

DIGITAL ELECTRONICS II
Revision Examples 2002

Three revision lectures will be given on the topics listed below. The example problems are all taken from past papers and are intended to cover the full range of the course. It should not be assumed that this year's questions will be on these precise topics.

Fri 3 May 10:00	1. State Machines (EEE 1994 Q1) 2. Adder Circuit Propagation Delays (ISE 1996 Q3)
Thu 9 May 10:00	3. Timing (ISE 1996 Q4) 4. CMOS gates (EEE 1994 Q2)
Thu 9 May 2:00	5. Memory interfacing (EEE 1996 Q3) 6. A/D Converter (EEE 1995 Q3)

1. *Figure 1* shows the state diagram of a synchronous state machine having a single input, IN, and a single output, OUT. The state is represented by a two-bit number S0:1 whose value is indicated within each state circle along with the value of OUT. Transitions from a state to itself have not been shown.

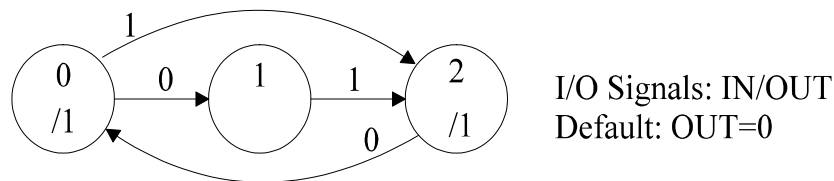


Figure 1

- (a) If the next state is represented by a two bit number NS0:1, derive Boolean expressions for NS1, NS0 and OUT in terms of S1, S0 and IN. You should ensure that the state machine will ultimately follow the correct sequence regardless of its initial state. [7]
 [2]
- (b) Using a 2-bit D-type register and as few logic gates as possible draw a circuit diagram for the state machine. The register operates on the rising edge of the signal CLOCK. [6]
- (c) Complete the timing diagram shown in *Figure 2* by showing the state during each clock cycle and the waveform of OUT. The circuit is initially in state 2. [5]

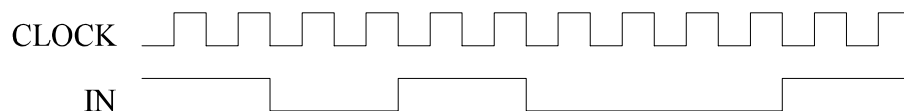


Figure 2

-----Solution-----

(a) We can draw Karnaugh maps for NS1, NS0 and OUT:

	NS1	IN			NS0	IN			OUT	IN	
		0	1			0	1			0	1
S1,S0	00	0	1		00	1	0		00	1	1
	01	0	1		01	1	0		01	0	0
	11	X	X		11	X	X		11	X	X
	10	0	1		10	0	0		10	1	1

From this we get:

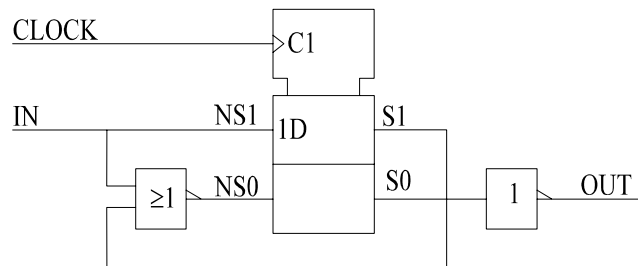
$$NS1 = IN$$

$$NS0 = \overline{S1} \cdot \overline{IN} = \overline{S1 + IN}$$

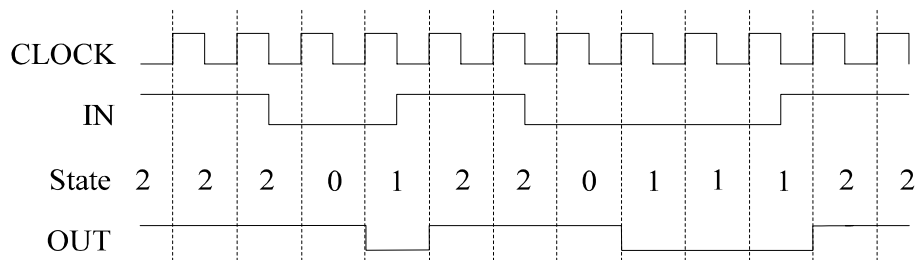
$$OUT = \overline{S0}$$

State 3 branches to either state 0 (IN=0) or state 2 (IN=1) so there is no chance of getting trapped.

