

ERRATA

LOGIC AND COMPUTER DESIGN FUNDAMENTALS – 2nd Edition Updated

M. Morris Mano and Charles R. Kime

© 2000 by Prentice-Hall, Inc.

The errata are given with reference to the Updated 2nd Edition of the book. The edition of your book is given on the cover and the printing of your book is the last number in the string of numbers below “Printed in ...” on the page following the title page. For example: 10 9 8 7 6 5 4 3 2 is printing 2. Only those errors believed to affect technical correctness or understanding are listed.

Chapter 2

Page 55 (Added 6/12/00).

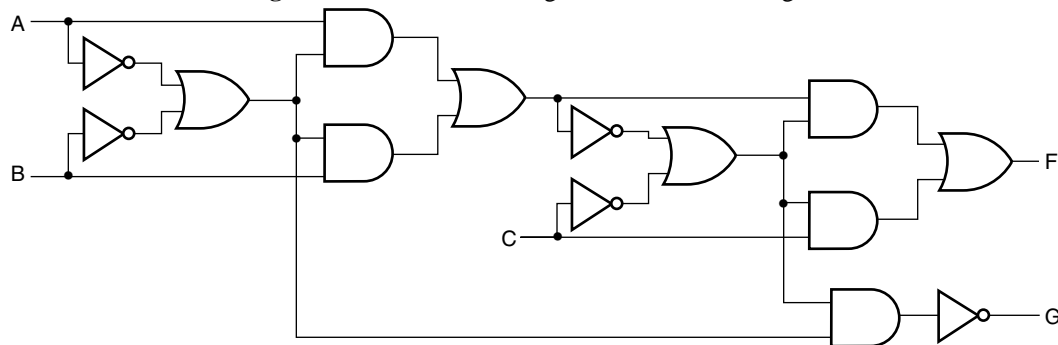
Line 11 from bottom. Change “21-5” to “2-15”

Page 87.

Problem 2-31. “2-25” should be “2-27” and “2-27” should be “2-29.”

Page 88.

Figure 2-46. Variables on right and left are missing. Should be:



Page 89.

Problem 2-41. “delay to find t_{PHL} .” should be “delay to find t_{pd} .”

Chapter 3

Page 133(Added 10/8/00).

Lines 11 and 15. “ $N - M$ ” should be “ $M - N + 2^n$ ”

Page 136(Added 10/8/00).

Line 10. Replace “ $-(M - N)$ ” with “ $-(N - M)$ ”

Page 162.

Figure 3-46. The “comment” designation for all of the line numbers on the right margin should be // instead of --.

Page 176.

Problem 3-47. “whether they are equal.” should be “whether B is greater than A.”

Page 177.

In paragraph marked with . “kime” should be “mano.”

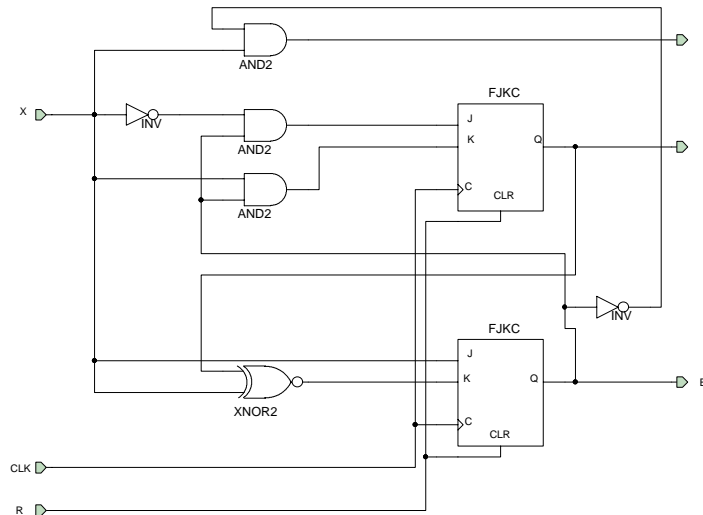
Page 177.

Problem 3-56. “given in Appendix A.” should be “given in the Companion Website Gallery.”

Chapter 4

Page 222.

Figure 4-28. Labels on flip-flops missing. Should be:



Page 235.

Line 4 from bottom. “`end case;`” should be “`endcase`”

Page 242.

Problem 4-17. Remove incorrect indentation.

Page 245.

Problem 4-31. “Table 4-6.” should be “Table 4-7.”

Page 246.

