디지털 논리회로2 프로젝트 제안서

ALU with Multiplier

학 과: 컴퓨터정보공학부

담당교수: 이형근 교수님

실습분반: 월요일 0, 1, 2

학 번: 2018202046

성 명: 이준휘

1. Title & Object
   1. Title

ALU with Multiplier

* 1. Object

해당 프로젝트를 통해 기존에 만들었던 ALU에서 더욱 확장하여 Multiplier를 추가할 수 있다. 또한 추가한 Multiplier를 Bus를 통해 Memory와 연결하여 데이터를 연산 및 저장을 할 수 있다. Testbench를 통해 명령을 내리고 이를 확인할 수 있다.

1. Component concept
   1. ALU with Multiplier

해당 모듈은 크게 데이터를 저장하는 레지스터 부분과 연산을 담당하는 모듈 부분으로 나눌 수 있다.

연산을 담당하는 모듈에서는 operand 1개 또는 2개를 사용하여 32bit의 연산을 수행하고 이를 result1 register에 저장한다. 만약 곱셈 연산을 진행하는 경우 operand1 과 2를 연산하여 해당 결과를 result1과 result2에 둘 다 저장한다. 연산의 종류로는 NOT 연산 2개, AND, OR, XOR, XNOR, 비교 연산 2개, LSL, LSR, ASR, 덧셈, 뺄셈, 곱셈 연산으로 이루어져있다.

기본적으로 해당 module 내의 모든 register는 32bit의 크기를 가진다. 레지스터에는 2개의 피연산자를 가진 레지스터, 연산자의 정보를 가진 레지스터, 결과 2개를 가지고 있는 레지스터(이 중 하나는 곱셈 연산을 진행할 때에만 사용된다. 시작을 확인하는 레지스터, 해당 연산일 끝났음을 확인하는 레지스터, 그리고 모든 수를 초기화할 수 있는 레지스터로 이루어져있다.

* 1. Bus

해당 모듈은 testbench로부터 데이터를 받아 이를 ALU with Multiplier 또는 Memeory에 저장하는 역할을 수행하는 bus이다. 해당 모듈에서는 1개의 마스터만을 가지고 있음으로 arbiter를 필요로 하지 않는다. 하지만 2개의 slave를 가지고 있기 때문에 주어진 메모리 주소를 토대로 주소의 범위에 따라 연결하는 register를 구분하는 address decoder를 만들어 준다. 또한 decoder를 통해 만들어진 신호를 통해 slave에서 다시 받아온 신호 중 보냈던 곳을 구분하여 데이터를 받는 역할을 수행한다.

