Digital Systems

Physics 5430

Chapter 3 Part 1 – Busses, Multiplexing ASM Inputs

Bus Notation

TYPO: ROM has MSB of address and data pins at the top. **Always put LSB at top.**

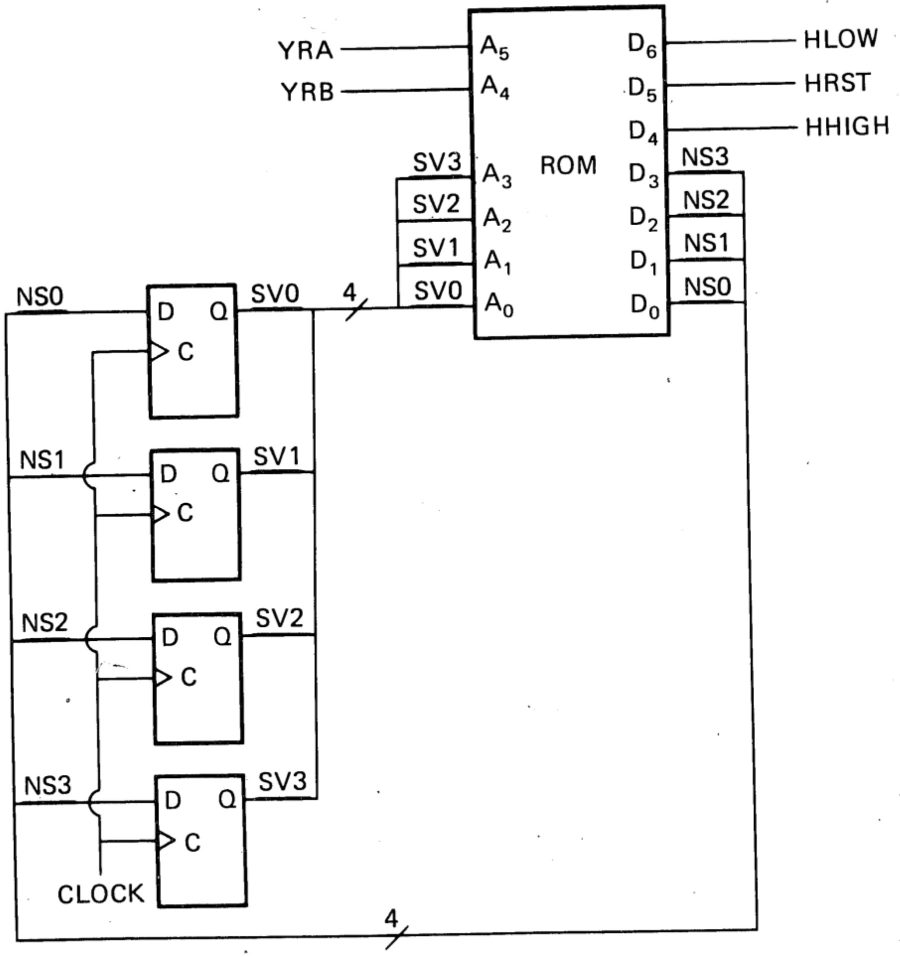
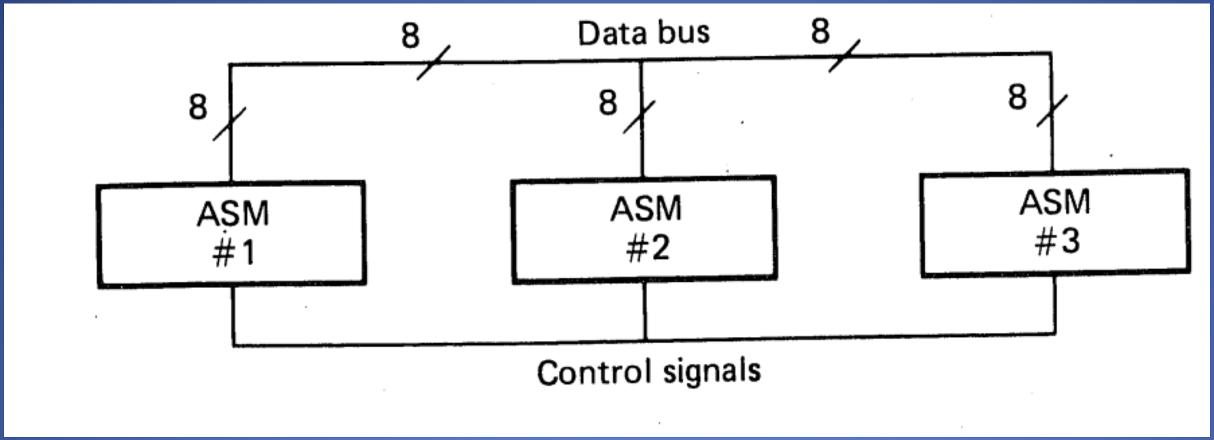


Figure 3-1 Notation for bus.

Showing all the signal lines clutters the schematic.
Instead, use a single line for bus, split out the inputs and outputs at the ends of the bus. Label bus with number of signal wires, e.g., use /4.

Bi-directional Data Bus

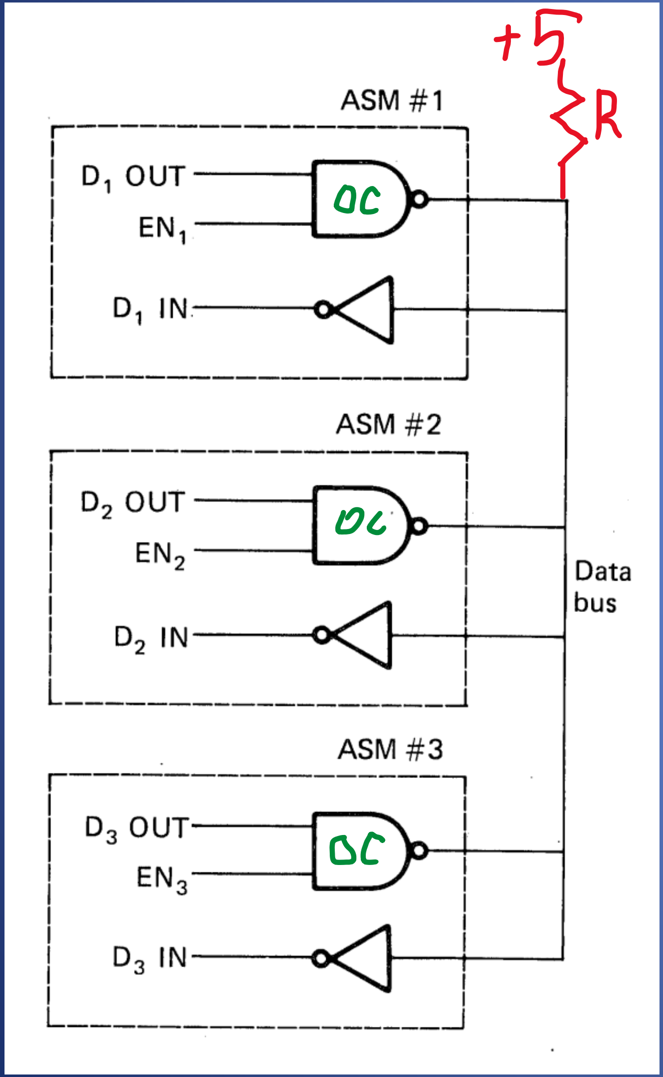


Data is passed between 3 ASMs.
Control signals specify which ASM is the transmitter.
All three ASMs are simultaneous receivers.

Two popular methods are:

1) Wired-AND (used in the IEEE-488 GPIB bus and the I^{2}C bus)
2) Tristates

Wired-And Data Bus



All inputs (e.g., D1 In) are connected to the bus through inverter/buffers.

Only one output is gated onto the bus at a time. An enable signal (e.g., EN1) Is applied to the OC NAND gate.

