

5555555555

# 컴퓨터 공학 기초 실험2 보고서

실험제목: Register File

실험일자: 2023년 10월 13일 (금)

제출일자: 2023년 10월 28일 (토)

학 과: 컴퓨터공학과

담당교수: 이형근 교수님

실습분반: 금요일 0, 1, 2

학 번: 2020202037

성 명: 엄정호

## 1. 제목 및 목적

### A. 제목

Register File

### B. 목적

이전 실습에서 만든 32bit 레지스터를 이용한 읽기/쓰기 기능이 가능한 32bit \* 8 reg file 을 설계 및 구현한다.

## 2. 원리(배경지식)

Register file : 레지스터 파일이란 중앙처리장치에 내부에 위치한 작은 메모리 공간이다. cpu에서 사용되는 여러 레지스터들의 집합체로 구성 되어있다. 레지스터 파일의 기능은 다음과 같다

1. 데이터저장 : 데이터를 일시적으로 저장하는 장소로 사용된다. Cpu가 연산을 수행할 때 필요한 데이터나 중간 계산 결과를 레지스터 파일에 저장하여 빠르게 접근 할 수 있다.
2. 데이터 조작 : 연산에 필요한 데이터를 조작하고 변형하는 데 사용된다. 덧셈, 뺄셈, 곱셈, 나눗셈 등의 연산을 위해 데이터를 레지스터에서 불러오고 연산 후에 결과를 다시 레지스터에 저장 할 수 있다.
3. 데이터전송 : 데이터를 메모리나 입출력 장치와 주고받는 데 사용된다. 데이터를 레지스터에 로드하거나 레지스터에서 데이터를 읽어 외부 장치와 소통할 때 중간 저장소로 활용된다.

Stack : 가장 마지막에 들어온 데이터가 가장 먼저 나가는 후입선출 특성을 가지는 자료 구조이다. 메모리에 새로 들어오는 데이터의 위치가 메모리 마지막 top위치에 저장 되고, 출력되는 데이터 또한 top위치에 존재하는 데이터가 출력하게 된다. Push와 pop 두 가지 연산을 지원한다.

Pop: 스택에 데이터를 추가하는 연산이다. 스택의 top에 데이터를 추가한다.

Push : 스택에서 데이터를 제거하는 연산이다. 스택의 맨 위에 있는 데이터를 제거하고 반환한다.

스택은 주로 재귀 알고리즘, 자료호출, 함수호출 등에서 사용된다.

Queue : 데이터를 저장하는 자료 구조중 하나로 데이터의 삽입과 삭제가 서로 다른 공간에서 이루어지는 구조이다.

Equeue : 큐에 데이터를 추가하는 연산이다. 데이터는 큐의 뒤쪽 rear에 삽입된다.

Dqueue : 큐에서 데이터를 제거하는 연산이다. 데이터는 큐의 앞쪽 front에서 제거된다.

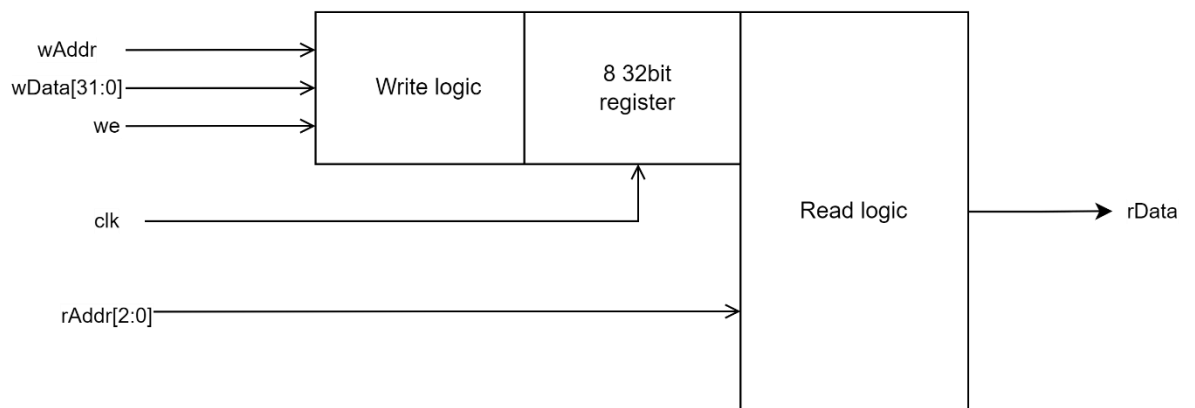
즉 가장 먼저 들어온 데이터가 제거되는 방식이다.

