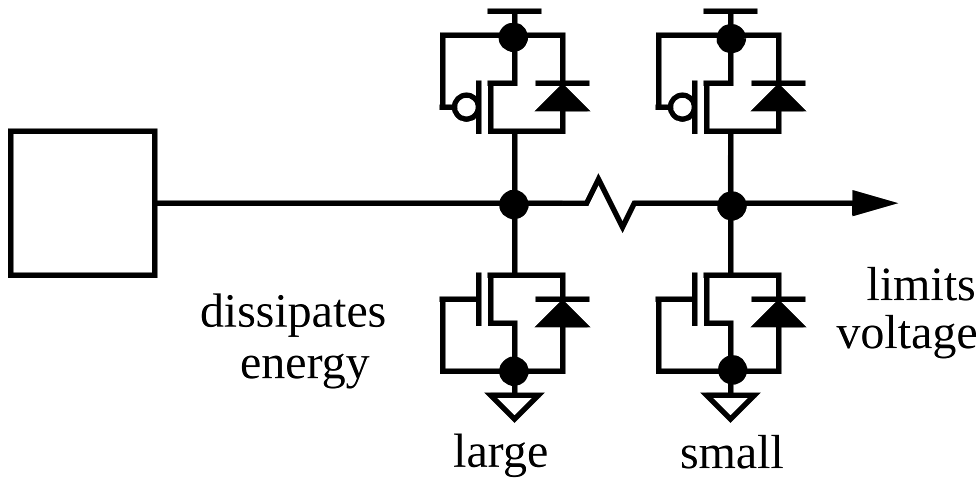
- Static Electricity

  Need to protect the inside circuitry from accidental high-voltage discharges

Input Protection

There is a lot of energy in a static discharge. You need to have large area structures to dissipate this energy without blowing up the wires or the protection device.

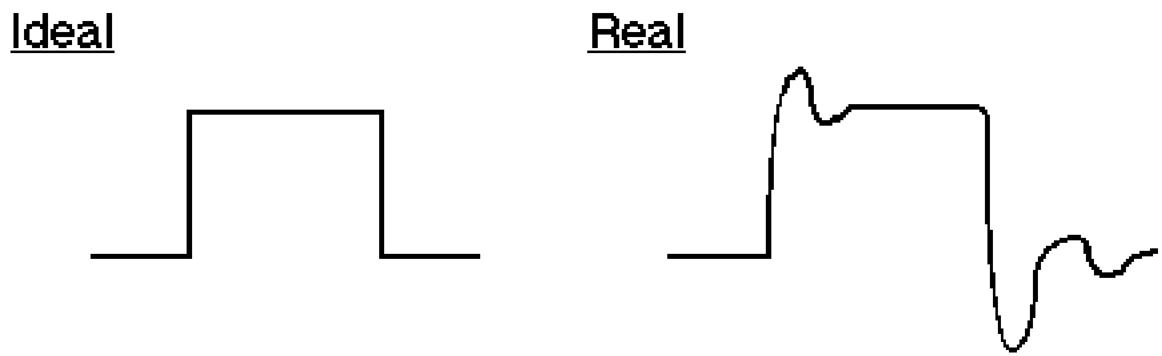
Circuit is pretty simple:



The diodes are the source/substrate diodes of the transistors. Near the pads you expect to have substrate currents. Need lots of well and substrate contacts. Even use guard rings - surround well with contact.

Input Voltages

Waveforms on boards are not ideal

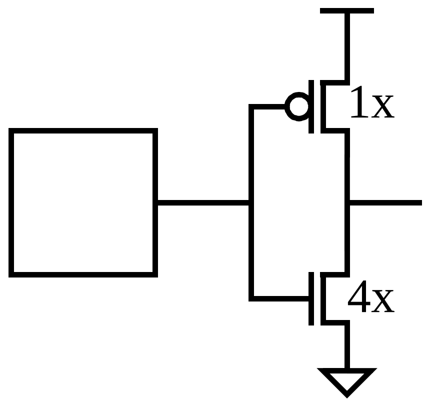


Also levels are not rail to rail

- T^2L high level is only 2.4V. You need to make sure you input cell senses this level as a high. This is a pretty low switch point for a CMOS gate.
- Need to have a special gate on the pad to generate a nice level for the rest of the chip.
- With scaling chip voltages, the input can be at a higher voltage than the power supply (it comes from an older part). Need to not do anything bad. This is becoming the main issue for tri-state outputs.

