Sr. No.	Page No	Line No	Original Text	New Text	Comment
1.	31	13	1-bit variables of types y1 and y2	1-bit variables of type wire, y1 and y2	
2.	41	25	Both of the blocking assignments	Both of the non-blocking assignments	
3.	46		else if (op_code == 2'b01;	else if (op_code == 2'b01)	Verilog example
4.	91	28	m>N	M>N	
5.	96	30	111xx	111xx	Need to be aligned by shifting to the left as done in multiplication
6.	96	37-38	This requires dropping of bits in the case of simple truncation	This may require dropping of bits but in this case no truncation is required	
7.	96	38	stored as 0_1010_00011	stored as 0_1011_00011	
8.	97	26	0.18125	01.8125	
9.	97	29	+1.9922	+1.9961	
10.	98	25	0x7FFF	0x7FF6	
11.	101	8	in Q2.6 format is 0.2813	in Q2.6 format is 0.28125	
12.	102	1	0111_011 in Q4.4 is 7.4375	0111_0111 in Q4.4 is 7.4375	Figure 3.11
13.	102	16	0111_0110 in Q4.4 format is 3.4375	0111_0111 in Q4.4 format is 3.4375	
14.	103	45	Q3.2 111 11 3.75	Q3.2 011 11 3.75	Original Figure 3.13 is correct
15.	104	1	Q2.2 0111 2.50	Q2.2 0111 1.75	
16.	104	3	Q2.2 0101 3.50	Q2.2 0101 1.25	
17.	111	31	Q10.6	Q6.10	
18.	111	33	Q10.6	Q6.10	
19.	111	38	<16, 10, SC_TRN, SC_SAT>	<16, 6, SC_TRN, SC_SAT>	
20.	112	15	fixpt = 0.25	fixpt = 0.3242	Table 3.6 (2nd Col.)
21.	112	16	fixpt = -0.25	fixpt = 0.2432	
22.	112	17	fixpt = 0	fixpt = 0.0889	
23.	112	18	fixpt = 0.75	fixpt = 0.7998	
24.	112	19	fixpt = -1	fixpt = -0.9004	
25.	129	9	Section 3.2.2	Section 3.2.3	
26.	139	45	y(m)=sink (y(i,j), y1(i,j), y2(i,j));	y(m)=sink (x(i,j), y1(i,j), y2(i,j));	
27.	155	34-38	$f_A^1 = f_B^1$ $f_B^1 = f_C^1$ $f_C^3 = f_A^1$	$f_A 1 = f_B 1$ $f_B 1 = f_C 1$ $f_C 3 = f_A 1$	The normal numbers are appearing as superscript
28.	194	6	Q1.15 format	//Q1.15 format	
29.	386		for(i=0;i<4;i++)	for(i=0;i<5;i++)	Two locations on the page.
30.	Figure 6.13		All arrows >>	<<	Multiplied by positive powers of 2
31.	Fig 13.21		No select line of mux	Select line	Data compression MUX select lines