Problem 4-39. Line 3 - “when D was” should be “when EN was” and Line 6 - “while EN = 1.” should be “when EN becomes 1.”

Page 247.

Problems 4-41, 4-42, and 4-43. Delete the sentence “Use the D flip-flop ...”

Page 247.

Problem 4-46. Line 3 - “when D was” should be “when EN was” and Line 6 - “while EN = 1.” should be “when EN becomes 1.”

Page 247.

Problems 4-48, 4-49, and 4-50. Delete the sentence “Use the D flip-flop ...”

Chapter 5

Page 270 (Added 10/8/00).

Line 13 from bottom. “Problem 5-19” should be “Problem 5-21.”

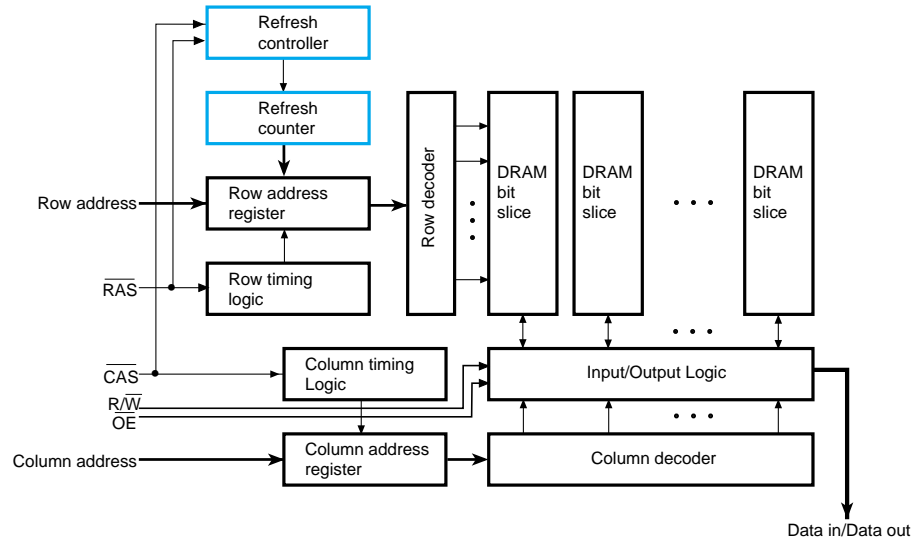
Page 283.

Problem 5-29. Add “*” before “Write ...”

Chapter 6

Page 303.

Figure 6-13. Outlines missing for Refresh Controller and Refresh Counter. Should be:



Page 310 (Added 02/11/01).

Line 3 from bottom. "PDL" should be "PLD."

Page 320 (Added 10/8/00).

Line 13 from bottom. m0, 1, 2, 4) should be m(0, 1, 2, 4).

Line 12 from bottom. m0, 5, 6, 7) should be m(0, 5, 6, 7).

Page 336.

Problems 6-15. Add "*" before "Obtain ..."

Page 337.

Problems 6-18. Add "*" before "Repeat ..."

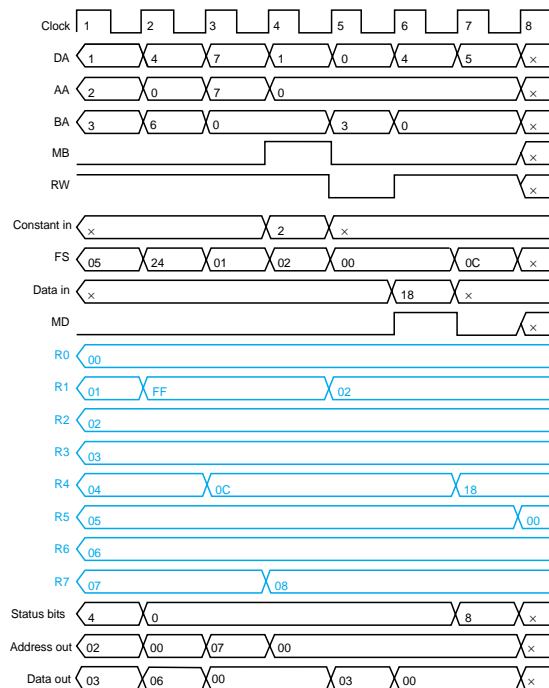
Chapter 7

Page 363 (Added 02/11/01).

Line 2. "Whenever C_{in} is 1, $Y + B$ has 1 added." should be "Whenever C_{in} is 1, $A + Y$ has 1 added."

Page 375.