Input Buffers

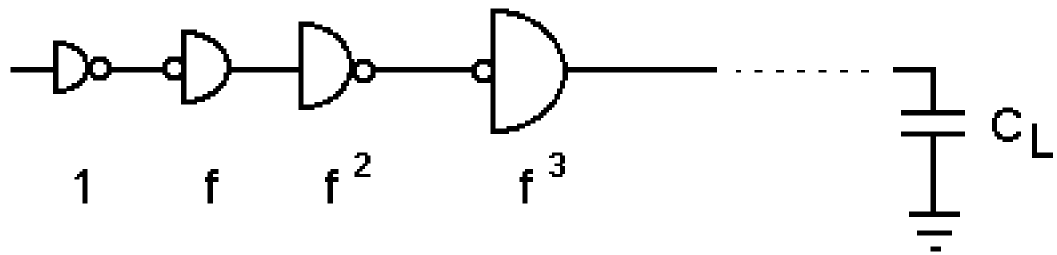
Basically the input is an inverter with a switch point set to the at the correct level. The switch point depends on the power supply and the ratio of the devices. The devices must be carefully sized to match the input voltage specification. A switching point below Vdd/2 is accomplished by making the nMOS much larger than the pMOS device.



Since the gate dimensions are weird, and because the input levels might not turn the transistor fully on or off, this inverter is usually small. This means to drive a long wire to the core of the chip will take some level of buffering.

Output

In nMOS this was hard, since the output needs a large inverter to drive the large capacitance load. In CMOS this is easier, you just use a large ramp up chain.



Use as many stages as you need. Might need most of the stages to drive the wire to the pad.

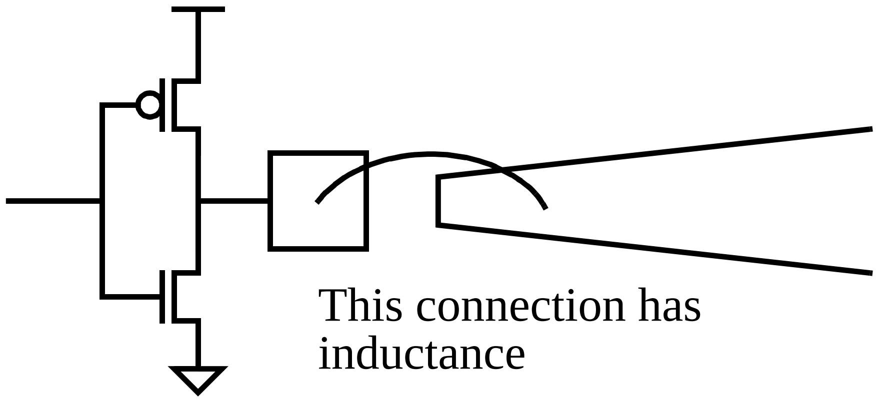
Caution:

> Pads don’t change much as technology scales. As the transistors get smaller the internal capacitances all scale down. But the pad loading stays about the same. Need more stages as technology scales.

Power and Ground Bounce

Just making the inverters the right size to drive the output capacitance is not enough. There is another issue that needs to be addressed, and that is power supply noise.

This noise is caused by inductance in the package

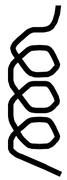


Bond wire is ~ 1nH/mm (2mm long)

Package is ~ 3-10nH

Inductors?

Inductor is a device that stores energy in a magnetic field:

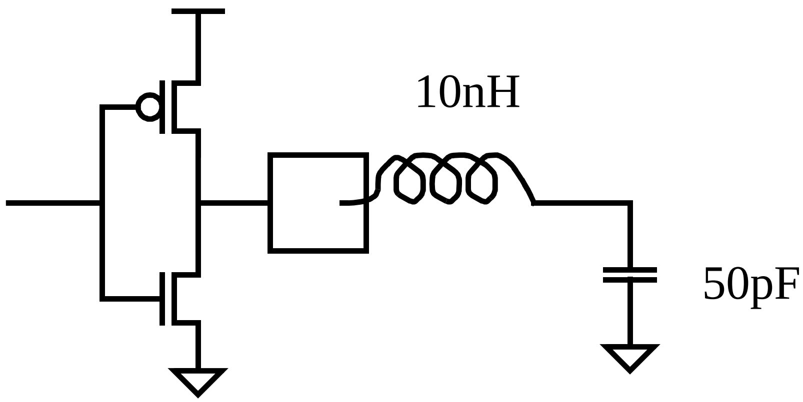


$$V = L \frac{di}{dt}$$

Inductors try to keep the current constant.

