Typos

1.1 Change on Page 225 in Section 17.7.8

Change

If signal burst_mode were to be maintained low until at least clock tick 10, the expression would result in a match as shown in Figure 17-12.

1.2 Example on Page 226, Section 17.7.9

Last example in Section 17.7.9

Change

!trdy[*7] within (($fell trdy) ##1 !trdy[*8])

1.3 Typo on Page 232 in Section 17.8

Change

1) When valid_in is true, x is assigned to pipe_in. Property e is true if five cycles later, pipe_out1 is equal to (x+1). Property e is false if pipe_out1 is not equal to (x+1).

1.4 Typo related to [* operator on Page 232-233 in Section 17.8

In this typo’s Replace * [ to [*

On Page 232 in the second example, change

```
sequence data_check;
  int x;
  a ##1 !a, x = data_in ##1 !b ##0:$] ##1 b && (data_out == x);
endsequence
property data_check_p
  int x;
  a ##1 !a, x = data_in |=> !b [*0:$] ##1 b && (data_out == x);
endproperty
```

On Page 233 in the second example, change

```
sequence sub_seq1;
  int v1;
  a ##1 !a, v1 = data_in ##1 !b [*0:$] ##1 b && (data_out == v1);
endsequence
```

On Page 233 in the third example, change

```
sequence sub_seq2(lv);
  a ##1 !a, lv = data_in ##1 !b [*0:$] ##1 b && (data_out == lv);
endsequence
```

On Page 234 in the fourth example, change

```
sequence sub_seq3(lv);
  int lv; // illegal since lv is a formal argument
  a ##1 !a, lv = data_in ##1 !b [*0:$] ##1 b && (data_out == lv);
```
1.5 Typo on Page 250 in Section 17.11.5, first paragraph

desalination destination

1.6 Change on Page 254-255 in Section 17.12.4

Change (last example)

```
module top(input bit clk);
  logic a,b,c;
  sequence seq3;
    @(posedge clk) b ##1 c;
  endsequence
  c1: cover property cover property(seq3);
  ...
endmodule
```

1.7 Change on Page 255 in Section 17.12.5

Change (First example)

```
always @(posedge clk) begin
  begin
    <statements>
    assert property assert property(rule);
  end
end
```

1.8 Examples on Page 256, Section 17.12.5

Change

```
property r3;
  @(posedge sm clk) (q != d);
endproperty
```

Change

```
property r3;
  @(posedge sclk mclk) (q != d);
endproperty
always @(posedge mclk) begin
  if (a) begin
    q <= d1;
    r3_p: assert property (r2 r3);
  end
end
```
The above example is equivalent to:

```verilog
property r3;
  @(posedge sclk mclk) a |-> (q != d);
endproperty
r3_p: assert property (r3);
always @(posedge mclk) begin
  if (a) begin
    q <= d1;
  end
end
```

Change

```verilog
property r4;
  @(posedge sclk mclk) (q != d);
endproperty
```

Change

```verilog
property r4;
  @(posedge sclk mclk) (a==1) |-> (q != d);
endproperty
```

1.9 Complete the example on Page 258, Section 17.13

Change

— default clock, for example

```verilog
default clocking master_clk @(posedge clk);
property p4; (a ##2 b);
endproperty
assert property (p4);
```

1.10 Change on Page 263-264 in Section 17.14.

Change (last example on pp 263); remove bold font from “enable”

```verilog
interface range (input clk,enable, input int minval,expr);
  property crange_en;
  @(posedge clk) enable enable|-> (minval <= expr);
endproperty
range_chk: assert property (crange_en);
endinterface
bind cr_unit range r1(c_clk,c_en,v_low,(in1&&in2));
```