자원공유, 네트워크 데이터 관리, 프린터 대기열, 작업 스케줄링 등에 사용된다.

큐는 배열이나 연결 리스트에도 사용될 수 있다. 배열을 사용하게 될 경우 큐의 크기를 고정 시킬 수 있지만 연결리스트이 사용의 경우 큐의 크기를 동적으로 지정 할 수 있다.

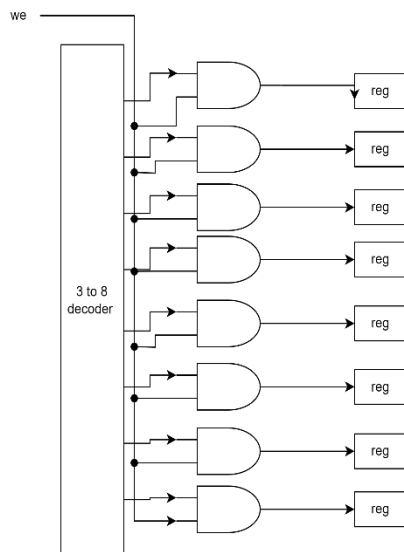
### 3. 설계 세부사항

#### Register file



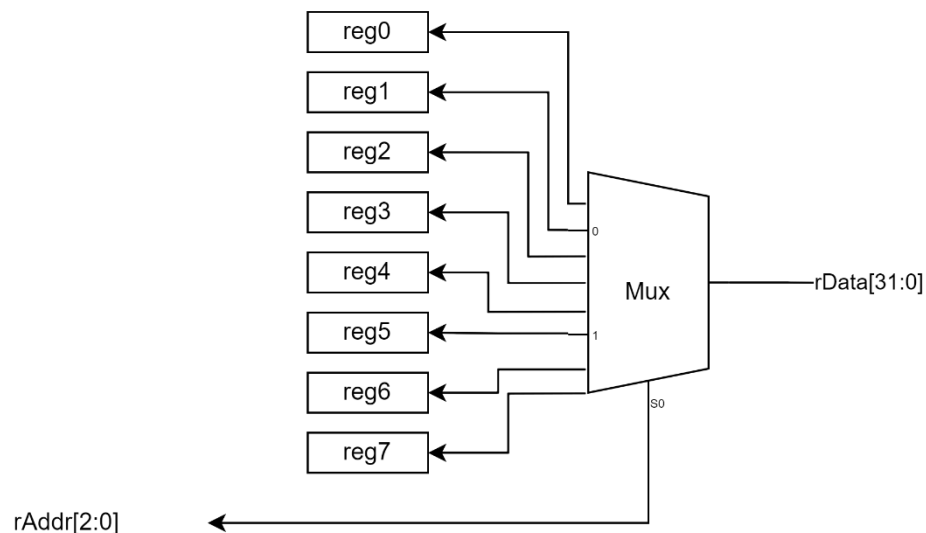
Register file은 32bit register에는 32bit 레지스터 8개가 인스턴스 되어 구성된다. Write 인스턴스는 we(write enable)에 의해 활성화 된다.

#### Write opreation



Write operation은 decoder를 통해 address를 해석하여 해당 레지스터를 사용한다. Read operation은 mux를 통해 8개의 레지스터중 하나를 선택하게 된다. Write operation의 경우 waddr를 이용해 we가 1일 경우 레지스터에 값이 입력된다. 선택된 레지스터는 write logic에 의해 사용자로부터 data를 저장한다.

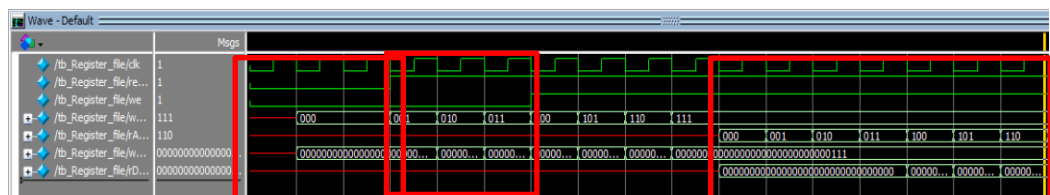
Read operation



Mux를 통해 read address로 8개의 레지스터 아웃풋 중 하나를 선택하게 된다.