Wired-AND Data Bus (continued)

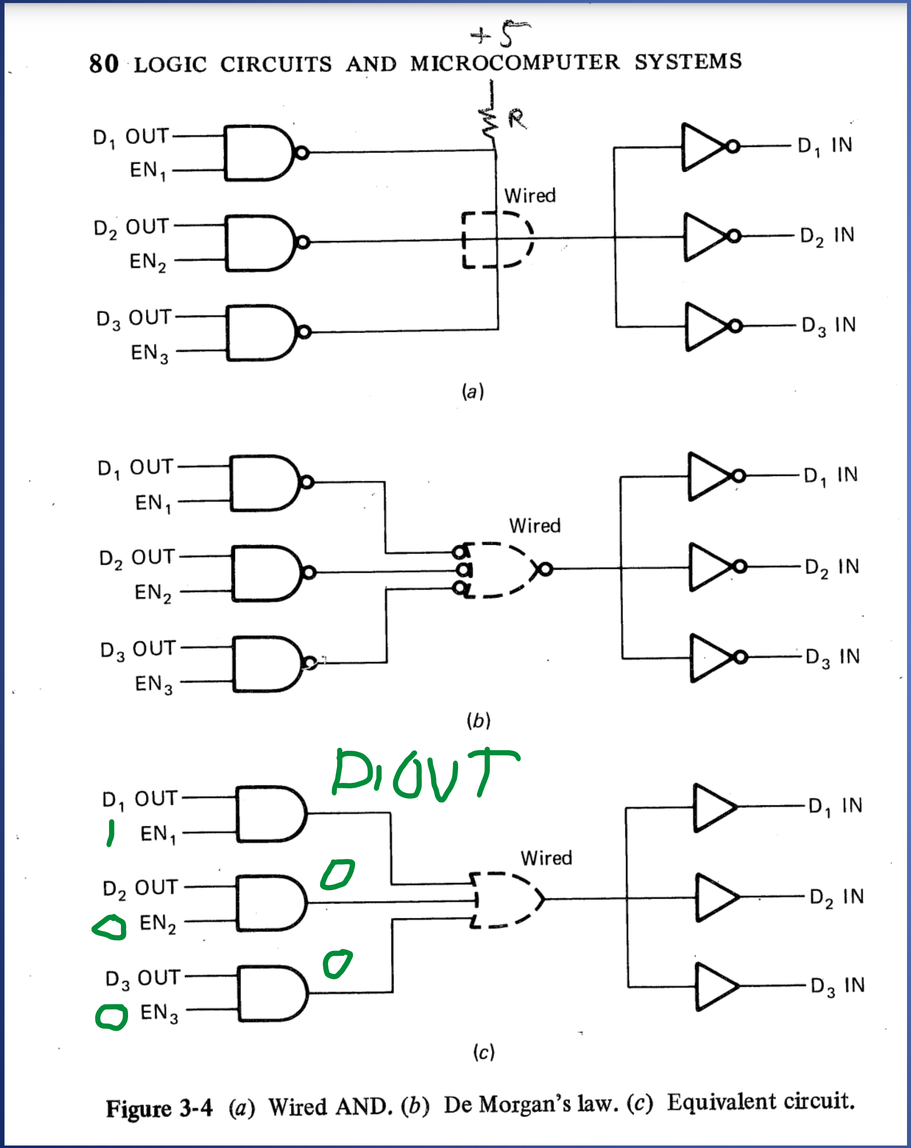


Figure 3-4 (*a*) Wired AND. (*b*) De Morgan's law. (*c*) Equivalent circuit.

Redraw Circuit with Inputs and Outputs grouped

DeMorgan's Law

Equivalent Circuit

Wired-AND Data Bus (continued)

Wired-AND data bus ends up with transmitters Ored to receivers. Each transmitter can put its data on the bus if it is enabled.

Disadvantage is slower propagation of signal through the gates.

Tristate Bus

Faster signal propagation than Wired-AND bus.

Tristate Inverter

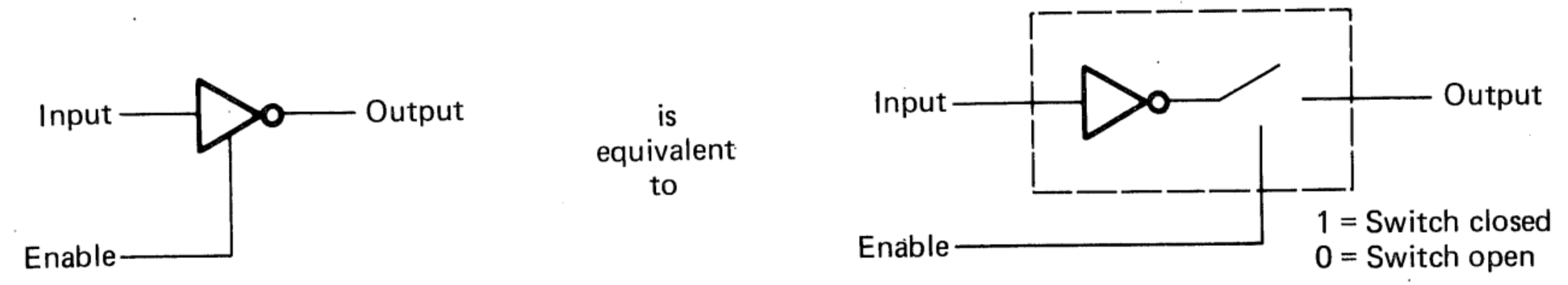
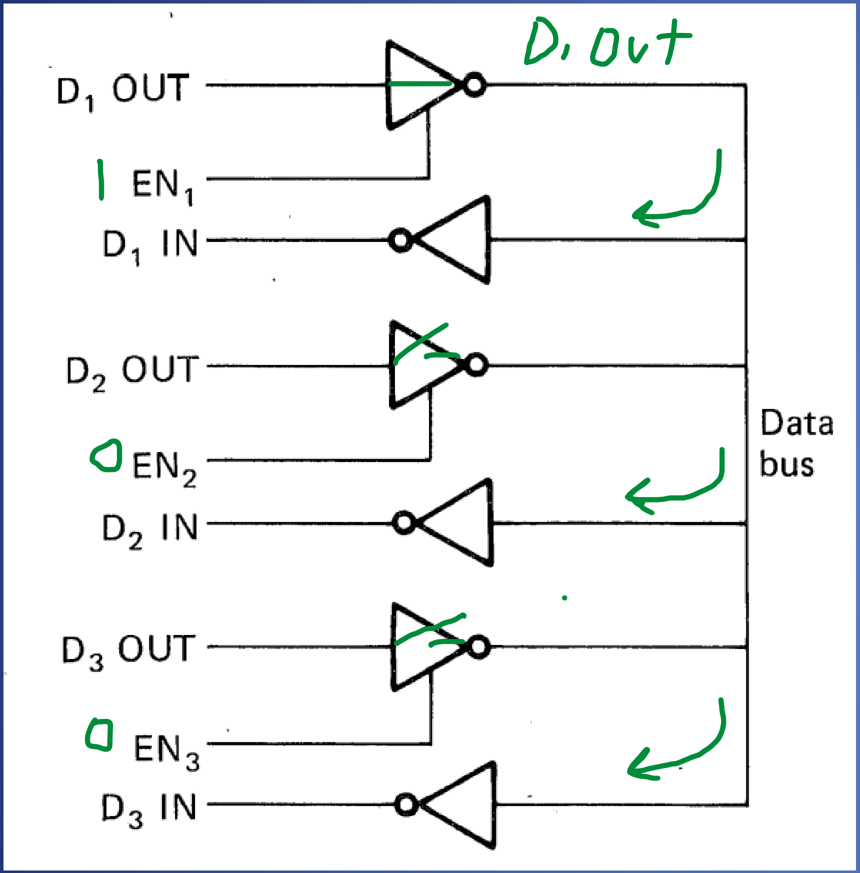


Figure 3-5 Inverter with three-state output.

Tristate Bus



With Enables set to (1,0,0), D1 OUT is connected to the bus.
D2 OUT and D3 OUT are disconnected from the bus.

Note: This circuit of inverters provides the same functionality as the previous Wired- AND bus.

Scale of Integrated Circuits

Standard circuits and algorithms are used repeatedly in digital electronics. These are MASS PRODUCED which (1)Lowers production costs, and (2)Lowers DESIGN costs, since the cost is shared by millions of IC's.

Chips are categorized according to complexity (the number of gates per chip).

SSI (Small Scale Integration) 1-20 gates per chip.

MSI (Medium Scale Integration) 20-100 gates/chip.

LSI (Large Scale Integration) 100-1000 gates/chip.

VLSI (Very Large Scale Integration) >1000 gates/chip.

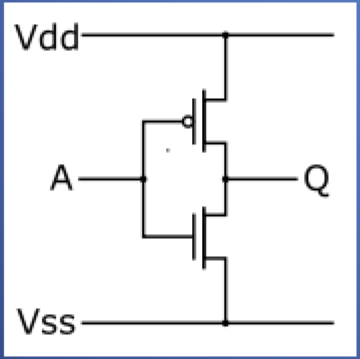
Generally, TTL is used for SSI and MSI. CMOS is used for LSI and VLSI. CMOS is simpler, with a smaller area, and low power.