Change their voltage when the current changes

- Circuit we need to study



Output Drive

For a 2.5V swing in 2.5ns

- Output must swing 1V/ns

  Current = CdV/dt = 50mA
- Current is turned on in about 2.5ns

  dI/dt = 50mA/2.5ns = 20mA/ns
- Voltage drop across the inductor

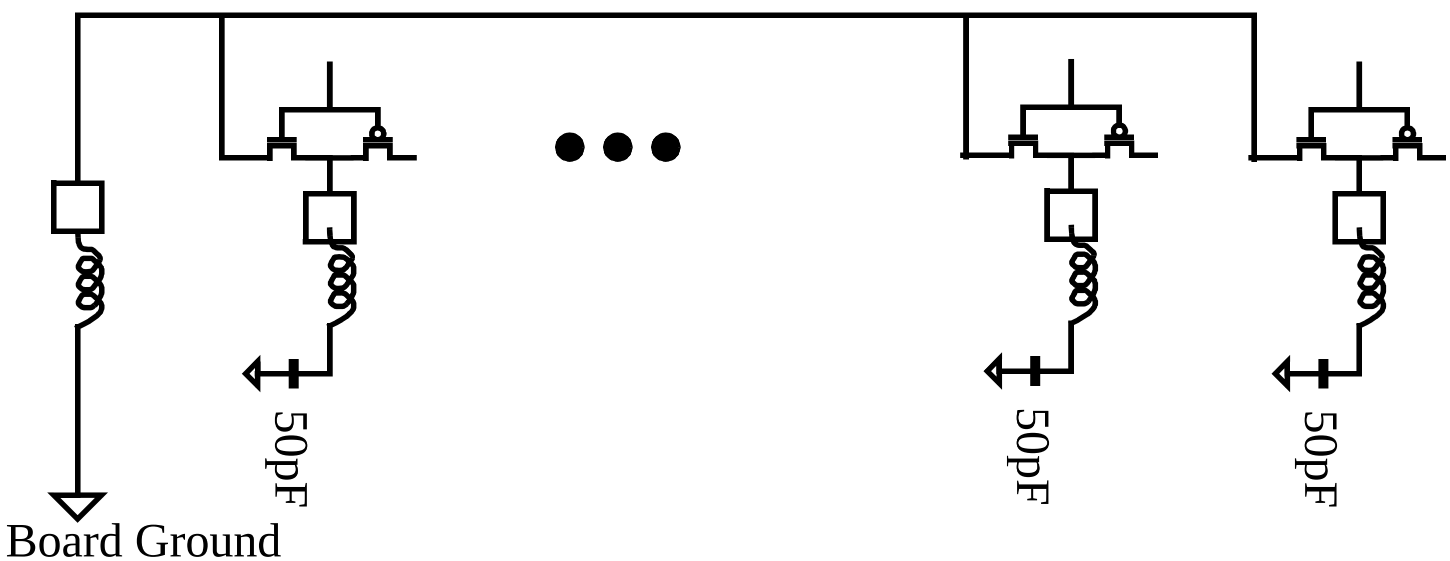
  10nH 20mA/ns = 0.2V

  This is not a large effect compared to the 2.5V supply

What is the fuss?

Power and Ground Pins

We were looking at the wrong place. The power supplies have to come on to the chip through pins and bounds leads too.There are usually fewer power and ground pins then there are output pins. That means that many output currents will have to flow through the supply line.[1]

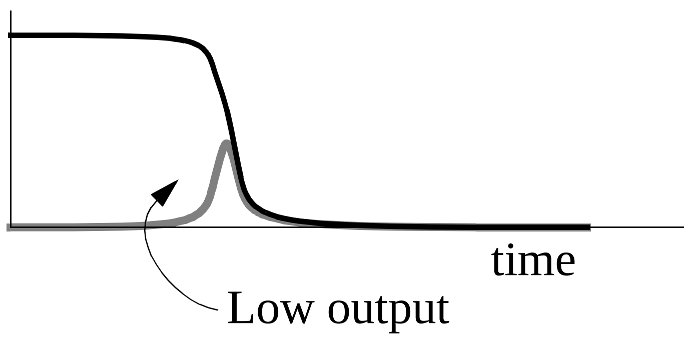


1. In the old TTL MSI packages that many components still use, there is only a single VDD and Gnd pin, and these pins are the highest inductance pins in the package. That is why this problem was first seem in the octal buffer parts.