#### 4. 설계 검증 및 실험 결과

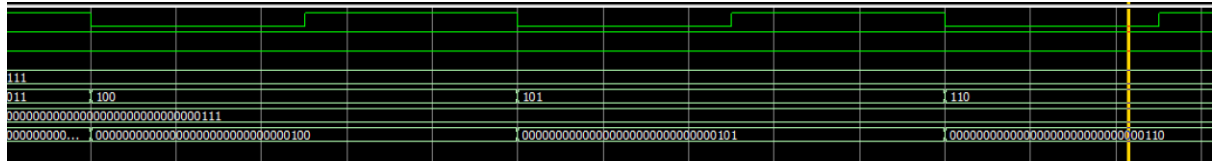
##### A. 시뮬레이션 결과



Reset이 0일 때 레지스터에 값을 저장하지 않고 0으로 초기화.

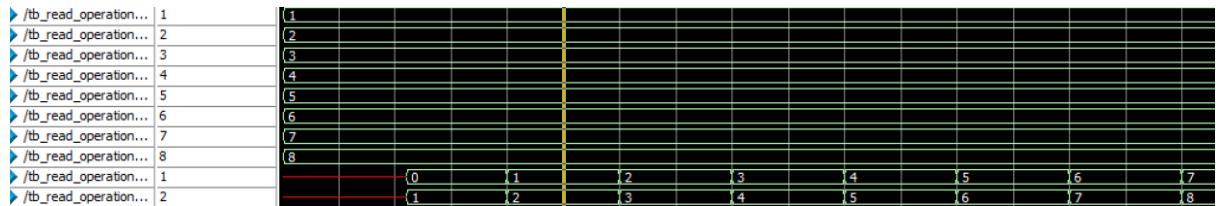
Enable이 0일 때 레지스터에 값을 저장하지 않는다.

100레지스터부터 값이 저장된 것을 확인 할 수 있다



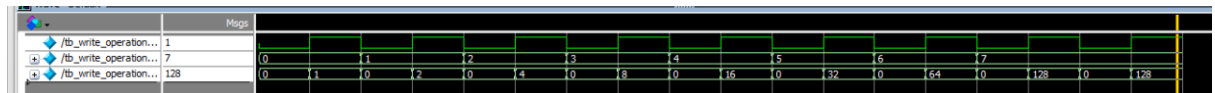
값이 정상적으로 전달된 것을 확인 할 수 있다.

## Read\_operation



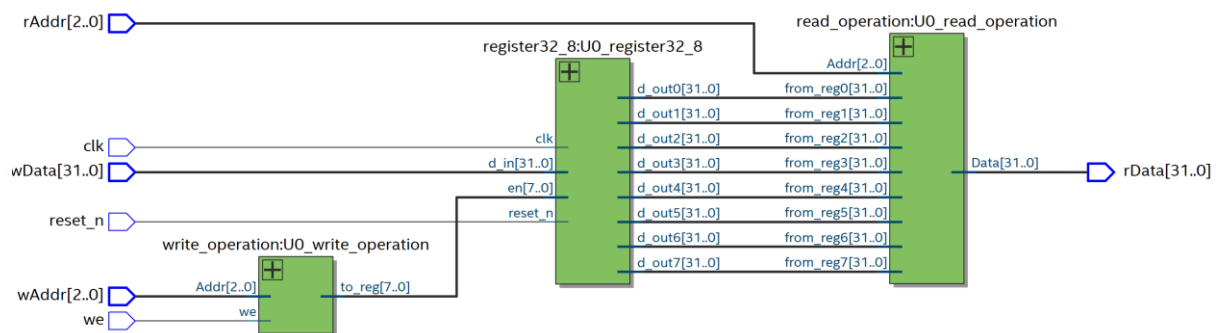
Addr에 따라 각 레지스터별로 저장된 값이 출력 되는 것을 확인 할 수 있었다.

## Write\_operation



En이 0일땐 0이 출력되고 0이 아닐땐 addr가 가리키는 비트가 출력된다.

