

Final Exam: Graduate Course – VLSI Testing

Auburn Univ., ELEC 7250, Spring 2006

May 6, 2006

Instructions (please read before you proceed):

1. Please read all problems before starting to answer. Problems can be answered in any order.
2. Attempt all five problems and attempt all parts within each problem.
3. Answers can be written on question sheets or separate sheets or a combination. Each sheet should have a page number and problem number. On the first sheet write your name and the total number of sheets you are submitting.
4. Before handing in your answers, please check them thoroughly. If necessary, extra 10 minutes can be allowed for checking.

Problem 1: Fault Coverage and Defect Level (20 Points)

- (a) The fault coverage of tests for a VLSI chip is given by:

$$T = f(v) = 1 - e^{-\alpha v}$$

where v is the number of vectors and α is an empirical constant determined from the fault simulation data. The tester yield of the chip is given by:

$$Y = Y_0 + (1 - Y_0)e^{-\beta v}$$

where Y_0 is the true yield and β is another empirical constant obtained from the tester data on chip fallout versus vector number. Derive the following expression for yield as a function of the fault coverage:

$$Y(T) = Y_0 + (1 - Y_0)(1 - T)^{\beta/\alpha}$$

(7 points)

(b) Derive the following expression for the defect level:

$$DL(T) = \frac{1 - Y_0}{1 - Y_0 + Y_0(1 - T)^{-\beta/\alpha}}$$

(7 points)

(c) Given, $\beta = 2\alpha$, $Y_0 = 0.8$ and $T = 0.98$, determine the defect level in parts per million (6 points)

Solution to Problem 1

(a) To obtain the tester yield as a function of fault coverage T , we first express v as a function of T from the first equation:

$$v = -\frac{1}{\alpha} \ln(1 - T)$$

When substituted in the second equation, this gives the required expression:

$$\begin{aligned} Y(T) &= Y_0 + (1 - Y_0) e^{(\beta/\alpha)\ln(1-T)} \\ &= Y_0 + (1 - Y_0)(1 - T)^{\beta/\alpha} \end{aligned}$$

(b) Defect level is obtained as follows:

$$\begin{aligned} DL(T) &= \frac{Y(T) - Y_0}{Y(T)} \\ &= \frac{(1 - Y_0)(1 - T)^{\beta/\alpha}}{Y_0 + (1 - Y_0)(1 - T)^{\beta/\alpha}} \\ &= \frac{1 - Y_0}{1 - Y_0 + Y_0(1 - T)^{-\beta/\alpha}} \end{aligned}$$

(c) Substituting the given data in the expression for $DL(T)$, we get

$$\begin{aligned} DL &= \frac{0.2}{0.2 + 0.8 \times 0.02^{-2}} \\ &= \frac{0.2}{0.2 + (0.8/0.0004)} \\ &= \frac{100}{1000100} \end{aligned}$$

The defect level is 100 parts per million.

Problem 2: Pseudorandom Pattern Generation (20 Points)

- (a) Input register RegA is initialized to 0 and RegB to decimal 6 as shown in Figure 1, and the output register RegC is initialized to 0. The content of register RegC, except for the most significant bit (MSB), is fed back to RegA. All three registers have a common clock. Does RegC produce a full-length four-bit pseudorandom pattern sequence? (8 points)

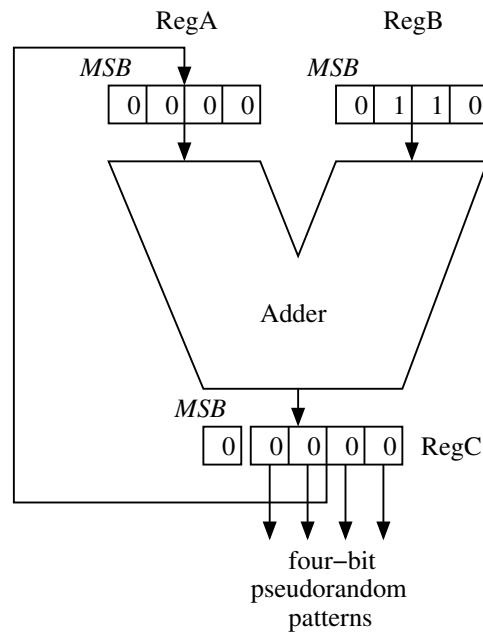


Figure 1: Circuit for pseudorandom pattern generator of Problem 2.