Figure 7-20. Signal AA is 0 instead of 6 in clock cycle 2; signal BA is 6 instead of 0 in clock cycle 2. (BA remains 0 for clock cycles 3 and 4.) Should be:



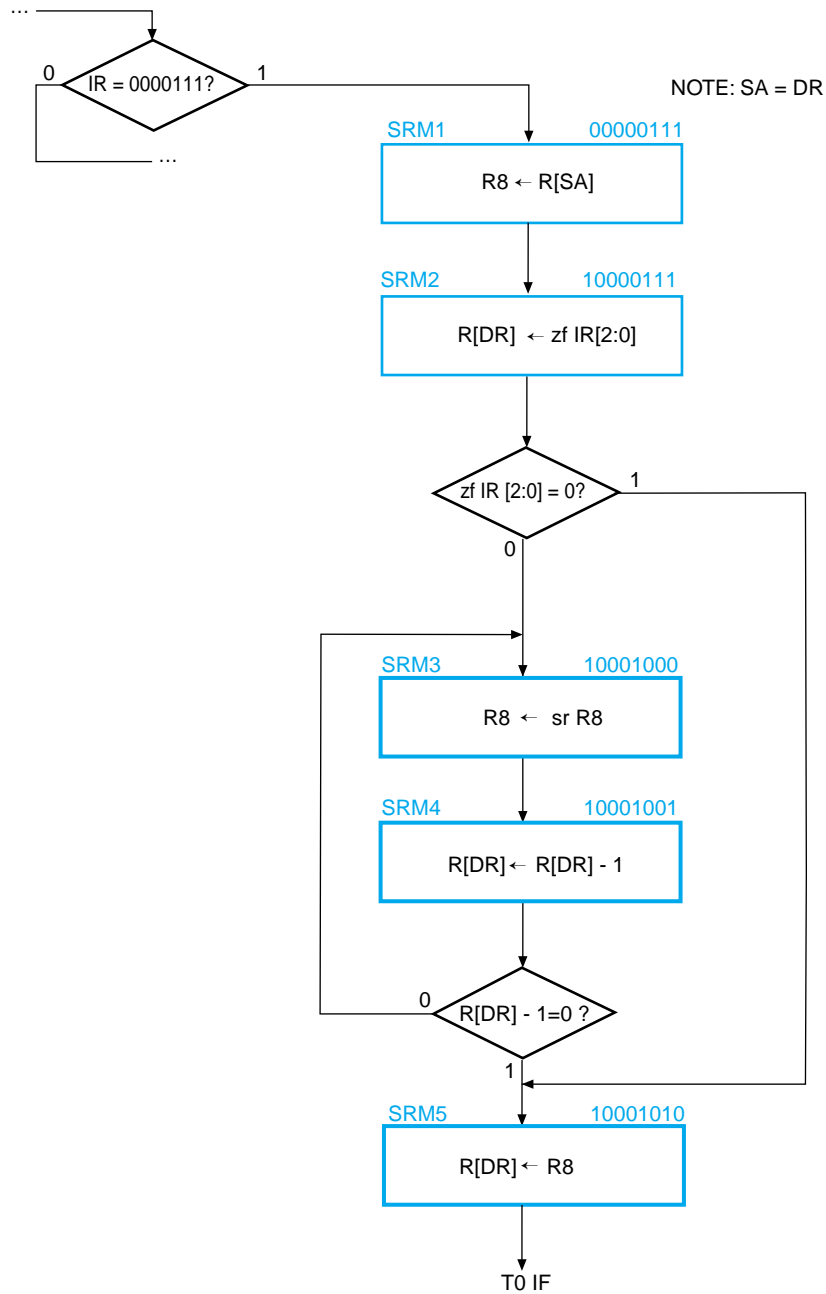
Chapter 8

Page 434 (Added 2/11/01).

Table 8-8. Row BRZ. Operation Code should be "1100011" instead of "1100000."

Page 445 (Added 2/11/01).

Line 1 bottom through page 447 line 5 and Figure 8-30. The ASM chart as originally given in the text ignores two factors: 1) decision blocks for the microarchitecture cannot go to two arbitrary next states, and 2) the register source for shift operations cannot be selected by SB which is the shift amount field in the instruction. The new Figure 8-30 and replacement text is given next.