CMOS Circuits

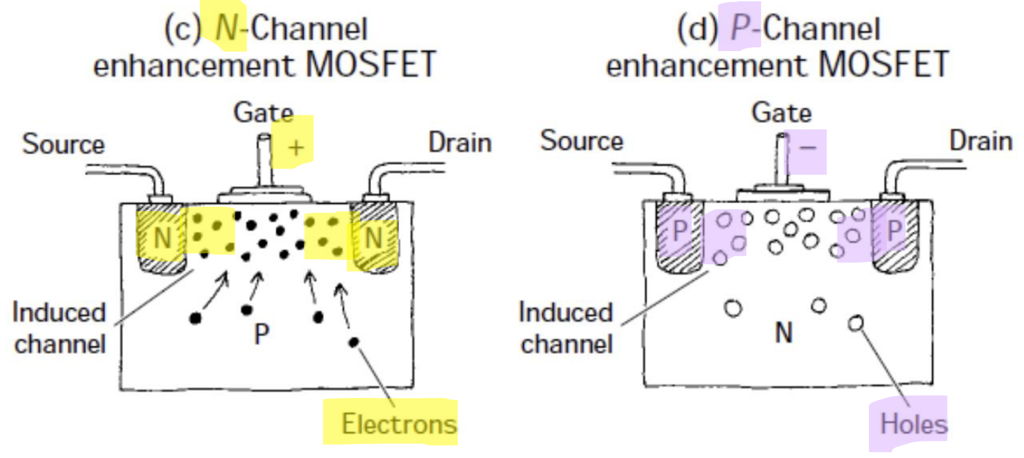
Complementary Metal Oxide Semiconductor

“Complementary” because the transistors come in n-type and p-type pairs.

What does the following CMOS circuit do?



8-7 MOS Technology

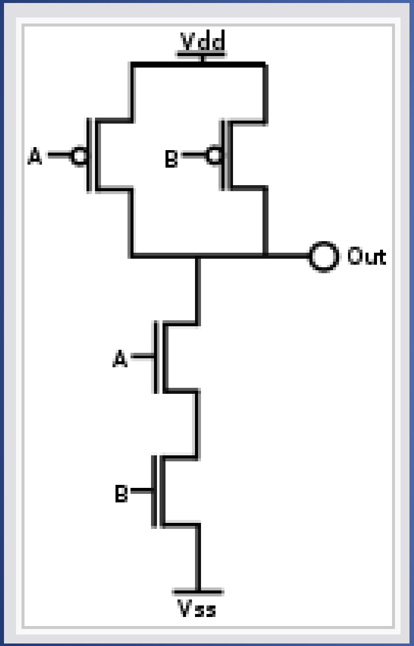


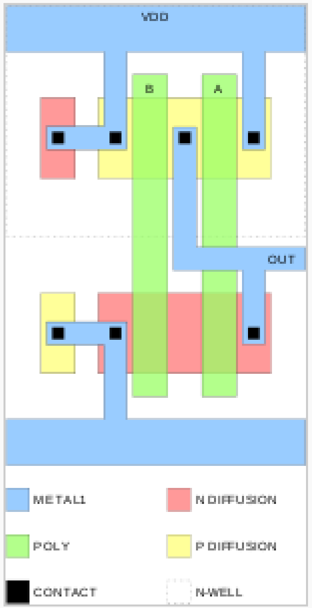
Practical Electronics for Inventors 2nd ed. page 169

FIGURE 4.64

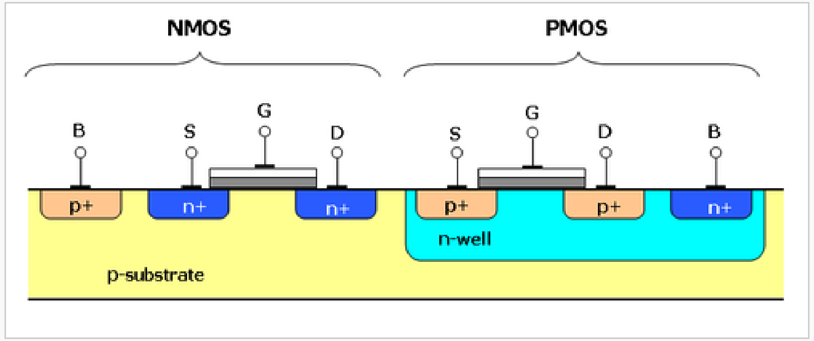
Enhancement MOSFETs, unlike depletion MOSFETs, have a normally resistive channel; there are few charge carriers within it. If a positive gate-source voltage is applied to an *n*-channel enhancement-type MOSFET, electrons within the *p*-type semiconductor region migrate into the channel and thereby increase the conductance of the channel (see Fig. 4.64*c*). For a *p*-channel enhancement MOSFET, a negative gate-source voltage draws holes into the channel to increase the conductivity (see Fig. 4.64*d*).

CMOS NAND Gate



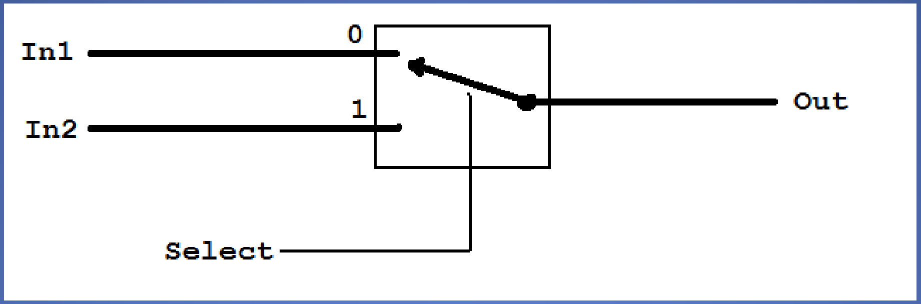


The physical layout of a NAND circuit

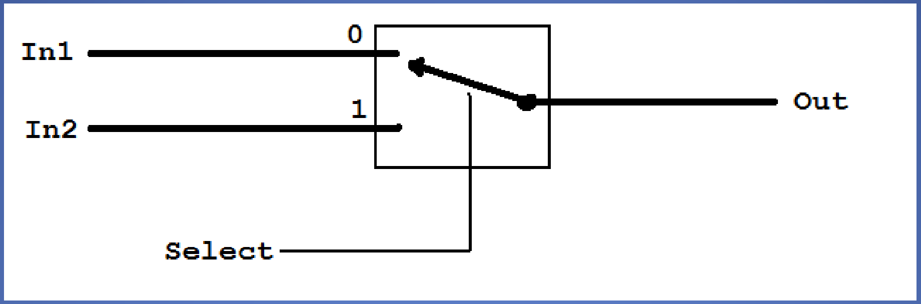


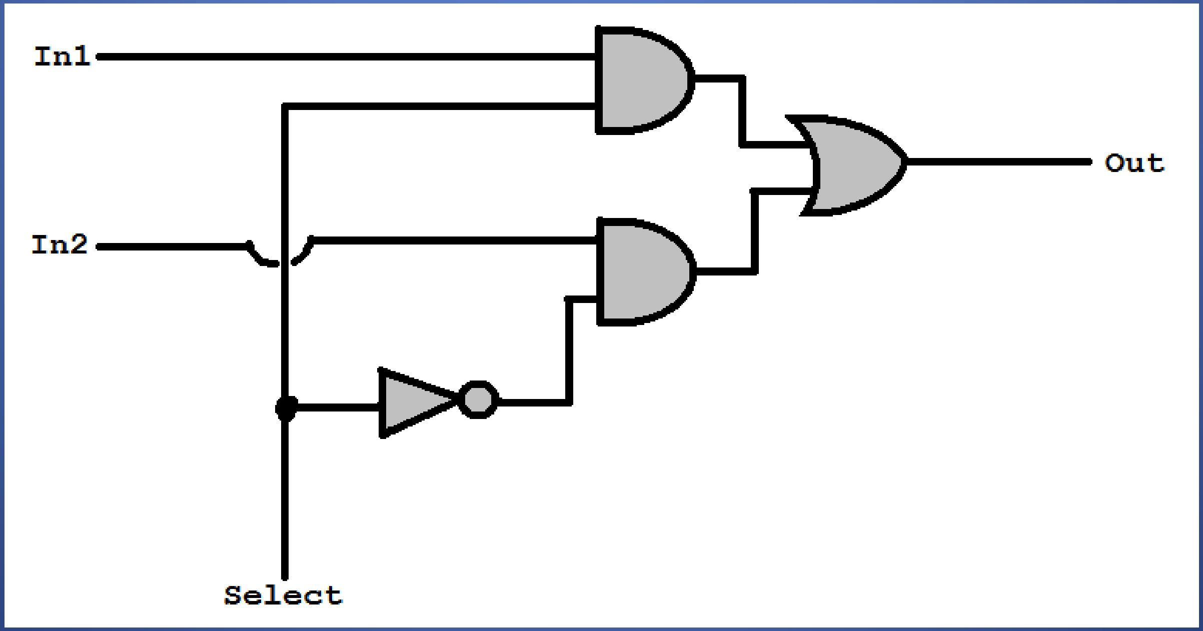
Cross section of two transistor in a CMOS gate, in an N-well CMOS process