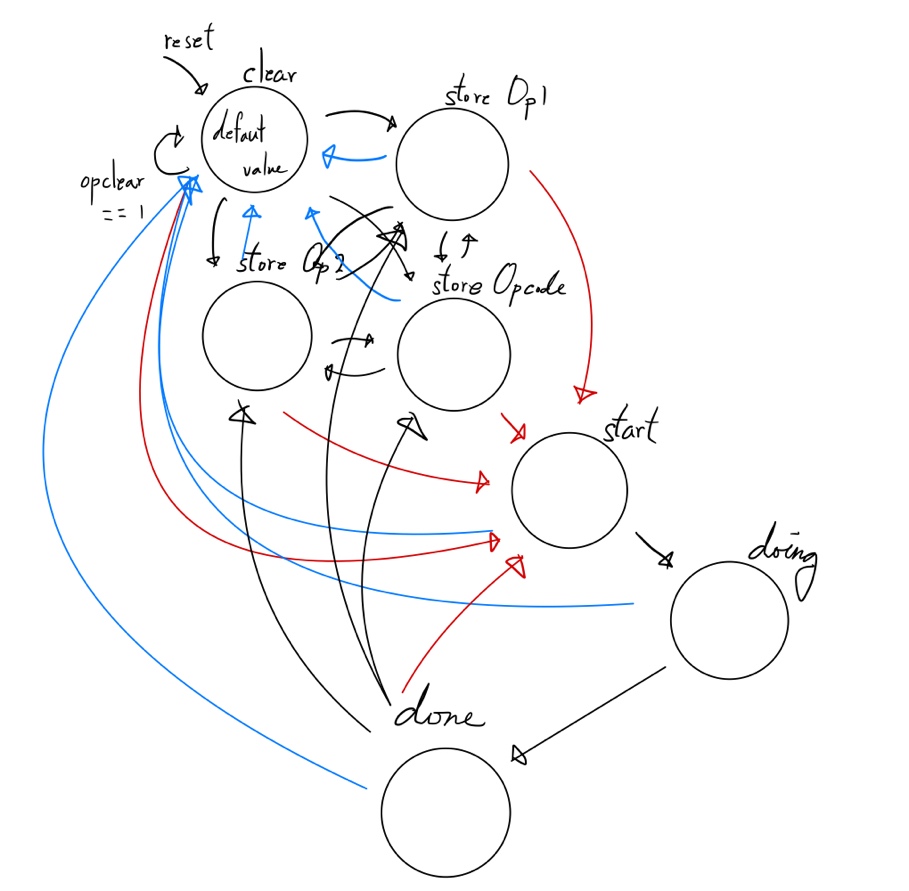
* 1. Memory

Memory에서는 데이터를 받아 저장하고 저장 중인 데이터를 꺼내는 역할을 수행한다. 해당 주소는 5bits이며 각각은 32bit의 정보를 저장한다. 저장한 데이터는 cen 신호나 wen 신호를 통해 익거나 쓸 수 있다. Cen이 활성화 될 경우에만 읽거나 쓰기 동작을 수행하고 이외의 경우에는 0을 출력한다. 만약 cen만 활성화된 경우에는 읽기 동작을, wen도 활성화된 경우에는 쓰기 동작을 수행한다.

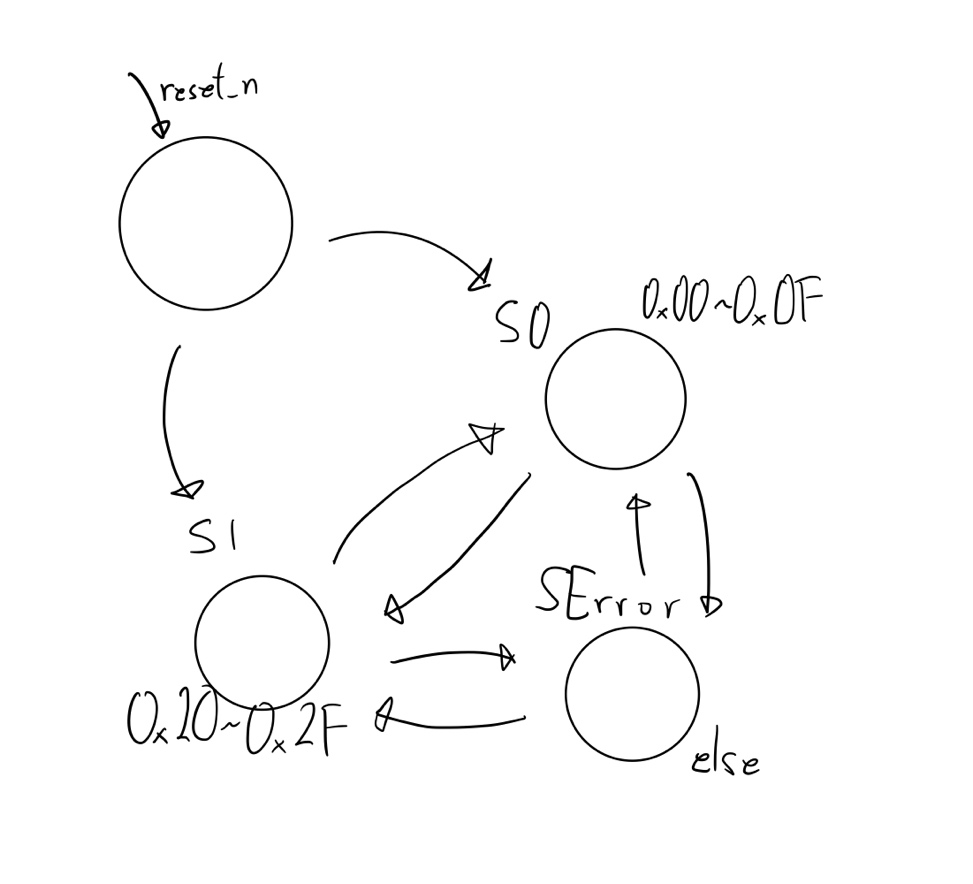
1. Schedule

|  |  |  |  |
| --- | --- | --- | --- |
|  | 11주차 | 12주차 | 13주차 |
| 제안서 작성 |  |  |  |
| 프로젝트 수행 |  |  |  |
| 오류 검증 |  |  |  |
| 보고서 작성 |  |  |  |

