컴퓨터 구조 도전 과제

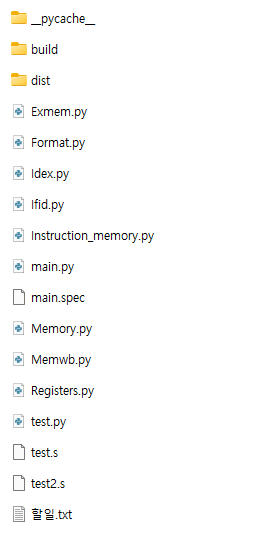
MIPS 파이프라인 시뮬레이터 만들기

이름 : 최태호

학과 : 소프트웨어융합학과

학번 : 20191685

코드 설명



다음과 소스코드파일을 들어가면 다음과 같은 파일들이 있음

main.py – 전체 ifid, idex, exmem, memwb, memory, register를 연결시켜주는 역할을 함.

해저드도 구현이 되어있음

registers – 레지스터들값 데이터들을 관리함

memory.py – 메모리들을 저장함.

Instruction\_memory.py – 코드들을 분석하여 바이너리 코드로 저장함.

ifid.py – instruction 메모리에서 코드를 읽어오고

pc값을 더하는 역할을 함

idex.py – ifid에서 instruction 코드를 얻어서 분석하고 Register에서 값을 불러오고

beq 연산도 실시를 한다.

exmem.py - idex에서 불러온 레지스터 데이터를 1번과 2번의 instruction 코드에 맞게 사칙연산을 실시함

memwb.py – 메모리파일에서 읽기 쓰기를 하는 역할을 함.

instruction\_memory.py – 처음에 파일을 불러오면 코드를 해석하고 바이너리 코드로 변환한후

제작하는데 Python을 사용하였고 Pyqt5를 사용하여 시각화 하였음

텍스트, 화면, 검은색, 스크린샷이(가) 표시된 사진

자동 생성된 설명

함수들만 가볍게 보면

\_\_init\_\_ :기본 값들을 초기화 함

InitUI : ui들을 초기화한다.

Reset : 새롭게 파일을 불러올 때 리셋을 이용하여 수치들을 초기화함

RisingEdge : RisingEdge 버튼을 눌렀을 때 실행이 되며 각 클래스마다 위에적은 해야할 일들을 수행함

LabelUpdate : RisingEdge같은 수치를 바꾸면 그 값을 시각적으로 업데이트 함.

다른 클래스들 마다 틀이 같음.

**사용법 설명**

EXE 파일 실행화면  
테이블이(가) 표시된 사진

자동 생성된 설명

**각 UI 설명**

테이블이(가) 표시된 사진

자동 생성된 설명

1. 현재 클럭을 표시함
2. 레지스터당 데이터를 표기한다 16진수로 표현
3. SPIM 파일을 컴퓨터에 연다.
4. Rising egde를 실시하여서 계산한다.
5. Code를 표시하고 각 코드마다 pc값, 16진수 표현, 코드원형을 표기함
6. 메모리에 데이터를 표기한다. 초기엔 0으로 초기화
7. 각 파이프라인의 IF/ID, ID/EX, EX/MEM MEM/WB에 저장된 값들을 표기한다.

사용법

1. 열기 버튼을 이용하여서 파일을 연다.
2. 2번 초기 레지스터값들을 수정함.
3. Rising edge 버튼을 이용하여서 한 클럭 단위로 계산을 보고 확인한다.