**2024 Spring CSED311 Lab 5 Report**

Team ID: 67735

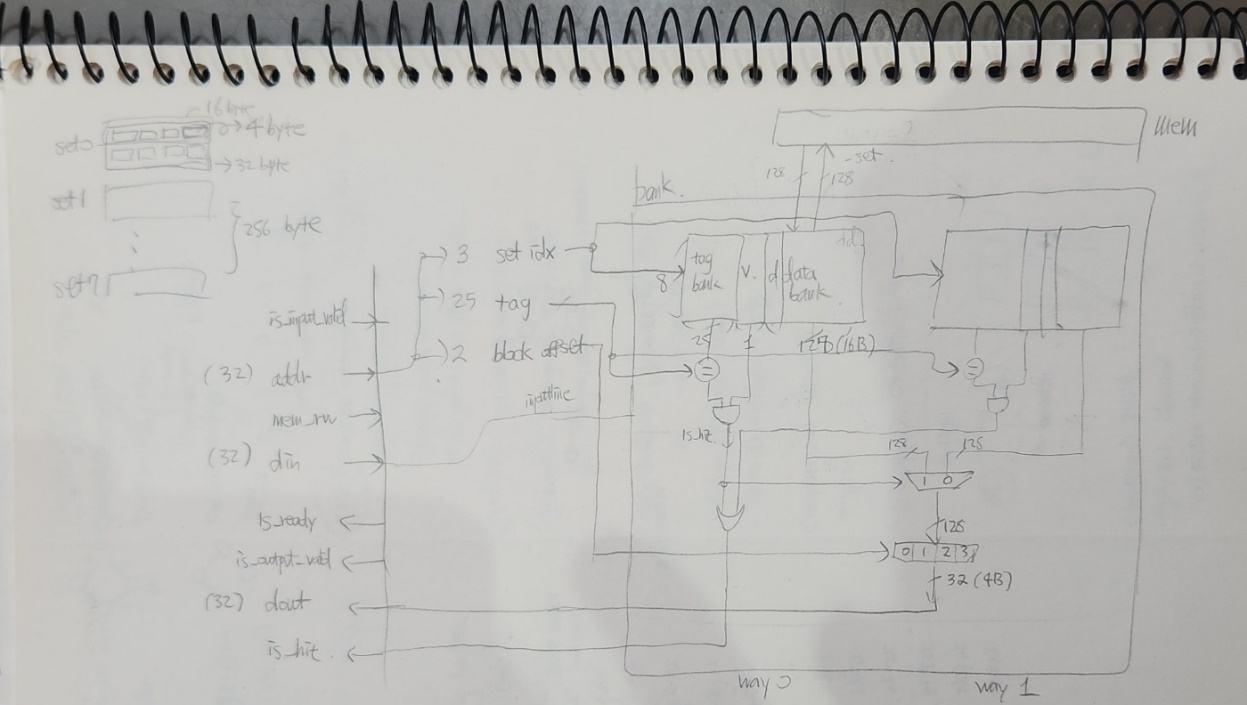
20220312 박준혁, 20220871 홍지우

**명예서약 (Honor Code)**

나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.

I completed this programming task without the improper help of others.

1. **Introduction**
   1. Direct-mapped Cache를 구현하였다.
2. **Design**
   1. 설계 내용  
      Associative로 구현하고 싶었으나 시간상 Direct-mapped로 구현하게 되었다  
        
      아래는 2-way로 설계한 그림이다. select하는 로직을 통째로 제외하면 본 과제에서 구현한 것과 동일한 구조를 가진다.



아래는 참고한 사진이다.  
ㅇ텍스트, 도표, 평면도, 기술 도면이(가) 표시된 사진

자동 생성된 설명

1. **Implementation**
   1. 캐시 구조
      1. 입출력

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명  
위와 같이 입력과 출력이 구성되어 있다.  
MEM stage에서 read, write를 구별하여 캐시에 접근하는 방식이다.  
cache\_is\_input\_valid의 경우는 다음과 같다.  
assign cache\_is\_input\_valid = EX\_MEM\_mem\_read | EX\_MEM\_mem\_write;

* + 1. 캐시 뱅크  
       텍스트, 스크린샷, 폰트이(가) 표시된 사진

       자동 생성된 설명 텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

       자동 생성된 설명  
       하나의 bank는 위와 같다. bank 안에는 tag, valid, dirty, data를 보관하고 있다  
       크기는 다음과 같이 정의되어 있다

텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

* + 1. Dmem과의 연결

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명  
dmem과는 다음과 같이 와이어링 되어 있다.   
캐시에서 요청하는 부분을 is\_input\_valid를 통해 조절한다.

* 1. Hit, Miss Policy
     1. Hit

텍스트, 폰트, 스크린샷, 그래픽이(가) 표시된 사진

자동 생성된 설명  
Hit 유무는 바로 async하게 알 수 있다.

* + 1. Miss  
       Miss관련해서는 state machine을 사용하였다.   
       4개의 state를 사용, dirty라인을 교체하여 메모리에 쓰고 (1)  
       메모리의 값을 read해서 캐시에 불러오고 (2)   
       위 두가지 동작을 를 구분, 연속적으로 작동시키기 위해 그렇다.  
       (CACHE\_IDLE, CACHE\_WRITE\_BACK\_REQUEST,   
       CACHE\_WRITE\_ALLOCATE\_REQUEST, CACHE\_DATA\_RECIEVED)
       1. Write-Back

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명  
Miss 발생시, Dirty가 있다면 교체해주는 로직이다.   
Dmem의 원래의 주소(output\_addr)에 바뀐 set을 통째로 전달한다

* + - 1. Allocate

텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명  
우선, write\_back이 있었던 경우를 대비, 쓰기 완료 여부를 입력으로 받아서  
write\_back -> write\_allocate로 넘어가는 로직을 추가해준다.  
(IDLE로 만들고 dirty를 0으로 바꾸면 다음 사이클에서 알아서 넘어감)  
allocate가 끝나면 (data\_ready=1) 뱅크의 데이터들을 업데이트해준다.

* + - 1. Write  
         Read의 경우는 위 단계에서 종료하면 되고, write면 지정된 위치에 쓰고 dirty를 1로 바꾸는 작업까지 해준다.

텍스트, 스크린샷, 소프트웨어, 멀티미디어 소프트웨어이(가) 표시된 사진

자동 생성된 설명

* 1. Hit Ratio

텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명 텍스트, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

Naive Opt

optimal에서 더 많은 hit이 나왔다.

1. **Discussion**텍스트, 폰트, 스크린샷, 대수학이(가) 표시된 사진

   자동 생성된 설명
   1. 본 과제에서는 direct mapped로 구현했다
   2. replacement policy 또한 associative cache에 해당하는 항목이다.  
      원래는 LRU로 하려고 했다.
   3. optimal에서 더 많은 hit이 나왔다.
   4. optimal의 경우 캐시의 사이즈 만큼을 단위로 iterate해, 그냥 순서대로 하는 것보다 캐시-친화적이며 conflict miss를 줄일 수 있다.
   5. way의 수를 늘리면 Conflict Miss가 줄어 hit ratio가 증가한다.