A Simple Multiplexer

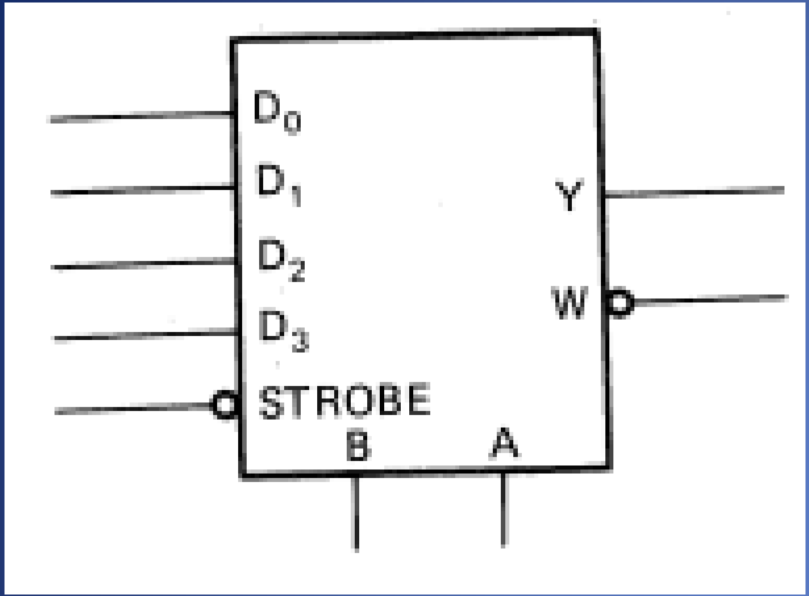


A Simple Multiplexer





4 to 1Multiplexer



Select lines BA connect 1 of the 4 inputs to output Y.   B is MSB.

W is Y bar.

Strobe is active-low enable.
Y = 0 when disabled.

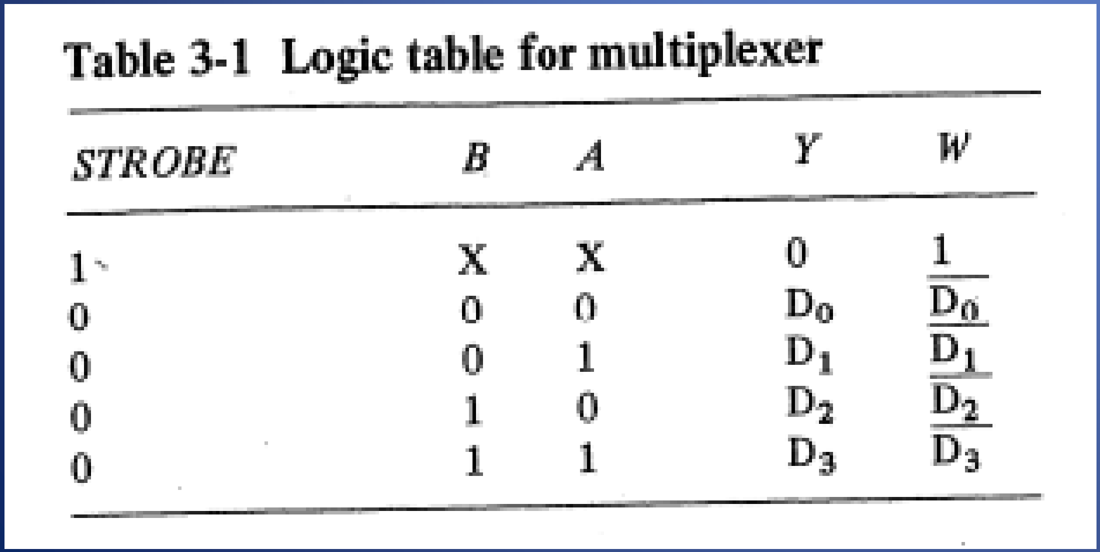


Table 3-1 Logic table for multiplexer

STROBE	*B*	*A*	*Y*	*W*
1	X	X	0	1
0	0	0	D_0	$\overline{D_0}$
0	0	1	D_1	$\overline{D_1}$
0	1	0	D_2	$\overline{D_2}$
0	1	1	D_3	$\overline{D_3}$

A = 0, B = 0 selects D0

A = 0, B = 1 selects D1

B = 1, A = 0 selects D2

B = 1, A = 1 selects D3

4 to 1Multiplexer Internals

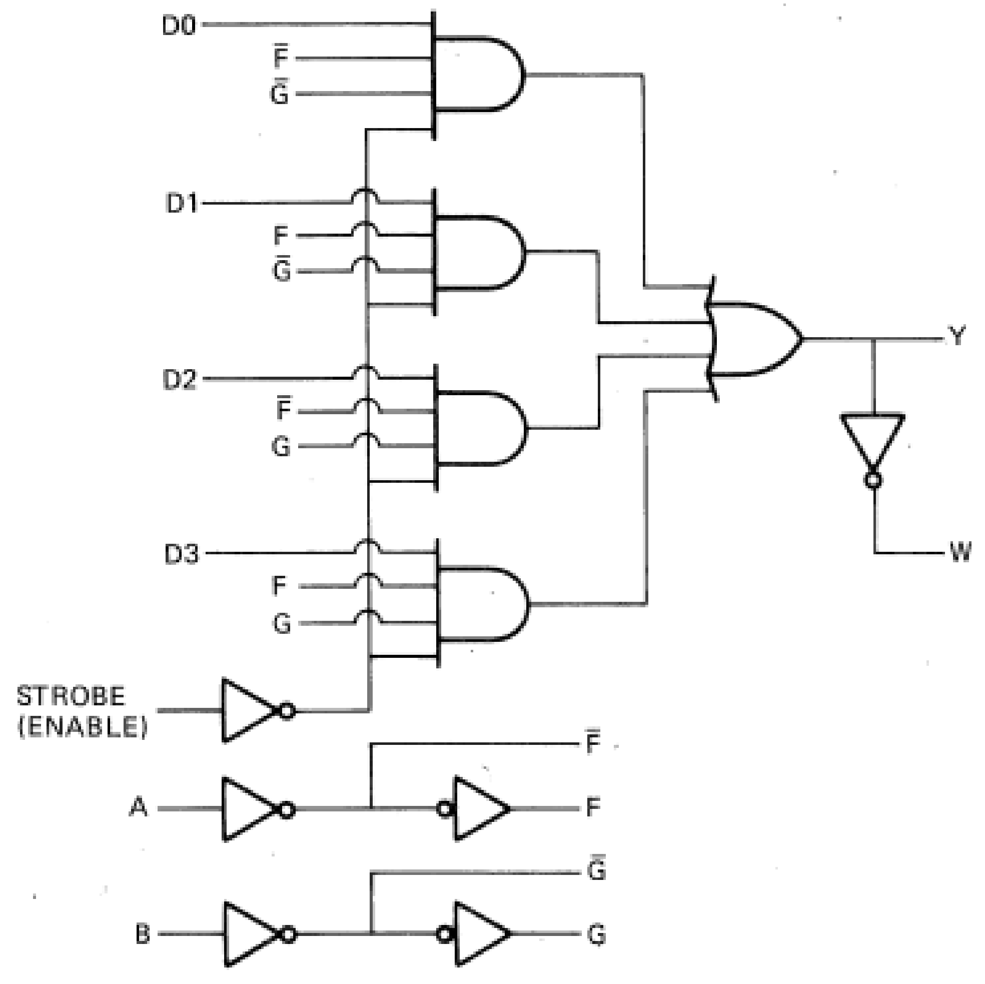


Figure 3-7 4-line to 1-line multiplexer.

Uses four enable gates (4-input AND gates)

F = A, F’ = A’
G = B, G’ = B’