Simultaneous Switching Noise

The inductance in the power supply lines causes a problem referred to as simultaneous switching noise. If all the outputs switch then a large di/dt will flow through the ground pad inductance. This will cause the voltage across the inductor to increase, which will raise the local Gnd level temporarily. If all the outputs are switching this is not a problem, since the output are in the middle of a transition when the noise occurs.

If however one output was already low when all the outputs change, then the output voltage on the low pad will reflect the value of Gnd on the chip, and this output will have a spike on it during the transition.



Switching Noise

This is a hard problem to fix, since the inductance is set by physics.

- Current generation packages have planes in them to reduce the inductance.
- Current chips have many power and gnd pins. In a 400pin package over 100 pins are power and ground.
- Need a power and ground pair for each 4-8 outputs.
- Also need to worry about internal circuits too

  Clock driver in Alpha (21064) has peak current of over 20A, and an edge rate of around 100A/ns. To do this they have on chip bypass capacitors

Timing Issues

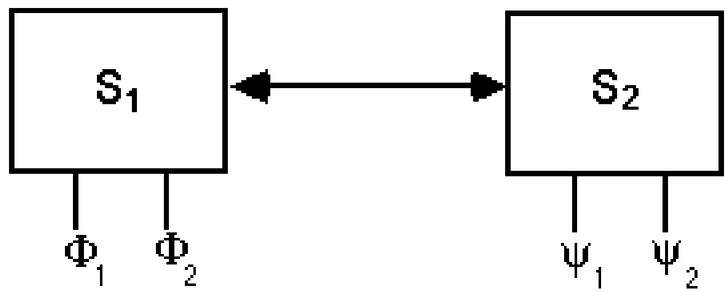
So far we have been talking about issues in sensing and driving signals on and off chip. But that is only part of the interface issue.

The other part is knowing when to when to look at the signals. Within a chip we use the clocking method to make sure that signals are stable when we latch them. What do we do with inputs and outputs?

- Force the whole world[1] to use your clock. This makes the entire system synchronous and the problem is solved
- Use an asynchronous timing discipline
- Use synchronizer