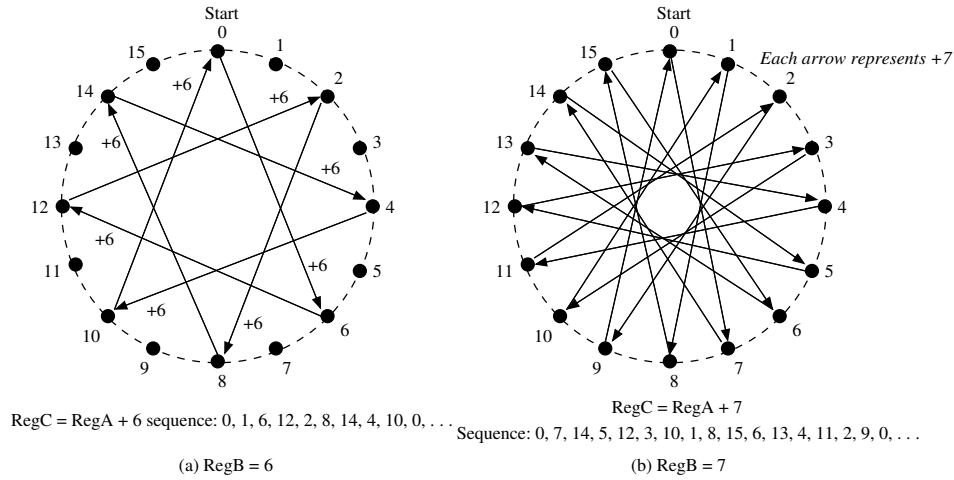
- (b) Find a value for RegB that will generate all sixteen patterns before repeating. Write the sequence. (6 points)
- (c) Find all possible values of RegB for which the pattern generator of Figure 1 will produce 16-pattern sequences. (6 points)

Solution to Problem 2

- (a) The output pattern after n th clock ($n = 1, 2, \dots$) is a four-bit binary integer that can be represented in decimal form as:

$$RegC(n) \text{ modulo } 16 = [RegC(n-1) + RegB] \text{ modulo } 16$$

Although $RegC(0) = 0$ here, it can start in any other state also. Modulo 16 integers are represented on a circle in the following figure. Starting at 0, every arrow corresponds to +6 and leads to the next number. As shown in the left figure (a), the sequence repeats after half of the numbers have been generated. This is not a full-length sequence.



Sequences generated by circuit of Figure 1.

- (b) $RegB = 6$ did not produce a full-length sequence because 6 and 16 have a common factor 2. Hence, they are not *relatively prime*. Since 7 and 16 are relative primes, i.e., they have no non-trivial (other than 1) common factor, $RegB = 7$ generates a full-length sequence. This is shown in part (b) of the above figure.
- (c) All numbers that are relative primes of 16, when placed in $RegB$ will generate full-length sequences. These are 1, 3, 5, 7, 9, 11, 13 and 15. Note that none has a non-trivial factor common with 16. Of these, 1 and 15 will simply generate numbers in increasing and decreasing orders, respectively.

Note: For details of relatively prime numbers, see D. E. Knuth, *The Art of Computer Programming, Second Edition, Volume 1, Fundamental Algorithms*, Reading, Massachusetts: Addison-Wesley, 1975, Section 1.2.4.

Problem 3: Fault Modeling and Simulation (20 Points)

- (a) For the exclusive-OR circuit of Figure 2 count the uncollapsed number of single stuck-at faults. (4 points)
- (b) Determine the fault equivalences among the faults, H s-a-0, J s-a-0 and K s-a-0. (8 points)
- (c) Using the parallel fault simulation algorithm (assuming a four-bit machine word), determine which of the three single faults, F s-a-1, H s-a-1, and L s-a-0, are detected by the input vector $A = 1$, $B = 0$. (8 points)

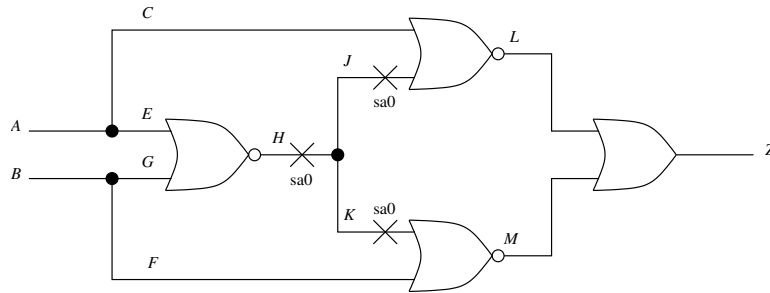


Figure 2: Circuit for Problem 3.