F = 0, G = 0 selects D0

F = 0, G = 1 selects D1

F = 1, G = 0 selects D2

F = 1, G = 1 selects D3

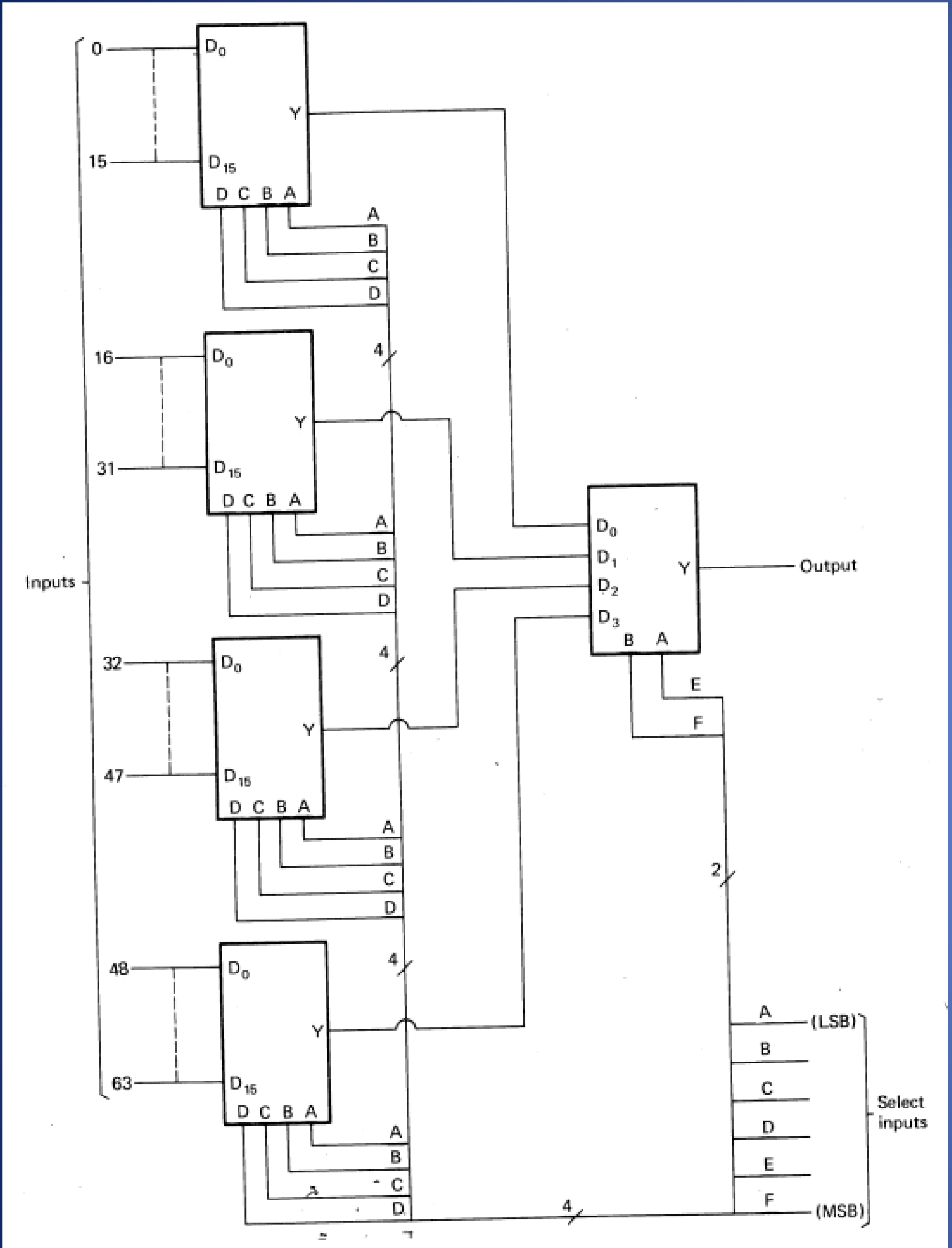


Figure 3-16 Cascading multiplexers.

Large Multiplexers

2 to 1, 4 to 1, 8 to 1, and 16 to 1 muxes are commonly available.

Can cascade muxes to make a larger one.

64 to 1 made from four 16 to 1 cascaded into a 4 to 1.

Has 6 select lines: $2^6 = 64$

Using MUX in an ASM

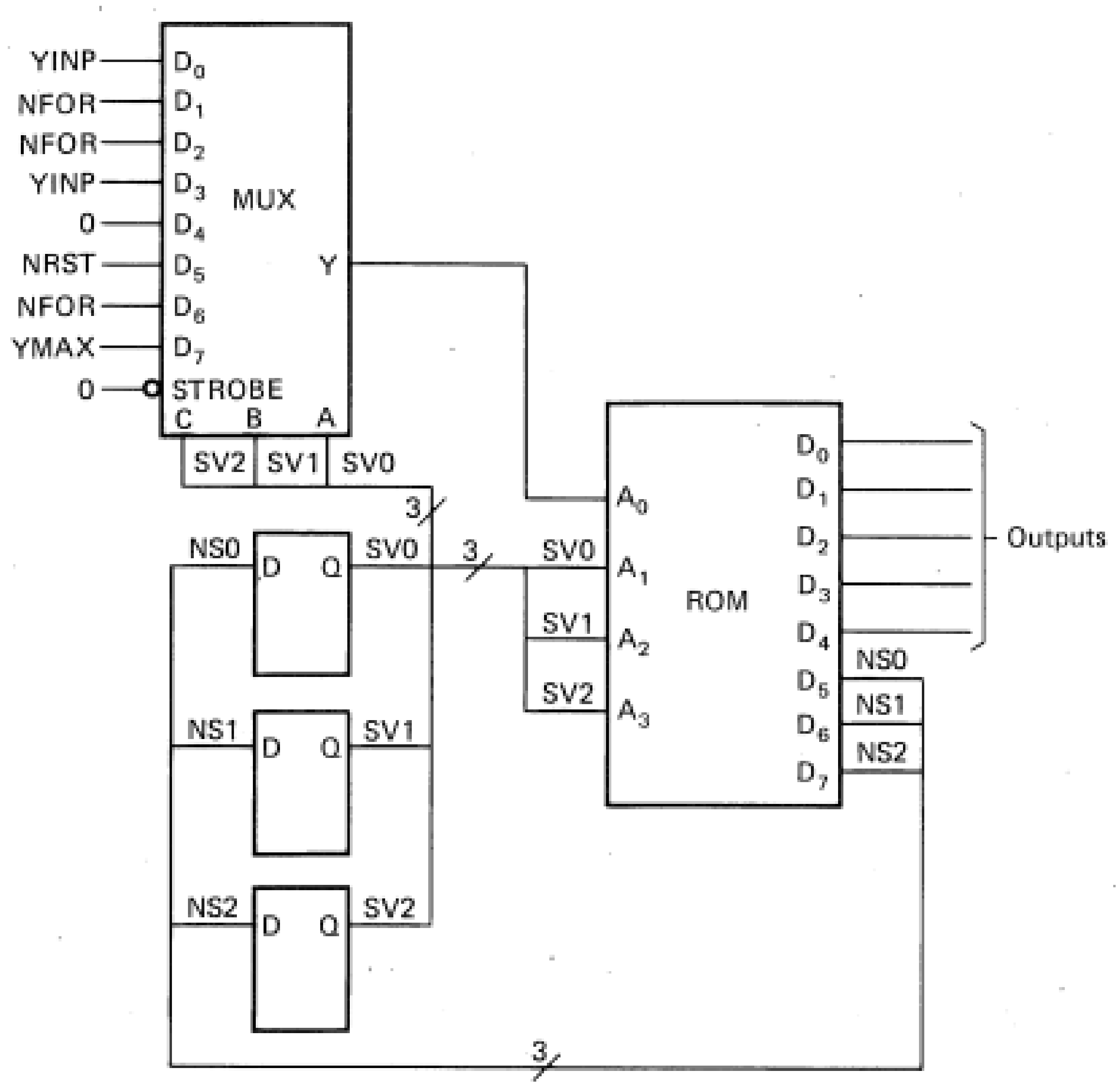


Figure 3-9 Multiplexer for input conditions.

8 to 1 mux connects four ASM inputs into ROM's address pin A0.

Mux select lines CBA tied to state variables.

ONLY 1 input can be tested in each state!

State 0 - YINP
State 1 - NFOR
State 2 - NFOR
State 3 - YINP
State 4 - no input
State 5 - NRST
State 6 - NFOR
State 7 - YMAX

Using MUX in an ASM

ONLY 1 input can be tested in each state (i.e., one decision diamond per state).

Need extra state every time need to test two inputs in same state, but can't test them at same time. ASM is also slower (has to step through more states). May need more FFs due to added states.

