7주차 결과보고서

전공: 컴퓨터공학과 학년: 2학년 학번: 20211533 이름: 남정연

**1.**

Even Parity bit는 패리티 비트를 포함한 전체 비트의 1의 개수를 짝수로 만드는 패리티 비트이다. 따라, generator의 경우에는 1의 개수가 홀수일 경우 1을, 짝수일 경우 0을 반환해야 하므로 XOR 연산을 사용하고, checker의 경우에도 또한 1의 개수가 홀수일 경우 1을(오류), 짝수일 경우 0을(정상) 반환해야 하므로 동일하게 XOR 연산을 사용한다.

Generator의 논리식은 e = a XOR b XOR c XOR d 이며, 아래와 같은 시뮬레이션 결과, 진리표 및 카르노 맵을 가진다.

시뮬레이션 결과

Graphical user interface

Description automatically generated

진리표

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b | c | d | E |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 |

카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ab | cd | 00 | 01 | 11 | 10 |
| 00 | 0 | 1 | 0 | 1 |
| 01 | 1 | 0 | 1 | 0 |
| 11 | 0 | 1 | 0 | 1 |
| 10 | 0 | 1 | 1 | 0 |

Checker의 논리식은 e = a XOR b XOR c XOR d 이며, 아래와 같은 시뮬레이션 결과, 진리표 및 카르노 맵을 가진다.

시뮬레이션 결과

Graphical user interface

Description automatically generated

진리표

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b | c | d | e |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 |

카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ab | cd | 00 | 01 | 11 | 10 |
| 00 | 0 | 1 | 0 | 1 |
| 01 | 1 | 0 | 1 | 0 |
| 11 | 0 | 1 | 0 | 1 |
| 10 | 0 | 1 | 1 | 0 |

**2.**

Odd Parity bit는 패리티 비트를 포함한 전체 비트의 1의 개수를 홀수로 만드는 패리티 비트이다. 따라, generator의 경우에는 1의 개수가 짝수일 경우 1을, 홀수일 경우 0을 반환해야 하므로 XOR 연산을 사용하고, checker의 경우에도 또한 1의 개수가 짝수일 경우 1을(오류), 홀수일 경우 0을(정상) 반환해야 하므로 동일하게 XNOR 연산을 사용한다.

Generator의 논리식은 e = NOT(a XOR b XOR c XOR d) 이며, 아래와 같은 시뮬레이션 결과, 진리표 및 카르노 맵을 가진다.

시뮬레이션 결과

Chart

Description automatically generated

진리표

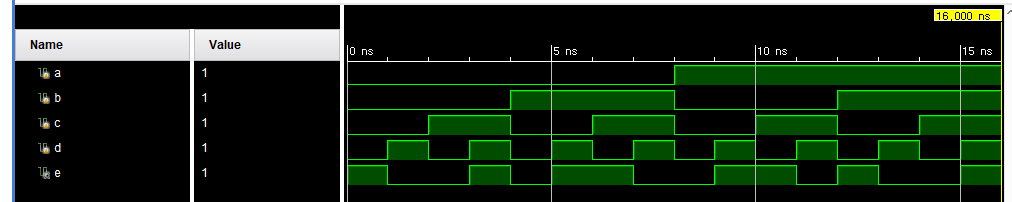
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b | c | d | e |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ab | cd | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 1 | 0 |
| 01 | 0 | 1 | 0 | 1 |
| 11 | 1 | 0 | 1 | 0 |
| 10 | 1 | 0 | 0 | 1 |

Checker의 논리식은 e = NOT(a XOR b XOR c XOR d) 이며, 아래와 같은 시뮬레이션 결과, 진리표 및 카르노 맵을 가진다.

시뮬레이션 결과



진리표

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b | c | d | e |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

카르노맵

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ab | cd | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 1 | 0 |
| 01 | 0 | 1 | 0 | 1 |
| 11 | 1 | 0 | 1 | 0 |
| 10 | 1 | 0 | 0 | 1 |

**3.**

2-bit binary comparator는 2 bit로 이루어진 두 개의 정수를 비교하는 비교기로, 두 정수를 A(a, b), B(c, d)라고 할 때 A>B일 때만 1을 출력하는 f1과 A=B일 때만 1을 출력하는 f2, A<B일 때만 1을 출력하는 f3에 대해서 아래와 같은 시뮬레이션 결과, 진리표, 카르노맵 및 Schematic을 가진다.

시뮬레이션 결과

Chart

Description automatically generated with medium confidence

진리표

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| a | b | c | d | f1 | f2 | f3 |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 1 | 0 |

f1에 대한 카르노맵

SOP = bc'd' + ac' + abd'

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ab | cd | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 0 | 0 |
| 01 | 1 | 0 | 0 | 0 |
| 11 | 1 | 1 | 0 | 1 |
| 10 | 1 | 1 | 0 | 0 |

f2에 대한 카르노맵

SOP = a'b'c'd' + a'bc'd + ab'cd' + abcd

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ab | cd | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 0 | 0 |
| 01 | 0 | 1 | 0 | 0 |
| 11 | 0 | 0 | 1 | 0 |
| 10 | 0 | 0 | 0 | 1 |

f3에 대한 카르노맵

SOP = a'b'd + a'c + b'cd

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ab | cd | 00 | 01 | 11 | 10 |
| 00 | 0 | 1 | 1 | 1 |
| 01 | 0 | 0 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 1 | 0 |

Schematic

Diagram, schematic

Description automatically generated

**4.**

Even Party Bit Generator와 Checker, Odd Party Bit Generator와 Checker에 대해서 각각 1의 개수이가 홀수일 때 1, 짝수일 때 1이 출력되는 결과가 나타났고 이는 올바른 결과로 Verilog 코딩 및 시뮬레이션이 정확했음을 알 수 있었다.

2 Bit Comparator의 경우에는 두 개의 정수를 비교함에 있어 f1은 입력 비트 a, b에 해당하는 정수가 더 클 경우에, f2는 입력 비트 a, b에 해당하는 정수가 입력 비트 c, d에 해당하는 정수와 같을 경우에, 마지막으로 f3는 입력 비트 c, d에 해당하는 정수가 더 클 경우에 1이 정확하게 출력되어 Verilog 코딩 및 시뮬레이션이 정확했음을 알 수 있었다.

**5.**

4 Bit Comparator는 2 Bit Comparator와 유사하게 숫자 A를 나타내는 a,b,c,d 비트와 숫자 B를 나타내는 e,f,g,h 비트를 입력으로 받아 A>B, A=B, A<B를 나타내는 f1, f2, f3를 각각 출력하는 비교기이다. 4 Bit Comparator를 연결하여 4n 비트 비교기를 만들 수 있는데, 각각의 비교기는 추가 입력으로 이전 단계의 f1, f2, f3를 입력으로 받는다.