Solution to Problem 3

(a) Total uncollapsed faults

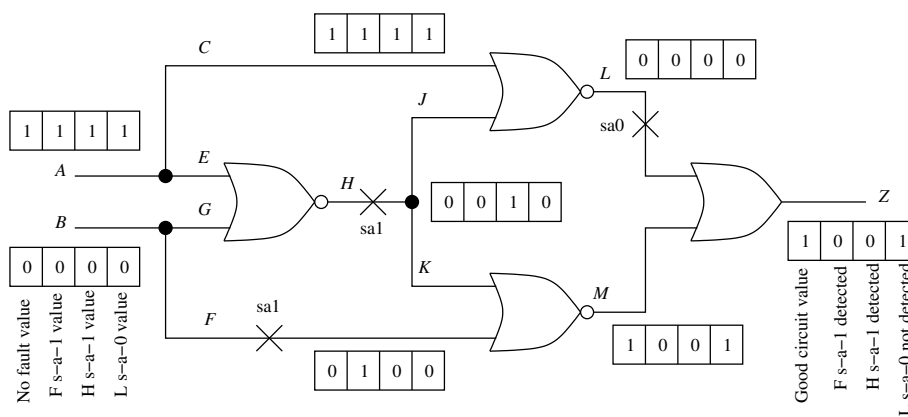
$$\begin{aligned}
 &= 2 \times (\#PI + \#gates + \#fanout_branches) \\
 &= 2 \times (2 + 4 + 6) = 24
 \end{aligned}$$

(b) Three faulty functions are:

$$\begin{aligned}
 Z(H \text{ s-a-} 0) &= \bar{A} + \bar{B} \\
 Z(J \text{ s-a-} 0) &= \bar{A} + \overline{\bar{A} + \bar{B} + B} = \bar{A} + (A + B)\bar{B} \\
 &= \bar{A} + A\bar{B} = \bar{A} + \bar{A}\bar{B} + A\bar{B} = \bar{A} + \bar{B} \\
 Z(K \text{ s-a-} 0) &= \bar{B} + \overline{\bar{A} + \bar{B} + A} = \bar{B} + (A + B)\bar{A} \\
 &= \bar{B} + \bar{A}B = \bar{B} + \bar{A}\bar{B} + \bar{A}B = \bar{B} + \bar{A}
 \end{aligned}$$

Since the faulty functions are identical, the three faults are equivalent.

(c) Parallel fault simulation of the three faults is shown in the following figure. Faults F s-a-1 and H s-a-1 are detected and L s-a-0 is not detected.



Parallel fault simulation of the circuit of Figure 2.

Problem 4: Combinational ATPG (20 Points)

- (a) Give the truth table of a two-input exclusive-OR gate for five-valued signals, 0, 1, X , D , and \bar{D} . (10 points)
- (b) For improved diagnosis it is desired to derive tests such that a target fault is detected at one output but not at another output. The following construction is proposed for a two-output circuit under test (CUT) with outputs $Z1$ and $Z2$. Prove that if a target fault is detected at the output of the exclusive-OR

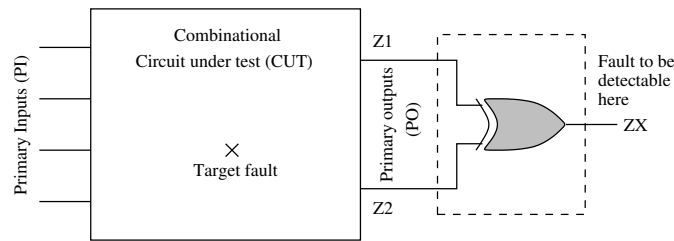


Figure 3: Proposed ATPG construction for diagnostic test in Problem 4(b).