(b)



(c)



2. *Figure 3* shows the symbol and propagation delay graph for an m -bit full adder. The delays from the Q inputs have been omitted from the graph as they are equal to those from the P inputs.

The circuit of *Figure 4* uses two full adders together with two multiplexers. The C-1 inputs of the two full adders are connected to logic 0 and logic 1 levels respectively. In the question below, the phrases *simple adder block* and *complex adder block* refers to the circuits of *Figure 3* and *Figure 4* respectively.

- Construct a simplified propagation delay graph for the *complex adder block*. Each multiplexer has a propagation delay of 3 gate delays from its select input (labelled G1) and 2 gate delays from the other inputs. [7]
- Show that the *complex adder block* is functionally equivalent to the *simple adder block*. Explain the relative advantages of the two implementations. [4]
- A 16-bit adder is constructed by cascading four 4-bit full adder blocks. A *simple adder block* is used for the least significant 4 bits and three *complex adder blocks* are used for the 12 most significant bits. Determine the longest propagation delay path for the 16-bit adder. [6]
- Determine the number of bits to which each of the complex adder blocks in part (c) can be increased without increasing the length of the longest propagation delay path. Give the total size of the resultant adder. [3]

[Note: I have changed the way I present this material slightly and no longer talk explicitly about propagation delay graphs. These graphs just give a pictorial representation of the worst-case delay from the inputs to the outputs of a circuit]

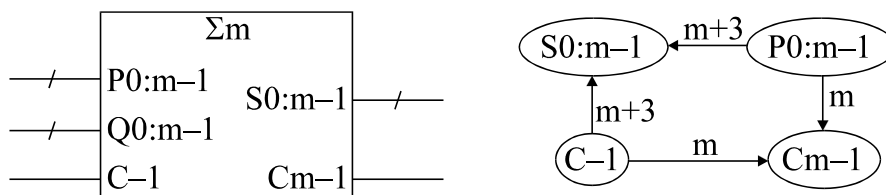


Figure 3

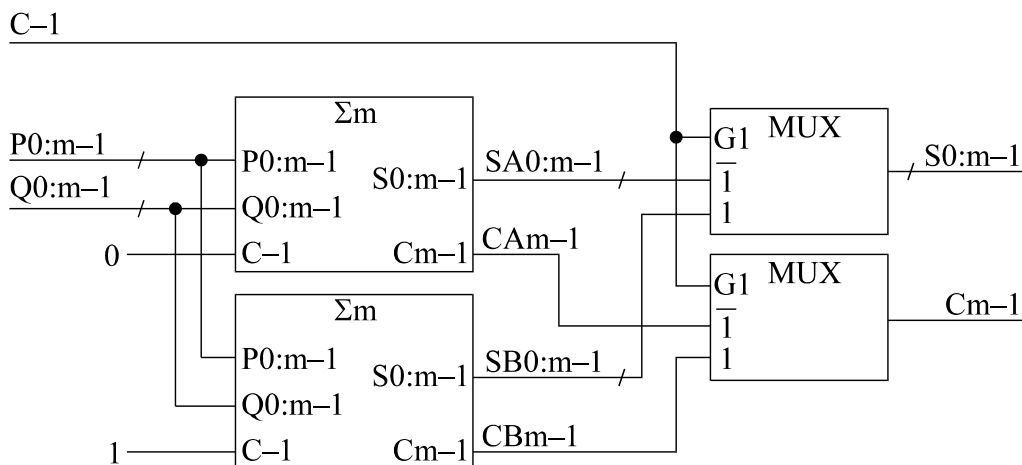
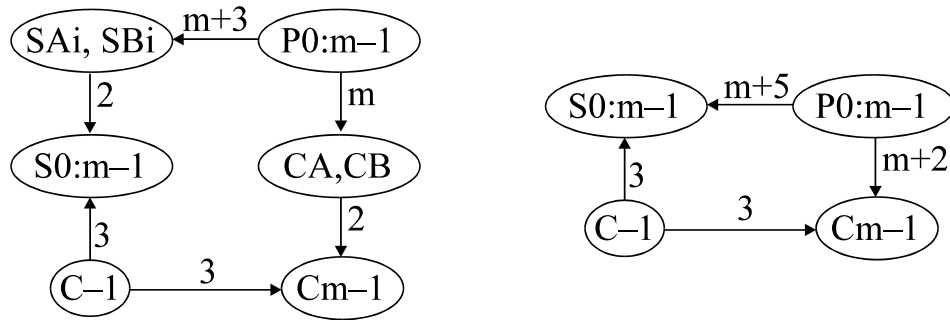


