

## Review of Multiplexors

A multiplexor takes in  $n$  select lines, and  $2^n$  inputs, and based on the value of the select lines, the multiplexor selects which input  $D_i$  to output.

## 5.1 RAM

32 bits of memory isn't enough, so that's why we have *Random Access Memory* (RAM). Static RAM (SRAM) uses D latches to store data. This memory is very slow compared to CPU speed; it's also not clocked.

### 5.1.1 Three-state Buffers

Has three outputs: 0, 1, and *floating* (which means neither connected to power or ground). Uses a control (which can either be 1 or 0) to connect or disconnect data flow. When connected, the data is either 0 or 1 (obviously) and when it's disconnected, the data is floating (which means that there is not connection)

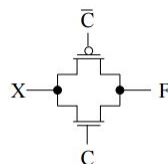


Figure 5.1: The mechanics of a three-state buffer. If control ( $C$ ) is 1,  $F$  copies  $X$  (i.e., there is a connection); otherwise,  $F$  is floating (i.e., there is no connection). Courtesy of Prof. Mann's slides.

On notes and assignments, a three-state buffer is drawn as

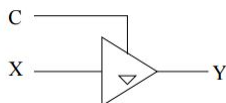


Figure 5.2:  $C$  defines the control, and  $X$  and  $F$  are the input and output respectively. Courtesy of Prof. Mann's slides.

**Example 5.1.1.** *An XOR from three-state buffers:*

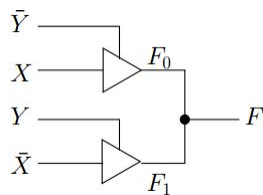


Figure 5.3: Courtesy of Prof. Mann’s slides.

If we fill in the truth table for this diagram we have

$X$	$Y$	$\bar{X}$	$\bar{Y}$	$F_0$	$F_1$	$F$
0	0	1	1	0	—	0
0	0	1	0	—	1	1
0	0	0	1	1	—	1
0	0	0	0	—	0	0

With three-state buffers, if we have multiple lines leading to one output (such as the previous figure, where  $F_0$  and  $F_1$  both lead to  $F$ ), you must ensure that at most one select input is 1, or a short-circuit may result.

## 5.2 Implementing a Traffic Light Controller Circuit

blah blah variables. Implementing the controller with a finite-state controller, the truth tables look like: From the graphical example. This flow determines where we end up based on our input stream.