

<ALU>

기능: SrcA와 SrcB로 들어온 값을 ALUControl에 들어온 값에 따라 연산을 수행하여 ALUResult로 내보내는 역할을 한다. 이 때, branch나 cmp 역할을 위해 subtract 시의 결과값 노멀 밸류가 0이라면, 제로 플래그 역할에 해당하는 ALUFlags값을 set해준다.

<Extend>

기능: ImmValue에 들어온 24비트 값을 ImmSrc에 따라 특정 비트 만을 골라 32비트로 extend 변환하는 역할을 수행한다. 변환된 32비트의 값을 ExtImm이라고 할때, mov instruction일 시에 immvalue의 하위 8비트, load나 store일 때는 immvalue의 하위 12비트를 채운다. 예외적으로, branch일 때는 immvalue의 하위 24비트를 취해 ExtImm의 [25:2]에 채워넣고 ExtImm의 [1:0]은 각각 0을 넣는다 (shifting left two bits). 앞서 비트들을 immvalue로부터 취한 이후, imm value의 부호에 따라 extension이 다르게 일어나는데, 들어온 값의 MSB와 같은 값으로 extended된 비트를 채워준다.

<ControlUnit>

기능: instruction의 32비트 중 상위 16개 비트의 값을 각각 4비트의 Cond, 2비트의 Op, 6비트의 Funct, 4비트의 Rd로 받는다. 이 16개 비트를 통해, AL인지 EQ인지, arithmetic add인지 str인지 branch인지 등등의 각 케이스에 따라 control signal들을 설정하여 내보낸다.

<RegisterFile>

기능:

1. 클락이 Rising edge일 때: RA1과 RA2에 들어온 레지스터 주소로 접근하여 레지스터에서 값을 가져와 RD1와 RD2로 내보낸다.

2. 클락이 Falling edge일 때: WD3로 들어온 값을 RA3 주소에 해당하는 레지스터에 저장한다.

<ARMPProcessor>

기능: PC값, PCPlus4, PCPlus8에 대한 값을 구현해주고 위의 4개의 서브모듈들을 각각 올바른 매

개변수 값들을 연결해주며 모두 호출해준다.