gate then it will satisfy the specified diagnostic requirement for outputs $Z1$ and $Z2$ of CUT. (10 points)

Solution to Problem 4

- (a) Five-valued signal truth table of exclusive-OR gate:

Input 1	Input 2	Output
0	0	0
0	1	1
0	X	X
0	D	D
0	\bar{D}	\bar{D}
1	0	1
1	1	0
1	X	X
1	D	\bar{D}
1	\bar{D}	D
X	0	X
X	1	X
X	X	X
X	D	X
X	\bar{D}	X

Input 1	Input 2	Output
D	0	D
D	1	\bar{D}
D	X	X
D	D	0
D	\bar{D}	1
\bar{D}	0	\bar{D}
\bar{D}	1	D
\bar{D}	X	X
\bar{D}	D	1
\bar{D}	\bar{D}	0

- (b) Consider the entries in the truth table where both inputs have either a D or a \bar{D} . The corresponding output is either 0 or 1, as shown in the right

part of the table by four boldface output entries. This means that whenever the fault is detected simultaneously on both outputs $Z1$ and $Z2$ in Figure 3, the exclusive-OR gate will suppress the fault effect at output ZX . Because the ATPG attempts to observe the fault effect at ZX , such tests will not be generated.

From the truth table we further notice that whenever the fault effect appears at one input of the exclusive-OR gate and the other input has a deterministic 0 or 1 value, it propagates to the output either with or without an inversion. Thus, ATPG will find only those tests that detect the fault either at $Z1$ or at $Z2$ but not at both. If a fault can only be detected at both outputs simultaneously, no test will be found by the circuit of Figure 3. ■

Problem 5: Scan Design (20 Points)

The circuit in Figure 4 is a four-bit shift register. It has three primary inputs and one primary output. The circuit detects a 1111 bit-stream at $INPUT$ and inhibits $CLOCK$ signal. Thereafter the state of shift register remains frozen until a $CLEAR = 1$ input asynchronously clears the register to 0XXX state.

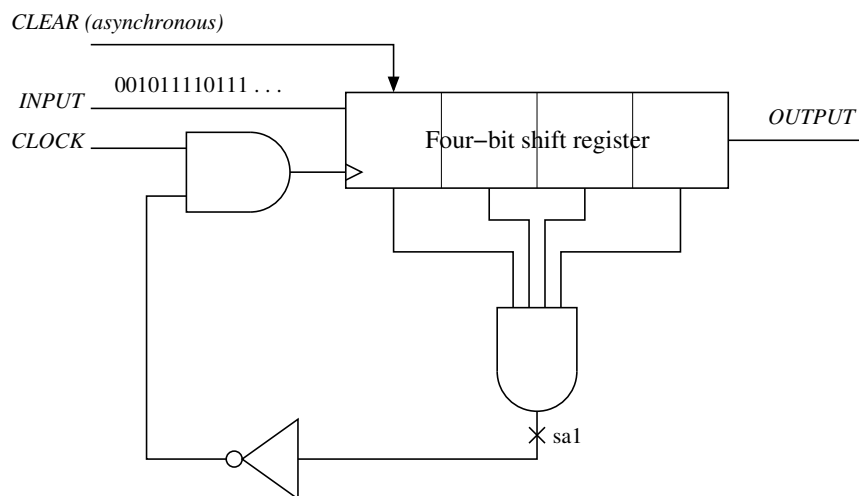


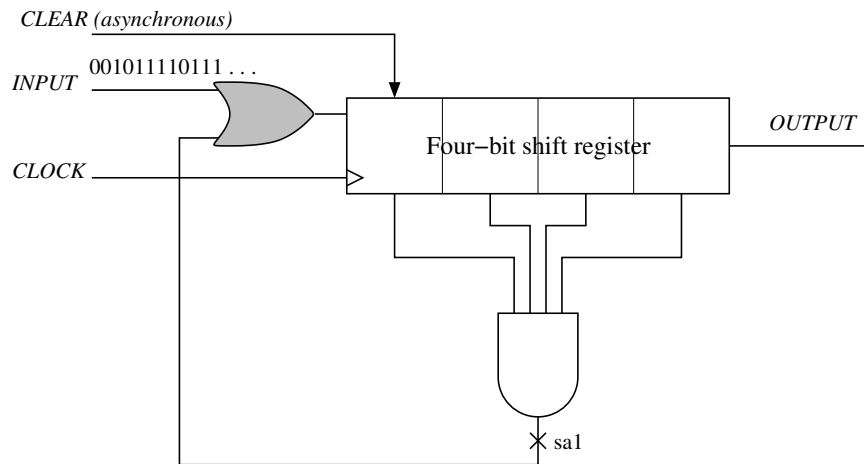
Figure 4: Circuit for scan design Problem 5.