Figure 4

-----Solution-----

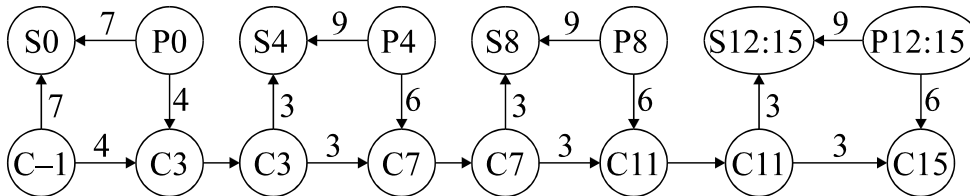
a)



b) The two simple adder blocks are identical except that the C_{-1} input is 0 for the upper one and 1 for the lower one. The multiplexers then select the outputs from the appropriate block according to the actual value of C_{-1} .

The complex adder block uses over twice as much circuitry as the simple adder block. Its main advantage is the reduction (for $m > 3$) in the carry path delay. The delays from P_i to S_i within the block have actually been increased.

c)



The longest path is $P_0 \rightarrow C_3 \rightarrow C_7 \rightarrow C_{11} \rightarrow S_{15} = 4 + 3 + 3 + 3 = 13$

[Note that using a complex block for the 4 least significant bits would lengthen this path by 2]

d) We have the following delays:

$$P_4 \rightarrow C_7 \rightarrow C_{11} \rightarrow S_{15} = 12$$

$$P_8 \rightarrow C_{11} \rightarrow S_{15} = 9$$

$$P_{12} \rightarrow S_{15} = 9$$

We can increase the size of each block to bring these numbers up to 13: make the second block 5 bits and the third and fourth blocks 8 bits. This gives a 25-bit adder with a total delay of 13 gate delays.

3. *Figure 5* shows a circuit comprising two microprocessors and two line-drivers. The microprocessors communicate via a cable that is 300 cm long through which both the clock and data signals travel. The propagation speed of the cable is 15 cm/ns. The propagation delay of the line-drivers may vary between 10 and 15 ns.

As indicated in *Figure 5*, the Z output of each microprocessor changes shortly after the *falling* edge of its clock input while data at the D input is clocked in on the *rising* edge. The timing specifications for these signals are:

$t_{zp} = 70$ ns	Propagation delay of Z output
$t_{zh} = 10$ ns	Hold time of Z output
$t_{ds} = 20$ ns	Setup time of D input
$t_{dh} = 30$ ns	Hold time of D input

- a) The signal CLOCK is a symmetrical squarewave of constant frequency. Determine f_{max} , the clock frequency up to which the circuit will allow microprocessor B to transmit data reliably to microprocessor A. [10]
- b) A supply of registers is available having a propagation delay of 10 ns. The registers are available in two types which respond to rising and falling edges respectively at their clock inputs. The data inputs of the registers have a setup time of 5 ns and a hold time of 5 ns relative to the active clock edge.

Determine the highest value of f_{max} that can be achieved by inserting registers at either or both of points X and Y in the figure. You should indicate the clock polarity of each register that you use and its position in the circuit.

State the additional delay in clock cycles that your circuit modification has inserted between the Z output of microprocessor B and the D input of microprocessor A. [10]

[**Note:** You should not assume that all the information given in this question is required in order to answer it]

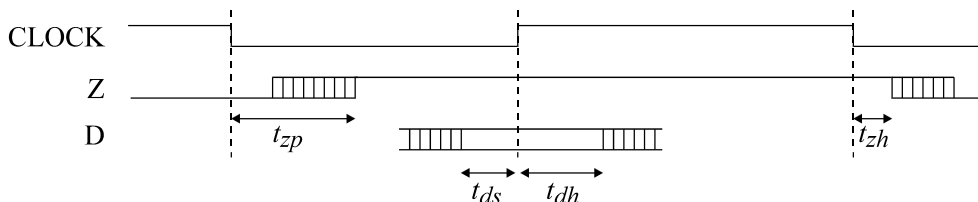
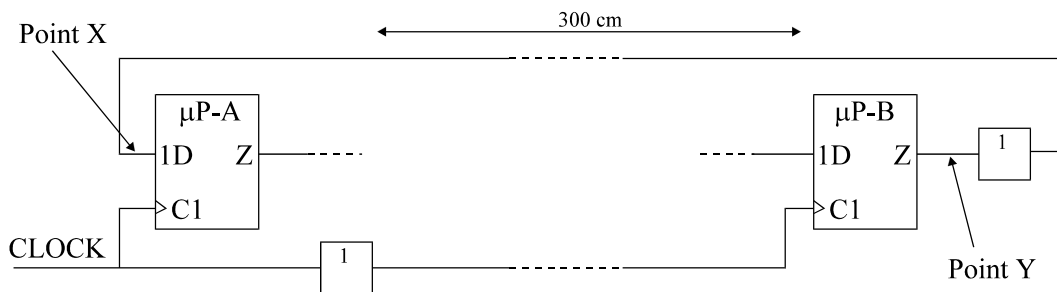