## B. 합성(synthesis) 결과



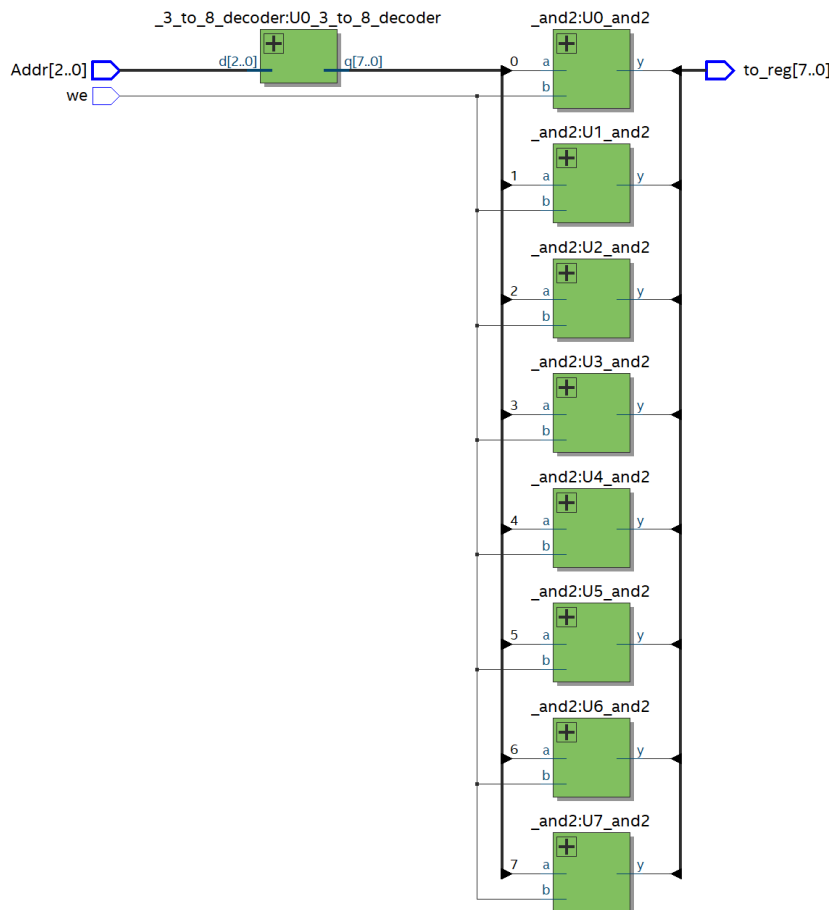
## Flow Summary

<<Filter>>

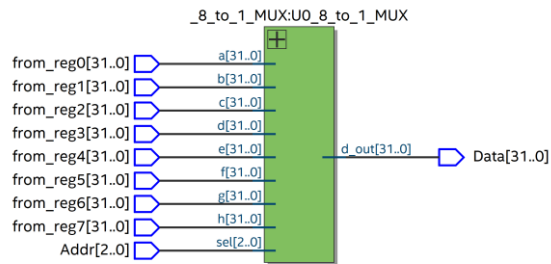
Flow Status	Successful - Fri Oct 27 09:22:24 2023
Quartus Prime Version	18.1.0 Build 625 09/12/2018 SJ Lite Edition
Revision Name	Register_file
Top-level Entity Name	Register_file
Family	Cyclone V
Device	5CSXFC6D6F31C6
Timing Models	Final
Logic utilization (in ALMs)	N/A
Total registers	256
Total pins	73

rAdd3 + wAdd3 + clk + reset + we + wData 32 + rData 32 total 63개의 pin사용

write\_operation



## Read\_operation



## 5. 고찰 및 결론

### A. 고찰

이번 실험에서는 traffic light나 shifter와 달리 별다른 문제 없이 실험이 진행되었다. 하지만 테스트 벤치 분석하면서 en과 reset이 작동할 때 don't care값이 계속 출력되어 테스트 벤치 분석에서 어려움을 겪었지만 제대로 작동한 것을 확인했다.

### B. 결론

레지스터와 멀티플렉서, 디코더를 이용한 레지스터 파일을 구성해 보았다. 처음 실험 진행시 write기능에서도 멀티플렉서를 사용 가능하지 않을까 생각했지만 여러 개의 인풋에서 하나의 를 선택하는 것과 여러 개의 아웃풋 중 하나만 1을 내보내는 것의 차이 라는 것을 알게 되었다. 또한 이번 실험을 통해 레지스터에 여러 비트의 값이 어떻게 저장 되는지 알 수 있었다.

## 6. 참고문헌

레지스터 파일/[https://en.wikipedia.org/wiki/Register\\_file](https://en.wikipedia.org/wiki/Register_file)

이준환/디지털논리회로2/광운대학교/2023

이형근/컴퓨터공학기초실험2/광운대학교/2024

최상호/데이터구조설계/광운대학교/2023