

# EE 100/42 Spring 2009

## Solutions to Homework 11

Problem 1 It is possible to design a counter that will count from 0000 (decimal 0) to 1010 (decimal 10), and return to 0000 again. By examining the sequence of states (see Table 1), we can derive the states that  $Q_0$ ,  $Q_1$ ,  $Q_2$ , and  $Q_3$  want to become. On the rising edge of

Table 1: STATE EVOLUTION IN PROPOSED COUNTER.

$Q_3$	$Q_2$	$Q_1$	$Q_0$
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0

the clock, it is possible to count from 0000 to 1111 by the following transitions:

$$\begin{aligned}
 Q_0 &\rightarrow \overline{Q_0} \\
 Q_1 &\rightarrow Q_1 \oplus Q_0 \\
 Q_2 &\rightarrow Q_2 \oplus (Q_0 Q_1) \\
 Q_3 &\rightarrow Q_3 \oplus (Q_0 Q_1 Q_2)
 \end{aligned}$$

These transitions will repeatedly count from 0000 to 1111. To force the states to reset after 1010, introduce the variable

$$R = \overline{(Q_0 Q_1 \overline{Q_2} Q_3)}$$

Notice that  $R = 0$  if and only if  $Q_3 Q_2 Q_1 Q_0 = 1010$ .

Now, to reset all of the states to zero after the last row in Table 1, use the following transitions:

$$\begin{aligned}
 Q_0 &\rightarrow (\overline{Q_0}) R \\
 Q_1 &\rightarrow (Q_1 \oplus Q_0) R \\
 Q_2 &\rightarrow (Q_2 \oplus (Q_0 Q_1)) R \\
 Q_3 &\rightarrow (Q_3 \oplus (Q_0 Q_1 Q_2)) R
 \end{aligned}$$

These were derived by recognizing that, for any boolean variable  $A$ ,  $1(A) = A$  and  $0(A) = \overline{A}$ .

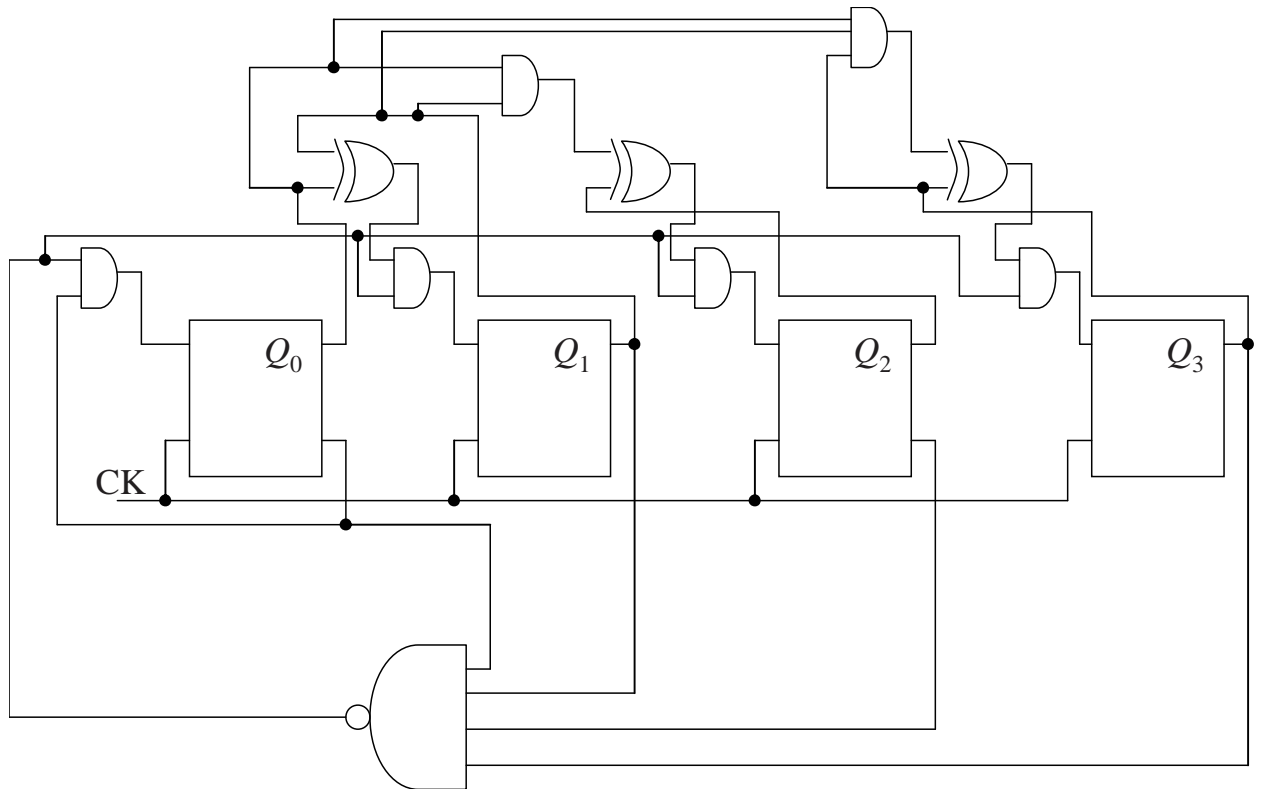


Figure 1: Circuit diagram for divide-by-11 counter. Connections are represented by black dots where the wires overlap.

The circuit diagram is shown in Figure 1.

It is possible to design a general divide-by- $n$  counter,  $1 \leq n \leq 16$ , by different definitions for  $R$ .

Problem 2 The PLA is given in Figure 2. The following identities were useful in designing the circuit:

$$A(1) = A$$

$$A(0) = 0$$

$$A + 0 = A$$

for any boolean variable  $A$ .