Figure 5

-----Solution-----

We choose the driver delay to be $t_d = 10$ or $t_D = 15$ according to whether it is on the right or left of an inequality respectively. The cable delay is $t_c = 20$ ns. Define the clock period to be T .

a) $t_D + t_c + t_{zp} + t_D + t_c < \frac{1}{2}T - t_{ds} \Rightarrow 15 + 20 + 70 + 15 + 20 < \frac{1}{2}T - 20 \Rightarrow T > 320$

Hence the max frequency is $1/320\text{ns} = 3.125$ MHz

b) A register inserted at point X must respond to a falling clock edge because the data will otherwise change during the hold period t_{dh} .

A register inserted at point Y can respond to either edge, but a falling edge will allow more time to encompass t_{zp} + cable delay.

We therefore put a falling-edge-clocked register at both points X and Y. This inserts an extra 2 clock cycles delay.

For the registers we have $t_s = t_h = 5$ and $t_p = 10$. This gives the following constraints:

i) $\mu\text{P-B to Register-Y: } t_{zp} < T - t_s \Rightarrow T > 75$

ii) $\text{Register-Y to Register-X: } t_D + t_c + t_p + t_D + t_c < T - t_s \Rightarrow T > 85$

iii) $\text{Register-X to } \mu\text{P-A: } t_p < \frac{1}{2}T - t_{ds} \Rightarrow T > 60$

Hence the max frequency is $1/85\text{ns} = 11.8$ MHz

-----Solution-----

(a) $X = \overline{A \cdot B \cdot C \cdot D + E} = (\overline{A + B + C + D}) \cdot \overline{E}$

(b) For a high output, the worst case is when Q5 is on but only one of Q1, Q2, Q3 and Q4 is on. The output impedance in this case is $\frac{2}{W_1} + \frac{2}{W_5}$ which we wish to set to 0.1 k Ω .

Hence $\frac{2}{W_5} = 0.1 - \frac{2}{W_1} = \frac{0.1W_1 - 2}{W_1} = \frac{W_1 - 20}{10W_1} \Rightarrow W_5 = \frac{20W_1}{W_1 - 20}$

(c) By symmetry, $W_1 = W_2 = W_3 = W_4$ and $W_6 = W_7 = W_8 = W_9$.

For a high output, we must satisfy the previous expression while minimizing $4W_1 + W_5$.

Substituting for W_5 gives $4W_1 + \frac{20W_1}{W_1 - 20} = \frac{4W_1^2 - 60W_1}{W_1 - 20} = \frac{4(W_1^2 - 15W_1)}{W_1 - 20}$

Differentiating and setting to zero to find the minimum gives:

$$\frac{d}{dW_1} = 4 \times \frac{(2W_1 - 15)(W_1 - 20) - (W_1^2 - 15W_1)}{(W_1 - 20)^2} = 0$$

$$\Rightarrow (2W_1 - 15)(W_1 - 20) - (W_1^2 - 15W_1) = W_1^2 - 40W_1 + 300 = 0$$

$$\Rightarrow W_1 = 10 \quad \text{or} \quad 30$$

$$\Rightarrow W_5 = -20 \quad \text{or} \quad 60$$

For a low output we must clearly have the resistance of Q10 = 100 Ω and of Q6 to Q9, 25 Ω each. From this we get $W_6 = W_7 = W_8 = W_9 = 40$ and $W_{10} = 10$.

5. The circuit of *Figure 7* forms part of a logic analyser in which the eight lines DATA0:7 are sampled repetitively and their values stored in a memory. The circuit comprises a static RAM memory, a counter, an 8-bit register and two flipflops. The propagation delays of the counter, register and flipflops may vary between 5 and 10 ns and may have different values for rising and falling output transitions. The flipflops have setup and hold times of 5 ns and 0 ns respectively relative to the clock rising edge. The timing requirements for a memory write cycle are shown in the figure.