1. Or at least the world that you communicate with

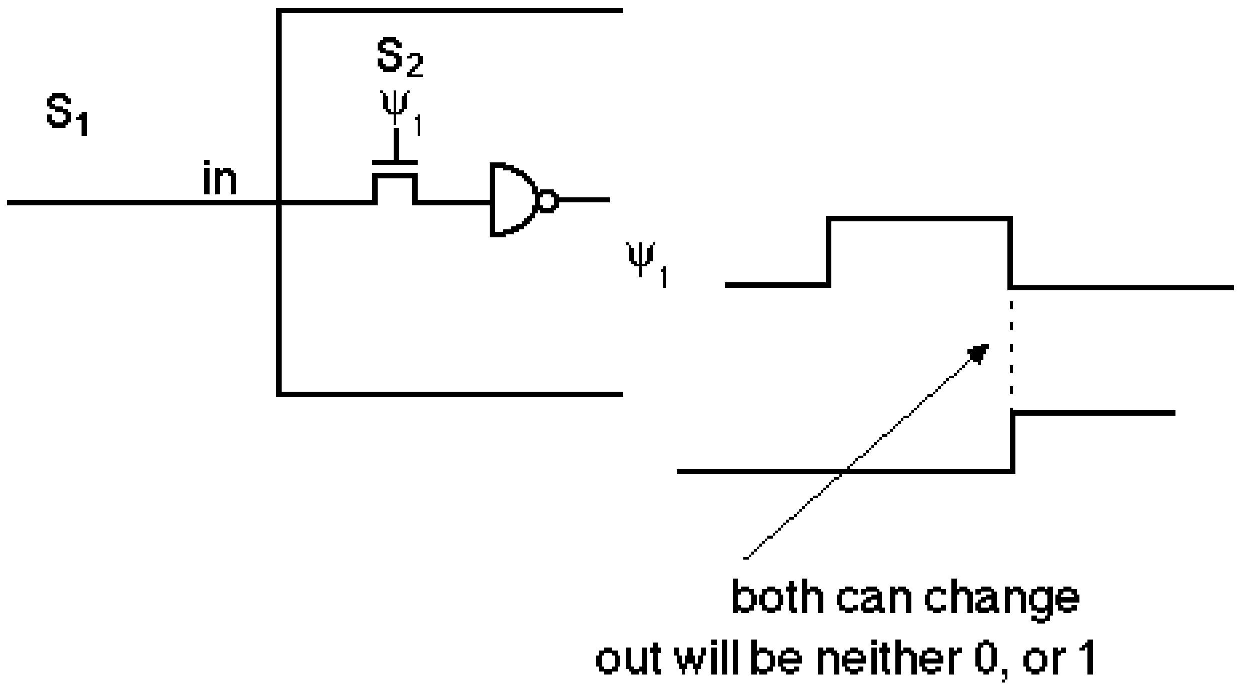
Synchronization



Within a box we can guarantee arrival times to a latch. They are determined from a known clock edge. $S_1 \leftrightarrow S_2$ is a problem

When S_2 tries to read an input, the input can change. Ψ1 and Φ1 are not related in time.

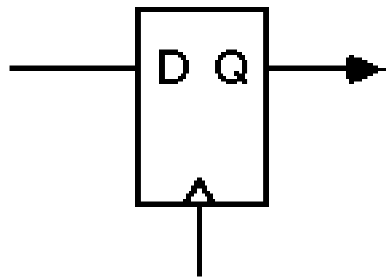
Latch Problem



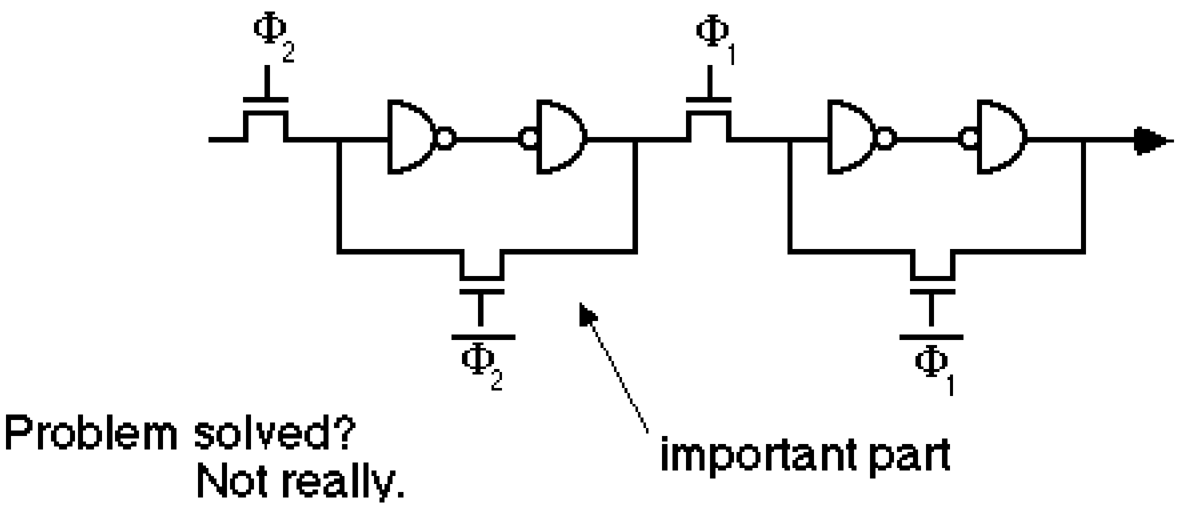
A simple dynamic latch could get an output that would be between 0 and 1.

Using Feedback

Use positive feedback on the latch. Use a static latch. That way if it is a little closer to one, the feedback will drive it to one, and if it is closer to zero, the feedback will drive it to zero.



