12주차 결과보고서

전공: 컴퓨터공학과 학년: 2학년 학번: 20231523 이름: 김민정

**1. 2bit 2진 counter**

2-bit 2진 counter는 00->01->10->11->00 상태 전이를 계속 반복하는 counter이다. 만약 input값이 1이 나오면 counter의 역할을 계속 수행하지만, input값이 0이 나오면 reset이 발생한다. 이를 FPGA로 표현할 때 input값이 1일때를 reset값이 1일 때(low active pin이기 때문에 1일 때는 reset이 비활성화 된다.), input값이 0일 때를 reset값이 0일때로 설정하였다. 아래는 2bit 2진 counter을 구현하기 위한 과정을 서술한 것이다.

1) Verilog code

|  |  |
| --- | --- |
| Source code | Testbench code |
| `timescale 1ns / 1ps  module tbc(  input clk, r,  output [1:0] out  );    reg [1:0] out;  always@(posedge clk)  begin  if(!r)  out <= 0;  else  out <= out + 1;  end  endmodule | `timescale 1ns / 1ps  module tbc\_tb;  reg clk, r;  wire [1:0] out;  tbc u\_test(  .clk (clk ),  .r (r ),  .out (out )  );  initial r=1'b0;  initial clk=1'b1;  always r = #100 ~r;  always clk = #7 ~clk;  initial begin  #1000  $finish;  end  endmodule |

2) schematic

도표, 텍스트, 평면도, 기술 도면이(가) 표시된 사진

자동 생성된 설명

3) simulation

스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

**2. 4bit decade counter**

4-bit decade counter는 0->1->….->9->0의 상태 전이를 4개의 bit 표현으로 계속 반복하는 counter이다. 이도 역시 1번의 2bit 2진 counter처럼 input값이 1이 나오면 counter의 역할을 계속 수행하지만, input값이 0이 나오면 reset이 발생한다. 이를 FPGA로 표현할 때 input값이 1일때를 reset값이 1일 때(low active pin이기 때문에 1일 때는 reset이 비활성화 된다.), input값이 0일 때를 reset값이 0일때로 설정하였다. 아래는 4-bit decade counter를 구현하기 위한 과정을 서술한 것이다.

1) Verilog code

|  |  |
| --- | --- |
| Source code | Testbench code |
| `timescale 1ns / 1ps  module fbdc(  input clk, r,  output [3:0] out  );  reg [3:0] out;  always@(posedge clk)  begin  if(!r)  out<=0;  else if (out == 9)  out <= 0;  else  out <= out + 1;  end  endmodule | `timescale 1ns / 1ps  module fbdc\_tb;  reg clk, r;  wire [3:0] out;  fbdc u\_test(  .clk (clk ),  .r (r ),  .out (out )  );  initial r=1'b0;  initial clk=1'b1;  always r = #100 ~r;  always clk = #5 ~clk;  initial begin  #1000  $finish;  end  endmodule |

2) schematic

도표, 텍스트, 평면도, 라인이(가) 표시된 사진

자동 생성된 설명

3) simulation

스크린샷이(가) 표시된 사진

자동 생성된 설명

**3. 4bit 2421-decade counter**

4-bit 2421 decade counter도 역시 위의 2번 문항의 counter와 같이 0->1->….->9->0의 상태 전이를 4개의 bit 표현으로 계속 반복하는 counter이다. 하지만 2번 문항의 counter은 비트 표현이 BCD 코드였다면 3번문항의 4-bit 2421 decade counter는 2421 code로 표현하게 된다. 이도 역시 위의 counter처럼 input값이 1이 나오면 counter의 역할을 계속 수행하지만, input값이 0이 나오면 reset이 발생한다. 이를 FPGA로 표현할 때 input값이 1일때를 reset값이 1일 때(low active pin이기 때문에 1일 때는 reset이 비활성화 된다.), input값이 0일 때를 reset값이 0일때로 설정하였다. 아래는 4-bit decade counter를 구현하기 위한 과정을 서술한 것이다.

1) Verilog code

|  |  |
| --- | --- |
| Source code | Testbench code |
| `timescale 1ns / 1ps  module fb2421dc(  input clk, r,  output [3:0] out  );  reg [3:0] out;  always@(posedge clk)  begin  if(!r)  out <= 0;  else if (out ==4)  out <=11;  else if (out ==15)  out <=0;  else  out <= out +1;  end  endmodule | `timescale 1ns / 1ps  module fb2421dc\_tb;  reg clk, r;  wire [3:0] out;  fb2421dc u\_test(  .clk (clk ),  .r (r ),  .out (out )  );  initial r=1'b0;  initial clk=1'b1;  always r = #100 ~r;  always clk = #5 ~clk;  initial begin  #1000  $finish;  end  endmodule |

2) schematic

도표, 라인, 평면도, 기술 도면이(가) 표시된 사진

자동 생성된 설명

3) simulation

스크린샷이(가) 표시된 사진

자동 생성된 설명

**4. 결과 검토 및 논의 사항**

Counter에 대해 알아보면서, 이번 실험에서부터 clock을 처음 사용하는 실습을 진행했다. 이를 통해 rising edge일 때 장치가 특정 동작을 수행할 수 있게 코드를 짤 수 있게 되었다. 이번 FPGA를 짤 때에는 rising edge를 기준으로 코드를 작성하였다. 하지만 falling edge일때도 똑같은 기능을 수행할 수 있게 장치를 구성할 수 있는데 이는 5번 문항에서 4bit decade counter에 적용해보도록 하겠다.

**5. 4bit decade counter ( + falling edge)**

Falling edge를 작동하게 하는 법은 단순하다 design source에서 posedge를 negedge로 변경

하면 된다. 아래는 falling edge를 적용한 4bit decade counter의 simulation 사진이다.

1) simulation

스크린샷이(가) 표시된 사진

자동 생성된 설명