- Draw a timing diagram showing the waveforms of CLOCK, CNT, $\overline{\text{WRITE}}$, A0:12 and D0:7. The signal CLOCK is a symmetrical squarewave of constant frequency. [7]
- For each of the timing constraints shown in the figure, determine the corresponding restriction on the period of CLOCK. [10]
- Hence determine the maximum CLOCK frequency for reliable circuit operation. [3]

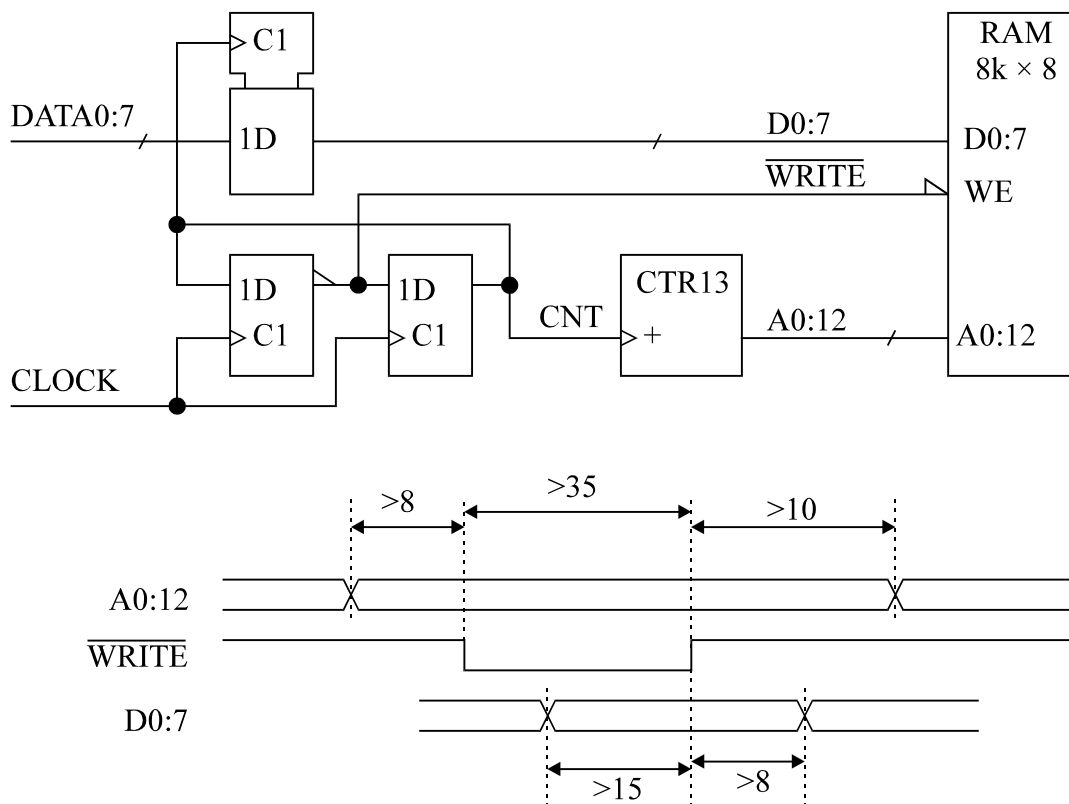
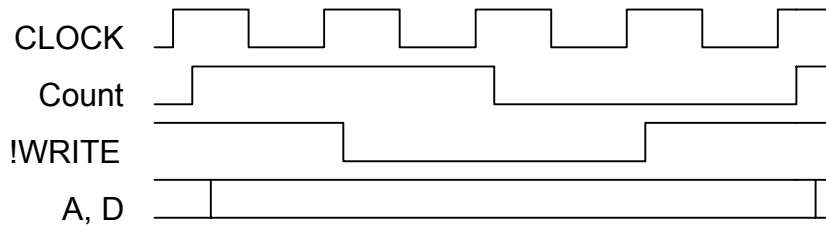


Figure 7

-----Solution-----

- a) The waveforms of the various signals are drawn below with exaggerated propagation delays:



- b) We need to check each of the timing requirements in the diagram. In the expressions below, we use T for the clock period and t for the propagation delay of a counter/register/flipflop. All the t values are then replaced by 10 or 5 according to whether they are on the left or right respectively of an $<$ sign.

$$\text{Addr setup: } 2t+8 < T+t \quad \Rightarrow \quad T > 20 + 8 - 5 = 23$$

$$\text{Write pulse: } t + 35 < 2T + t \quad \Rightarrow \quad T > (10 + 35 - 5)/2 = 20$$

$$\text{Addr hold: } t+10 < T+2t \quad \Rightarrow \quad T > 10 + 10 - 10 = 10$$

$$\text{Data setup: } 2t+15 < 3T+t \quad \Rightarrow \quad T > (20 + 15 - 5)/3 = 10$$

$$\text{Data hold: } t+8 < T+2t \quad \Rightarrow \quad T > 10 + 8 - 10 = 8$$

- c) The address setup is therefore the limiting factor and the maximum clock frequency is 43.4 MHz.