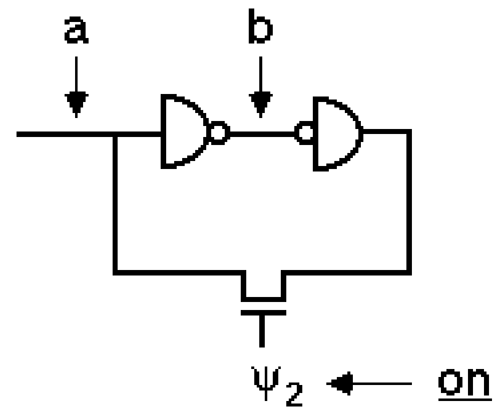
Note that is all these pictures the nMOS transistors should be full CMOS transmission gates. I just got lazy in drawing the figures



Decision Circuit

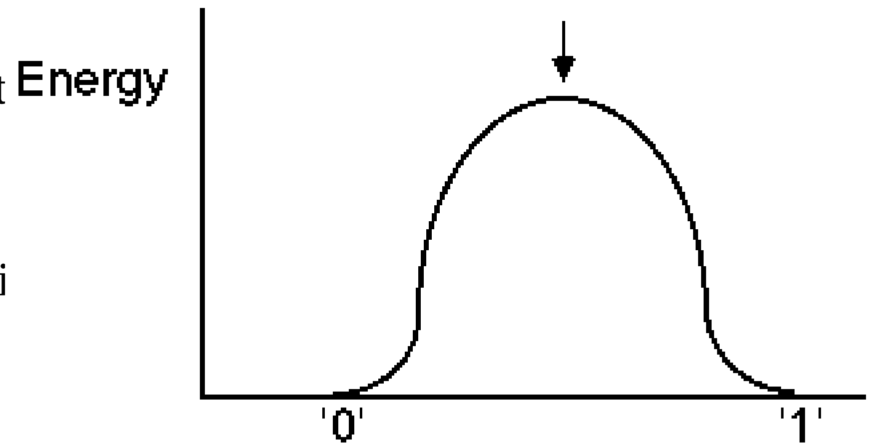
Basic Circuit

Problem is that the circuit is stable if Va = Vb. If the voltages are slightly different then they will resolve to good values



Energy Diagram

You will eventually fall off, but the closer you start to the middle, the longer it will take you to settle. The circuit is metastable when it is not 0, 1.



If the logic looks at the output of the decision circui and it has not settled it is called a synchronization failure.

Failure Rate

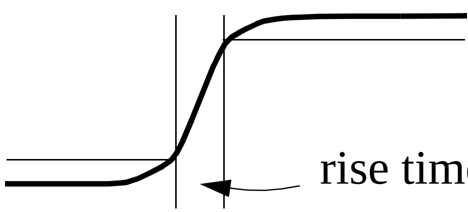
How many failures?

$$R = \frac{\text{failures}}{\text{sec}} = \frac{\text{metastab}}{\text{sec}} \cdot \Pr\{\text{failure} \mid \text{meta}\}$$

$$\frac{\text{metastab}}{\text{sec}} = M$$

$$M = \frac{\text{async trans}}{\text{sec}} \cdot \text{Fraction of time vulnerable}$$

$$\text{Fraction of time vulnerable} = \frac{tr}{tcycle}$$



rise time, time when signal is not 0,1

Failure Probability

While the circuit is metastable, the output voltage is growing exponentially:

$$Vout = (Vin - V_s)e^{\frac{t}{t_d}} + V_s$$

If it is not settled after time t, then the input must have been very close to V_s. How close depend on t, the time you have waited:

$$-V_s e^{-\frac{t}{t_d}} < Vin - V_s < (V_{dd} - V_s)e^{-\frac{t}{t_d}}$$

If we assume that the input was a ramp, then Vin should be uniform probability in the range from 0 to Vdd (all that really matters in the region around Vs). Then the probability that the output is not settled is just the size ratio of 0,Vdd to the equation, which is exp(-t/td).

A Few Numbers

The number of failures will be:

$$R = \text{Events} \times \frac{t_{rise} + t_{fall}}{t_{cycle}} \times e^{-\frac{t}{t_d}}$$

Events	10^8/sec
t_{rise}+ t_{fall}	2ns
t_{cycle}	20ns
t	10ns
t_d	0.2ns

$$10^8 \times 0.1 \times e^{-50} = 2\times10^{-15} = 6\times10^{-8}\text{years}$$

If you wait 1/2 as long, the error rate increases by $7x10^{10}$