With mux: 16 x 8 ROM

Testing 2 Inputs at Same Time

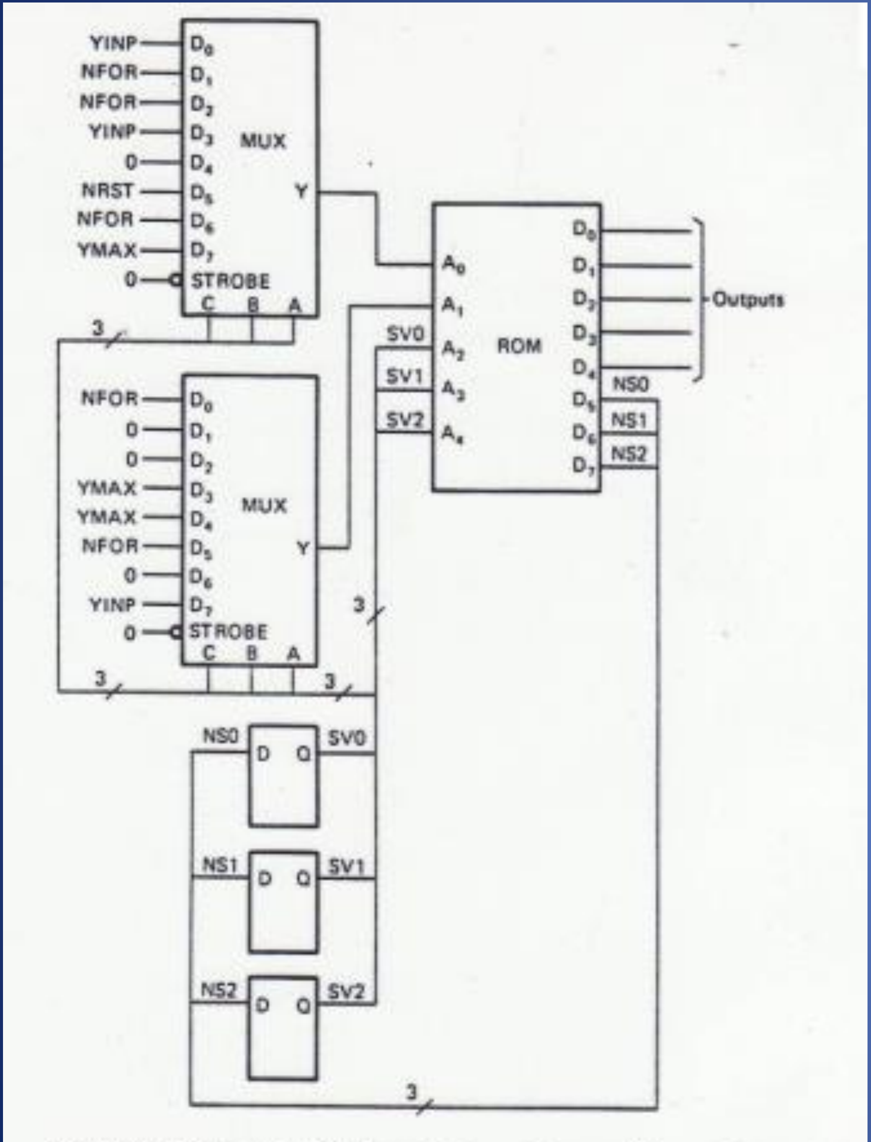


Figure 3-10 Using two multiplexers to test up to two conditions simultaneously.

Use two multiplexers.

Each MUX feeds one address line.
ROM is twice the size (32 x 8).

Smaller than 128 x 8 without muxes.

If have complex ASM – 10 inputs
(2 or 3 tested at the same time)
big savings in ROM size using 2 or 3 muxes.
Assume still has <= 8 states (3 FFs).
10 inputs, 3 tested simultaneously:
With muxes: 64 x 8 ROM
Without muxes: 8192 x 8 ROM
Type 8192 lines into ROM table!!!

MUX ASM with Large Number of States

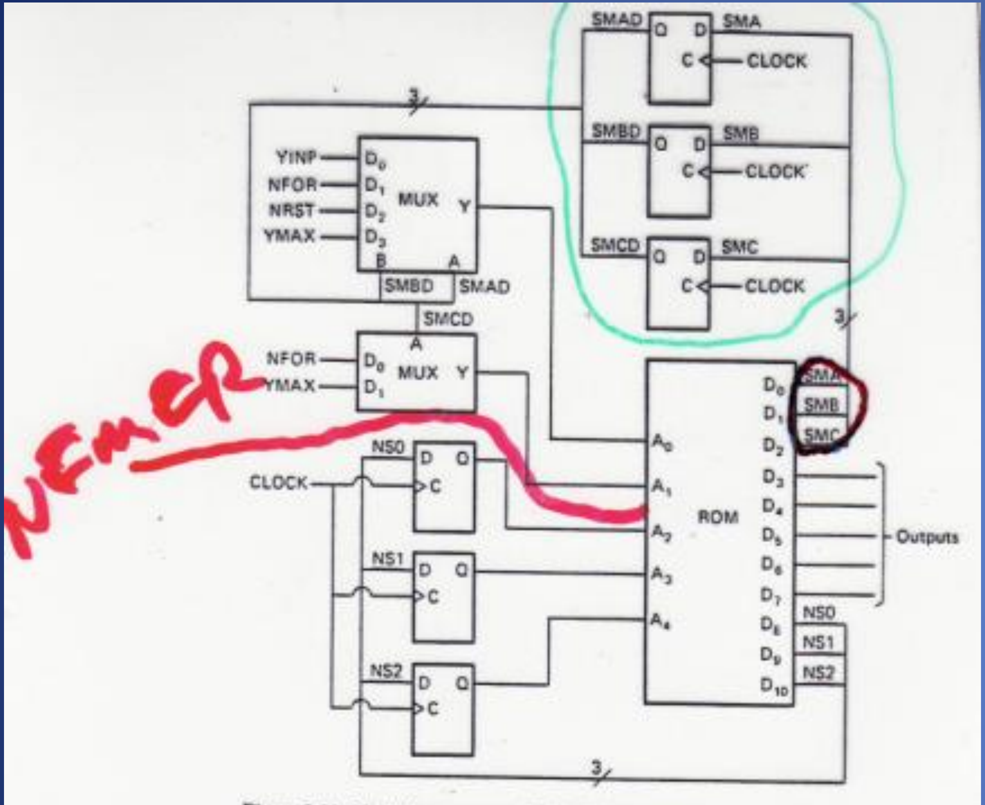


Figure 3-11 Multiplexers controlled by ROM outputs.

If the ASM has 7 state FFs, need 2^7=128 input MUX! But may only use just a FEW of the 128 inputs.

Therefore, connect MUX select inputs to ROM OUTPUT instead of states. Then the ROM can be programmed to select any input in any state.

Multiple MUX's of different sizes can be used, with some often-used inputs connected directly to the ROM (see the NEMER line [in the figure to the right] that needs to be checked in all states).

NOTE: Need flip-flops in MUX select lines. Why?

MUX ASM with Large Number of States

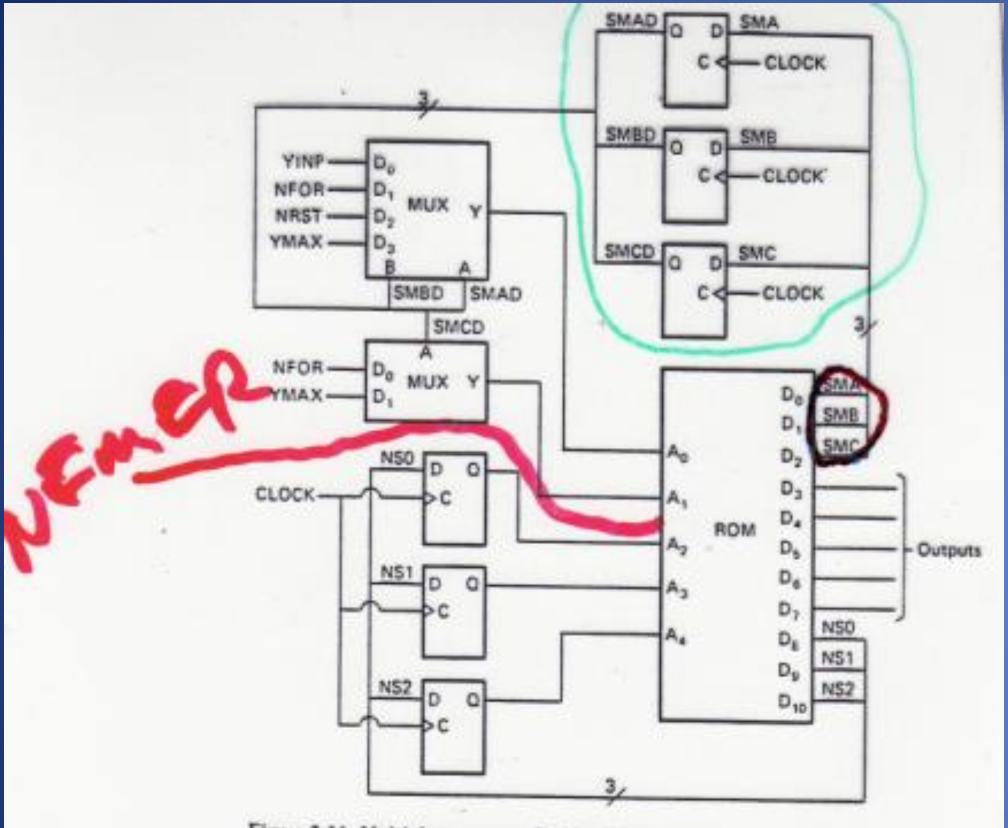


Figure 3-11 Multiplexers controlled by ROM outputs.

NOTE: Need flip-flops in MUX select lines. Why?
ANSWER: The changing input to the ROM could change the outputs of the ROM, changing the select lines, changing the input… Could cause a runaway cycling situation with NO CLOCK needed. FF's synchronize the select lines with the clock. This delays the action of the select line signals one clock pulse, so must be asserted early.