6. *Figure 8* shows the circuit of a 4-bit flash analogue-to-digital converter consisting of 14 identical resistors, 15 comparators and a combinational logic block. The input voltage V_{in} lies in the range ± 0.9 V and the converter is designed to convert V_{in} to a 2's complement value Z where Z is the nearest integer to $(V_{in} \div 0.125$ V). Thus an input voltage of 0.3 V would be converted to the value +2.

a) Calculate the required values of V_L and V_H . [6]

b) Show that the logic for Z_1 can be implemented as

$$Z_1 = C_{13} + \overline{C_{11}} \cdot C_9 + \overline{C_7} \cdot C_5 + \overline{C_3} \cdot C_1 \quad [8]$$

Derive similar sum-of-products expressions for Z_0 , Z_2 and Z_3 .

c) If each logic gate introduces a delay of 1 unit, derive the delay introduced by the logic circuitry when, due to a rising input voltage, the output [6]

(i) changes from 0 to 1 and

(ii) changes from 1 to 2

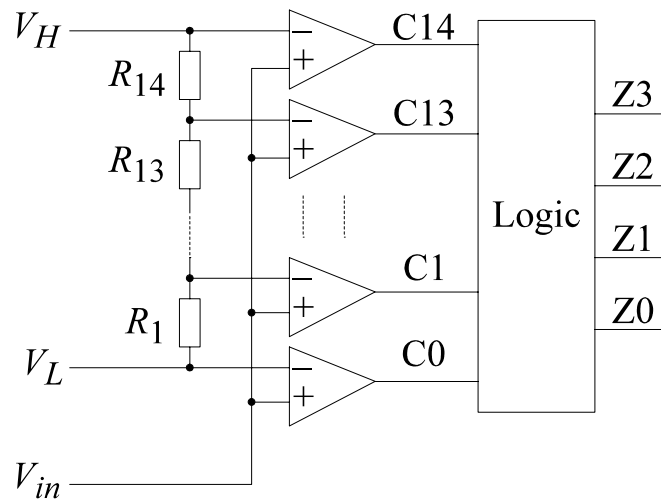


Figure 8

-----Solution-----

a) The upper and lower thresholds are at $13/16$ and $-15/16$ respectively. Thus $V_H=0.8125$ and $V_L = -0.9375$ V. Note that the ± 0.9 V mentioned in the question doesn't affect these values at all.

b)

C14, ..., C0	Z3, ..., Z0
1111111111111111	0111
0111111111111111	0110
0011111111111111	0101
0001111111111111	0100
0000111111111111	0011
0000011111111111	0010
0000001111111111	0001
0000000111111111	0000
0000000011111111	1111
0000000001111111	1110
0000000000111111	1101
0000000000011111	1100
0000000000001111	1011
0000000000000111	1010
0000000000000011	1001
0000000000000001	1000

From inspection we get

$$Z3 = \overline{C7}$$

$$Z2 = C11 + \overline{C7} \cdot C3$$

$$Z1 = C13 + \overline{C11} \cdot C9 + \overline{C7} \cdot C5 + \overline{C3} \cdot C1$$

$$Z0 = C14 + \overline{C13} \cdot C12 + \overline{C11} \cdot C10 + \overline{C9} \cdot C8 + \overline{C7} \cdot C6 + \overline{C5} \cdot C4 + \overline{C3} \cdot C2 + \overline{C1} \cdot C0$$

If we know that V_{in} is in the range ± 0.9 V we can actually omit the final $C0$. We could save an inverter and speed things up a bit by reversing the connections to the $C7$ comparator.

c) When Z goes from 0 to 1, $C8$ goes high causing $Z0$ to go high with a delay of 2. When Z goes from 1 to 2, $C9$ goes high causing $Z1$ to go high with a delay of 2 and $Z0$ to go low with a delay of 3 (because of the inverter).