- Is the s-a-1 fault at the four-input AND gate output detectable? (4 points)
- Specify the scan design rule that this circuit violates. Modify the circuit to make it scan compatible. Without inserting the scan hardware, is the s-a-1 fault at the output of the four-input AND gate in the modified circuit detectable? If yes, then specify that test. (8 points)

- (c) Further modify the circuit in part (b) for scan assuming that no modification in the shift register block is possible. (8 points)

Solution to Problem 5

- (a) We assume that all four flip-flops of the shift register are initially in the unknown (X) state. The output of the four-input AND gate will also be in the unknown state. When $CLEAR = 1$ is applied, the first flip-flop will change to 0, enabling $CLOCK$. Then all flip-flops can be initialized through the $INPUT$ signal. In the faulty circuit, clock cannot be enabled and, therefore, the shift register will stay in $0XXX$ state. Hence, the fault is only *potentially detectable*.
- (b) The circuit violates the rule, “clock must not be gated by an internal combinational signal.” The following schematic shows how this design rule violation can be corrected. Here, instead of disabling the clock, we stop the input data once the shift register gets to 1111 state.

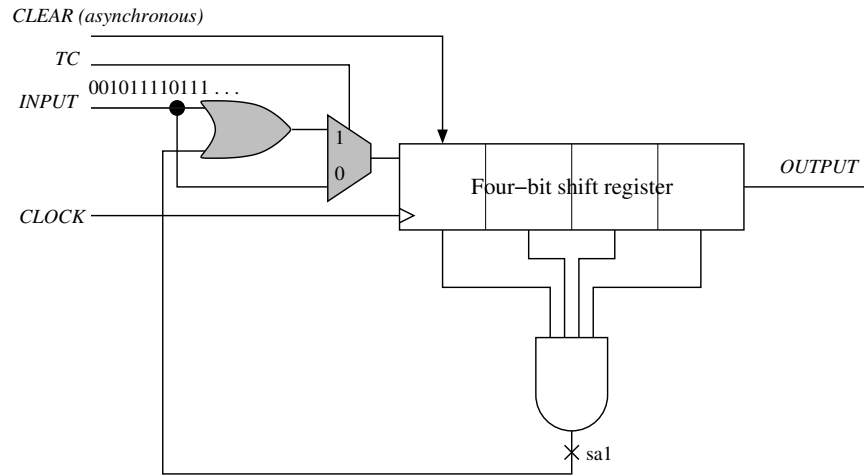


Circuit of Figure 4 modified to eliminate clock-gating.

Now both good and faulty circuits can be initialized. The following procedure will detect the specified s-a-1 fault:

1. Apply $CLEAR = 1$. This will set the shift register in $0XXX$ state and activate the fault as \overline{D} .
 2. Hold $CLEAR = 0$ and $INPUT = 0$, and clock the circuit four times. This will propagate \overline{D} to $OUTPUT$
- (c) The circuit is converted for scan testing by inserting a single multiplexer as shown in the following schematic. The multiplexer is controlled by a test control signal TC , which is a new primary input. The normal mode operation

($TC = 1$) is exactly the same as that of the previous schematic. In the scan mode ($TC = 0$) and *INPUT* is routed directly to the shift register and serves as the scan-in primary input. *OUTPUT* serves as the scan-out primary output.



Scan design of circuit of Figure 4. Only on test control (*TC*) pin is added.

Note that no modification is made to *clear*. This is because it is controllable as a primary input. In the scan mode, *clear* must be 0 (inactive). Had this signal not been a primary input, it will have to be disabled by *TC*. However, coverage of faults associated with any asynchronous signal like *clear* is not guaranteed by the scan test. Besides, these signals should be properly synchronized with clock to avoid timing or race problems during the scan test.