Example Multiplexing ASM Inputs

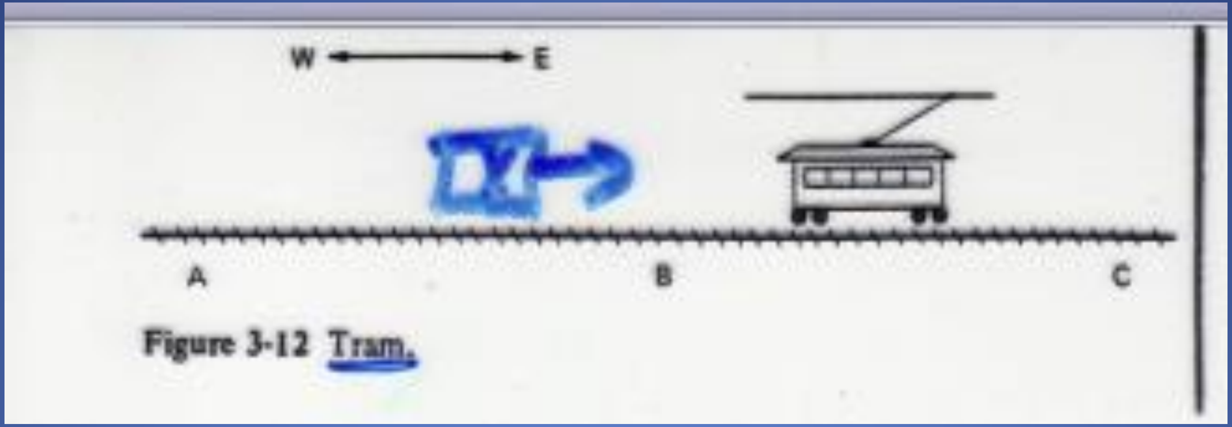


Figure 3-12 Tram.

Airport tram car travels between terminals A, B, and C. It must stop within 1 second of actuating sensors at A, B, C; so using 1s clock. It stops 10 seconds (10 states) at each station. Override button keeps it stopped longer at a station as long as conductor is pressing it.

Example Multiplexing ASM Inputs

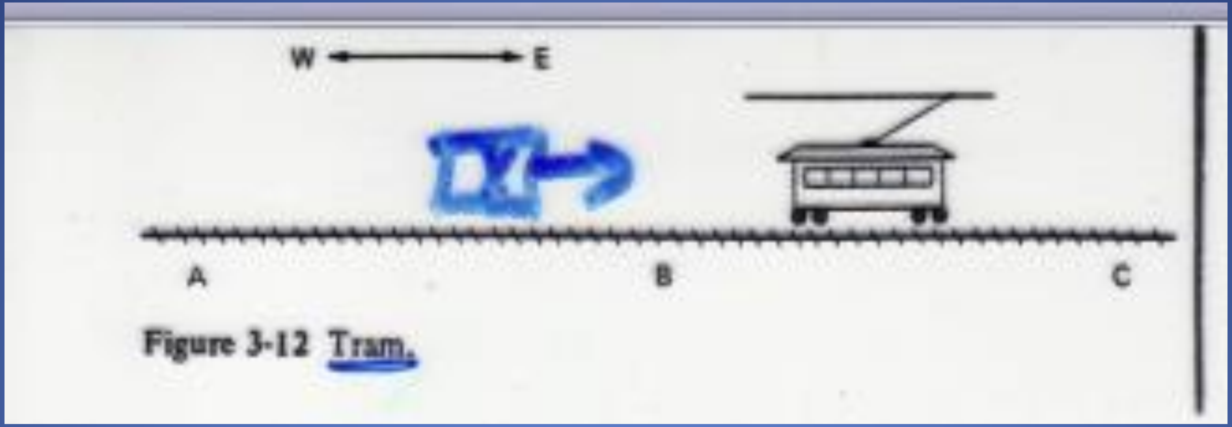
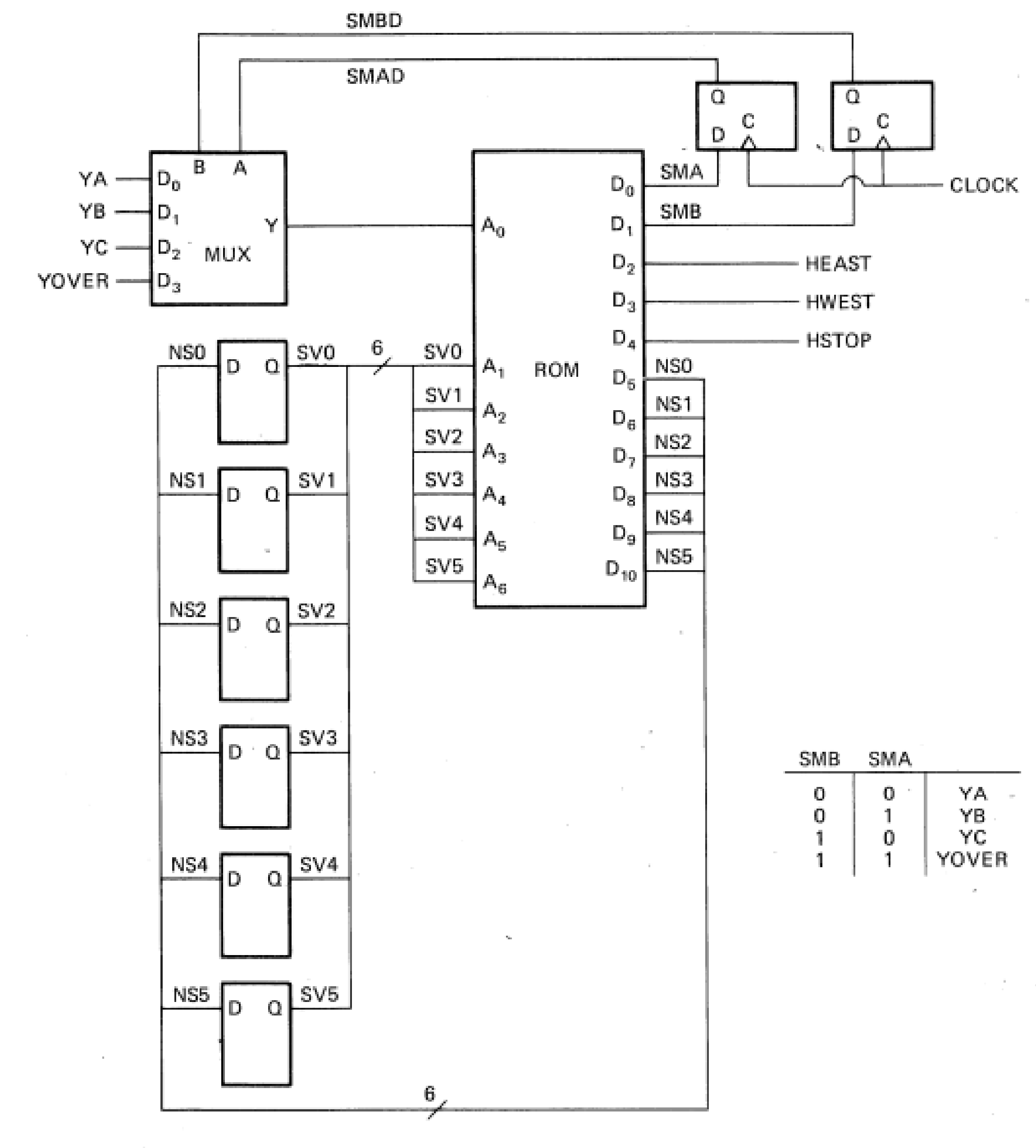


Figure 3-12 Tram.

Using a 4-to-1 Mux for inputs A,B,C, and OVERRIDE, so ASM chart tests one input per state. We will have 44 states. That requires 6 FFs and address lines (2^6=64), plus one address line for the MUX output. Using ROM outputs for the MUX select lines requires two extra ROM outputs. So we are saving ROM inputs at expense of extra ROM outputs.

Circuit for Tram with MUX



SMB	SMA	
0	0	YA
0	1	YB
1	0	YC
1	1	YOVER

Figure 3-14 Tram controller circuit.

4 to 1 MUX has 2 select lines tied to ROM outputs through FFs.

Notice signal labels:

SMBD =

select mux B delayed

Must activate select lines one state early.

If used state variables for select lines, would need 64 to 1 MUX.