For this operation, it is required that $DR = SA$. In state SRM1, the value to be shifted, originally in R[SA], is placed in R8 which can be addressed as an input to the shifter. Next, in state SRM 2, the shift amount in the last three bits of the IR is placed in R[DR] with the 13 positions to the left of the shift amount filled with 0's. At the same time, the shift amount passing through the Function Unit is tested to determine if it is 0. If it is, then the value in R8 is moved back to R[DR] and the state becomes IF completing the shift operation. If the shift amount is nonzero, then the next state is SRM3 which is entered by counting up the CAR. In SRM3, R8 is shifted one bit position to the right. In SRM4, the shift amount contained in R[DR] = R[SA] is decremented. At the same time, the decremented amount passing out of the Function Unit is tested to determine if it is 0. If it is, the state becomes SRM5 and the shifted result is moved from R8 to R[DR], completing the operation. If the decremented shift amount is non-zero, the state becomes SRM3 and a shift and decrement are repeated until the decremented shift amount become 0 at which time the contents of R8 is returned to R[DR] and the state become IF. The instruction requires $2s + 5$ clock cycles, where s is the number of positions shifted. The range of clock cycles required is from 5 to 19. In the same operation were performed by using a right-shift instruction s -times, then $3s$ instructions would be required. The improvement in the required number of clock cycles required is $s - 5$, so for shifts up through five bit positions, there is no improvement. For shift of 6 or 7 positions, 1 or 2 clock cycles are saved. Also, $s - 1$ fewer memory locations are required for storage for the SRM instruction, in contrast to s shift-right instructions. Clearly, the advantage for a maximum shift of seven positions is small, but this advantage will grow for larger shift amounts of 15 or 31 positions.

Page 463 (Added 2/11/01).

Problem 8-36. There is no ADD operation in Table 8-8. So change all ADD's to SUB's.

Page 465 (Added 2/11/01).

Problem 8-43. Figure "8-24" should be Figure "8-28."

Chapter 9

Page 476.

Figure 9-2. Box missing around "D." Use 0255MANO0902-2cor.eps from zip disk.

Page 509.

Problem 9-28(a). "2's complement" should be "signed 2's complement."

Chapter 10

Page 534 (Added 10/8/00).

Table 10-8. In rows beginning with XCH2 in the leftmost column, the ROM inputs column should contain:

MODE || S = XXX1

MODE || S = 0000

MODE || S = XX10

MODE || S = X1X0

MODE || S = 1XX0

Page 535 (Added 10/8/00).

Table 10-10. Replace table entries as follows in the row 1RL2: replace "0" with "3" in column MA, and, and replace "3" with "0" in column MB.

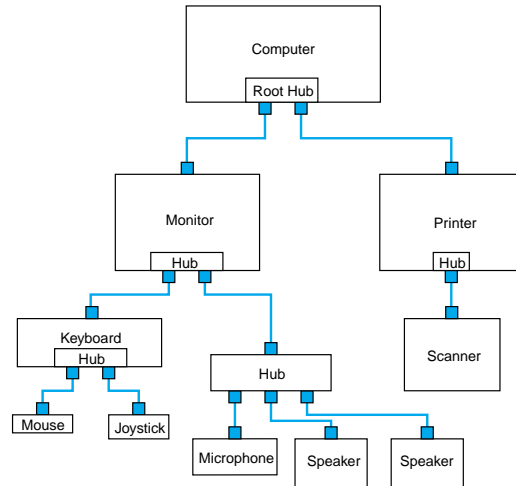
Page 557 (Added 02/11/01).

Figure 10-19. The gap between Bus B and the right input to the Adder should be closed.

Chapter 11

Page 591.

Figure 11-10. Connections missing between components. Should be:



If You Find Additional Errors

Please mail additional errors found or comments to:

kime@engr.wisc.edu

Thanks very much to the faculty and students sending in corrections to the first and second editions!

CRK

By M. Morris Mano, Charles R. Kime - Prentice Hall (2007) - Hardback - 678 pages - ISBN 013198926XFeaturing a strong emphasis on the fundamentals underlying contemporary logic design using hardware description languages, synthesis, and verification, this book focuses on the ever-evolving applications of basic computer design concepts with strong connections to real-world technology. Treatment of logic design, digital system design, and computer. design. Ideal for self-study by engineers and computer scientists. Find many great new & used options and get the best deals for Logic and Computer Design Fundamentals by Charles R. Kime and Mano (1996, Hardcover) at the best online prices at eBay! Free shipping for many products!Â Text will be unmarked and pages crisp. Satisfaction is guaranteed with every order. LOGIC AND COMPUTER DESIGN FUNDAMENTALS By Charles R. Kime
****Mint Condition**.**