1. State transition diagram
   1. ALU with Multiplier



* 1. Bus

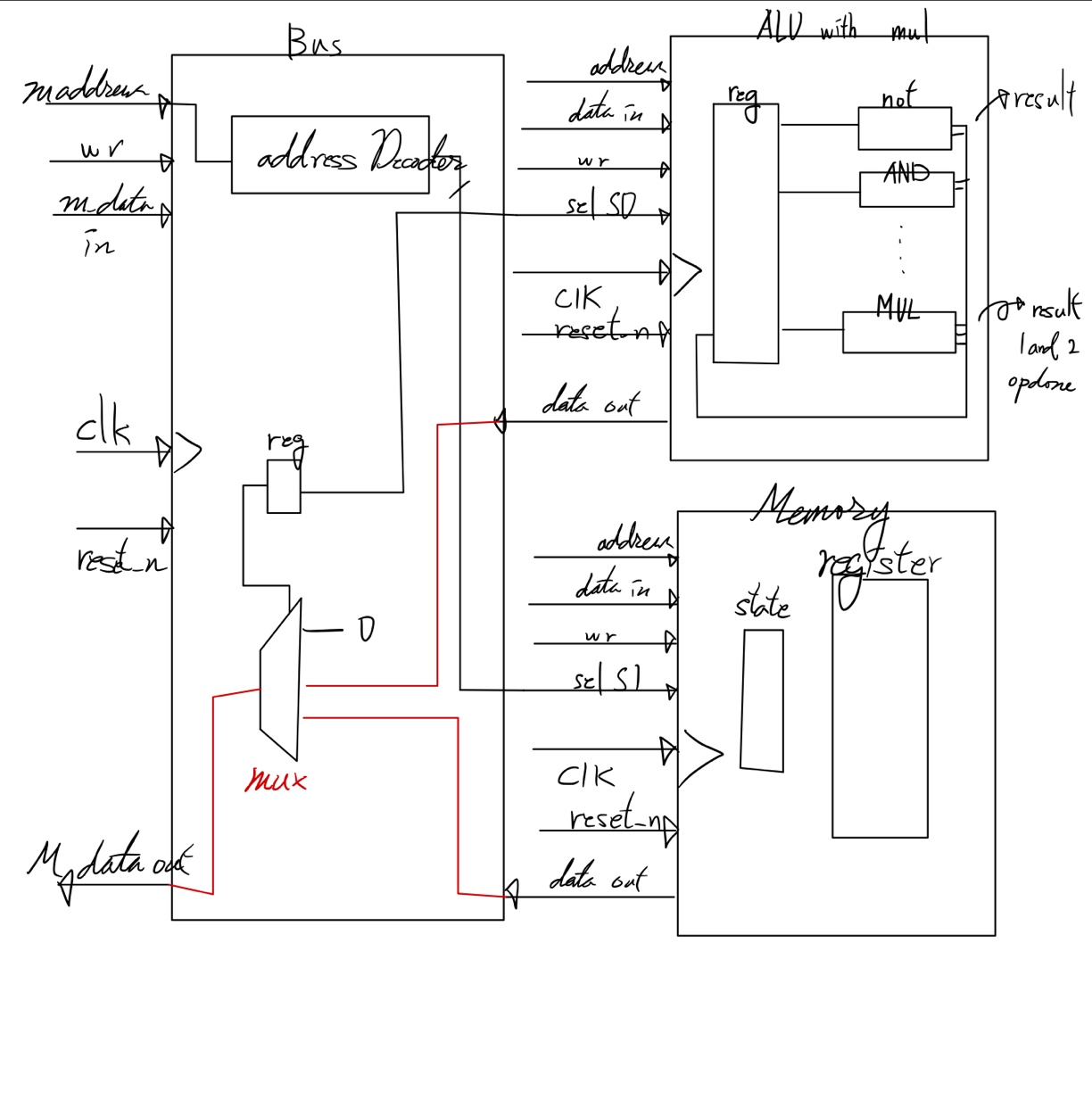


* 1. Memory

화살이(가) 표시된 사진

자동 생성된 설명

1. Module instance design



1. Design verification strategy

해당 프로젝트는 각각의 모듈별로 testbench를 진행한 후 정상적으로 작동되면 다음 모듈을 제작하고 검증하는 방식으로 진행할 것이다. 우선 ALY with Mulitplier의 모듈에서 Gate 연산 및 MUL 연산을 진행하는 모듈을 작성하고 해당 모듈이 정상적인 값을 출력하는지 확인한다. 그 후 ALU with Multiplier를 작성하고 주소에 따라 값이 정상적으로 register에 들어가고 데이터를 읽어올 수 있는지 확인한다. 해당 작업까지 정상적으로 완성되었을 경우 Memory를 제작한다. Memory는 기존 수행했던 과제에서 약간의 변형을 거쳐 모듈을 작성하기 때문에 간단한 확인만 수행한다. 해당 작업을 마친 후 BUS를 작성한다. 해당 Bus 모듈 또한 기존의 모듈을 활용하여 제작하며 약간의 변형만을 하기 때문에 간단한 입출력만 검증한다. 그 후 Bus ALU Ram을 연결하여 하나의 모듈로 제작하고 해당 모듈을 테스트한다. 해당 모듈에서는 Bus와 Memory, Bus와 ALU가 정상적으로 연결되었는지를 중점으로 테스트한다.