Airport Tram ASM chart

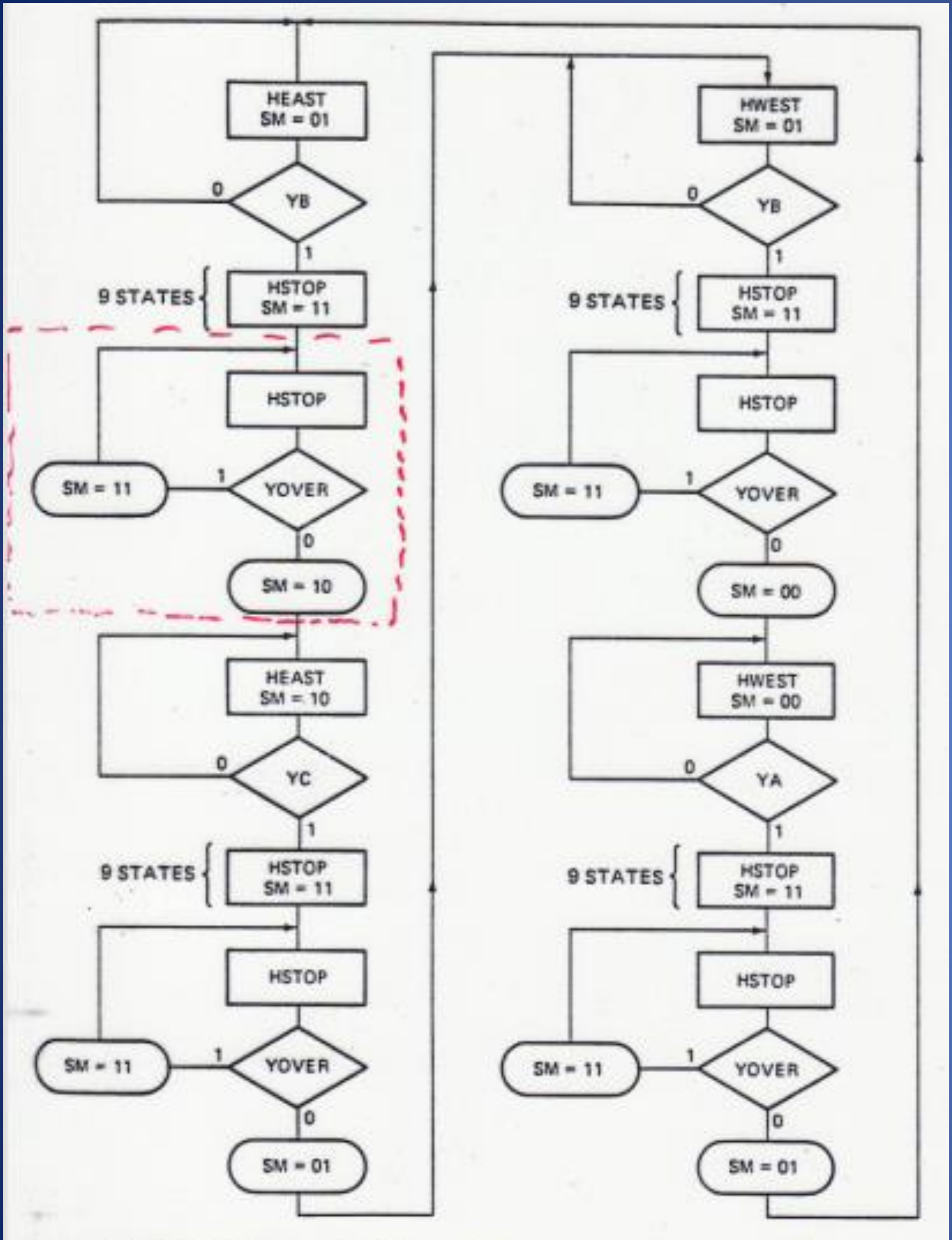


Figure 3-13 ASM chart for tram controller. Clock = 1 second

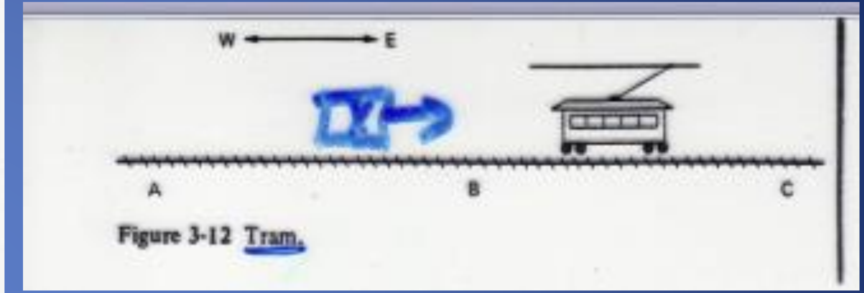
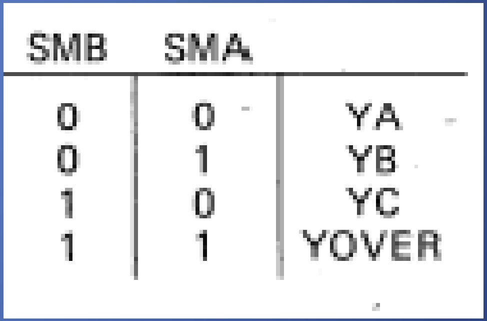


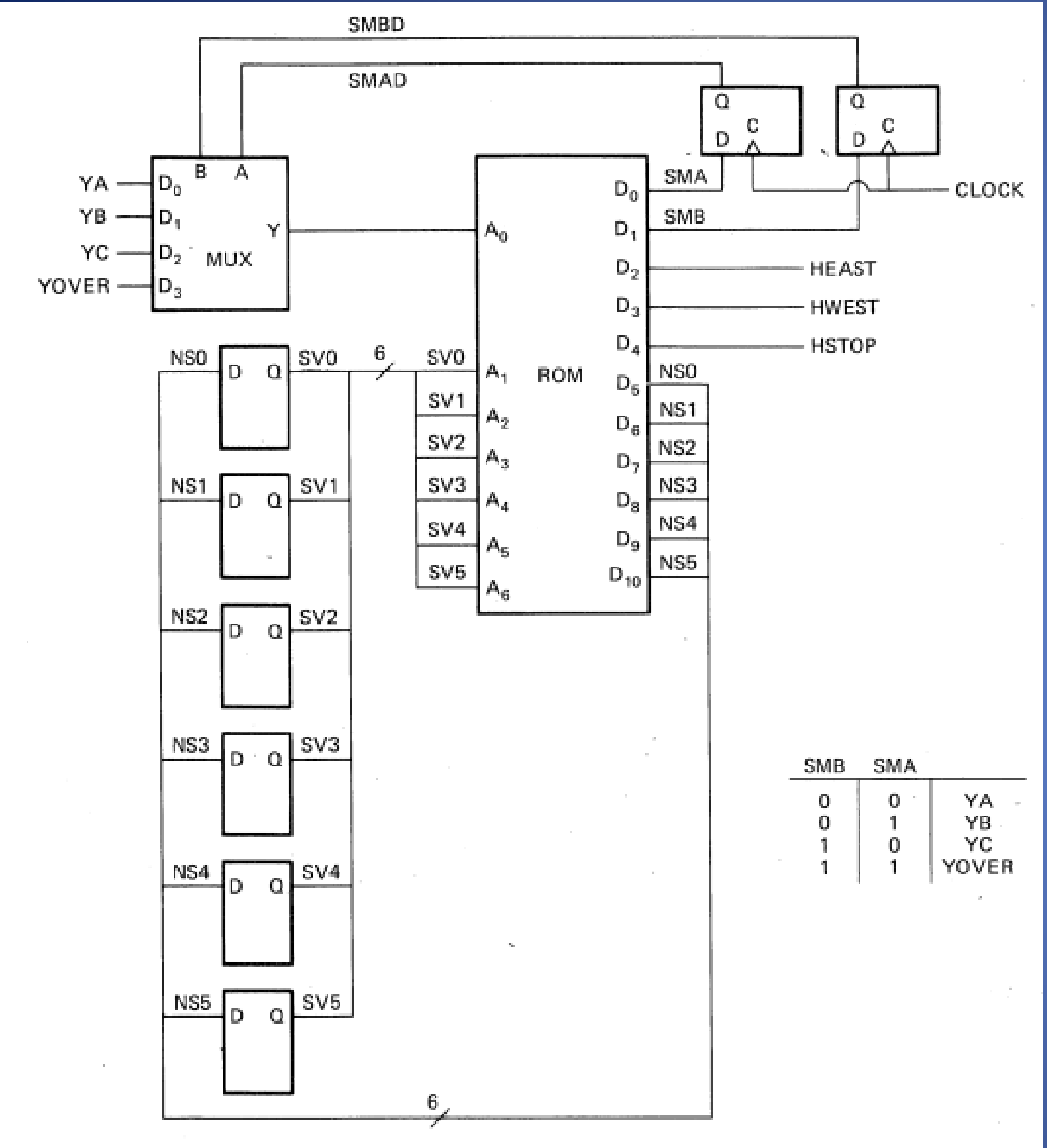
Figure 3-12 Tram.



SMB	SMA	
0	0	YA
0	1	YB
1	0	YC
1	1	YOVER

SMB and SMA are asserted one state early.

Circuit for Tram with MUX



SMB	SMA	
0	0	YA
0	1	YB
1	0	YC
1	1	YOVER

Figure 3-14 Tram controller circuit.

MUX saves 3 address lines, so ROM is 2^7 = 128 x 11

Without mux 2^{10} = 1024 x 9

Circuit saves both ROM and MUX size.

# ROM bits
With MUX: 128 x 11 = 1408
No MUX: 1024 x 9 = 9216

We can use DeCODERS to lower # ROM outputs to 8.

END