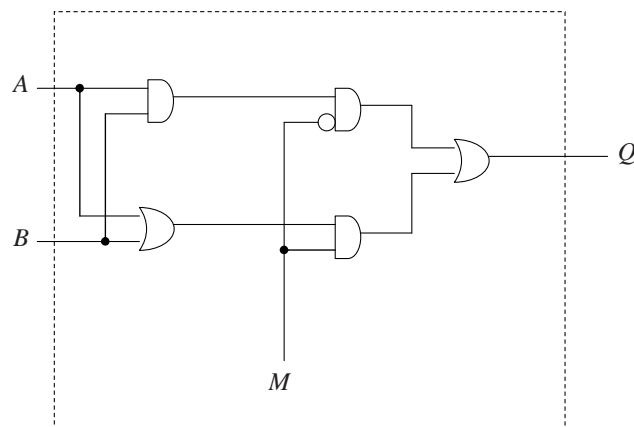


Figure 2: Circuit diagram for programmable logic array.  $M$  is included inside the PLA because it represents memory. Connections are represented by black dots where the wires overlap.

Problem 3

- **medium**

Material substance that is used to carry information from a sender to a receiver. Examples of different media include air, fiber optics, and copper.

- **coding**

The altering of signal information to achieve one or more of the following characteristics: efficiency, security, and robustness. An example of coding for efficiency is the run length coding employed by fax machines, where the lengths of sequences of 1s and 0s are transmitted instead of the actual sequences themselves. Coding for security is exemplified by secure shell (SSH), where the notion of public and private keys make secure communication over a shared medium possible. Coding for robustness is illustrated by using 000 for “0” and 111 for “1,” which makes the transmission of information more robust to individual changes in data bits.

- **full-duplex**

System in which communication is allowed in both directions (versus broadcast) simultaneously (versus half-duplex). An example of this is telephone, where two people can communicate in opposite directions at the same time.

- **packet**

Formatted unit of data that represents part of a signal that is typically transmitted across the Internet. Packets may arrive at the destination out of order, so each one contains a header in addition to the payload (data from the original signal). The header may contain synchronization bits, sender/receiver information, and packet order information.

- **virtual packet network**

Network that assigns dedicated, fixed, real-world lines between two communicating parties. An example of this is a telephone network, where the connection between two users is fixed and held constant until the phone call terminates. This is in contrast to the Internet, where packets arrive at the destination from possibly different paths.

Problem 4 A two-input CMOS NAND gate will output  $Q = 0$  if and only if both inputs  $A$  and  $B$  are 1. The circuit realization using FET switches is depicted in Figure 3.

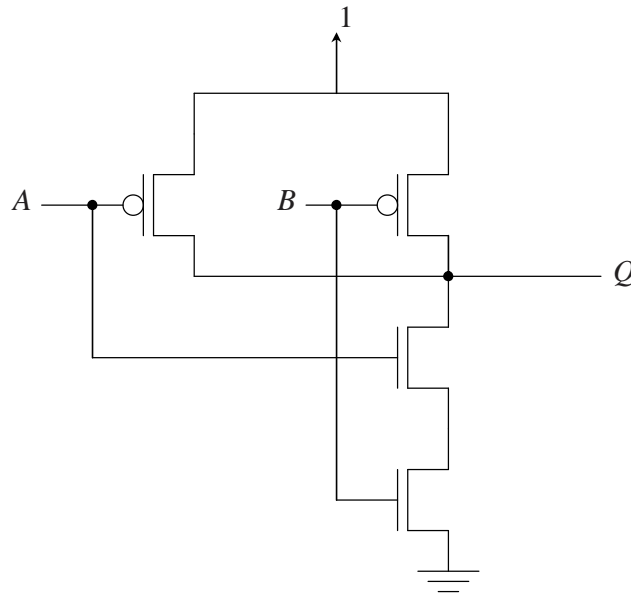


Figure 3: Circuit diagram for CMOS NAND gate using FET switches. Connections are represented by black dots where the wires overlap.

- 10.40 (a) The output is high if either or both of the inputs are high. If both inputs are low, the output is low. This is an OR gate.  
(b) The output is high only if both inputs are high. This is an AND gate.
- 10.48 (a) Assuming that the diode is an open-circuit, we can compute the node voltages

using the voltage-division principle.

$$\begin{aligned}v_1 &= 4 \frac{200}{200 + 200} \text{ V} \\ &= 2 \text{ V} \\ v_2 &= 4 \frac{100}{300 + 100} \text{ V} \\ &= 1 \text{ V}\end{aligned}$$

Then, the voltage across the diode is  $v_D = v_1 - v_2 = 1 \text{ V}$ . Because  $v_D$  is greater than the diode activation voltage of  $0.7 \text{ V}$ , the diode must be operating in the “on” state.

- (b) Assuming that the diode operates as a voltage source (to incur the constant voltage drop of  $0.7 \text{ V}$ ), we can use KVL to write:

$$v_1 - v_2 = 0.7 \text{ V}$$

We can perform KCL at nodes 1 and 2, defining the current flowing from node 1 to node 2 as  $i_D$ .

KCL at node 1:

$$\frac{4 - v_1}{200} + \frac{0 - v_1}{200} - i_2 = 0$$

KCL at node 2:

$$\frac{4 - v_2}{300} + \frac{0 - v_2}{100} + i_2 = 0$$

Now, we have three equations and three unknowns. Solving these equations gives:

$$\begin{aligned}v_1 &= 1.829 \text{ V} \\ v_2 &= 1.129 \text{ V} \\ i_D &= 1.1714 \text{ mA}\end{aligned}$$

Since  $i_D$  is positive, the diode operation is consistent with the assumption.