CO PROJECT

ARM Simulator

Detailed Description Of 5 Stages

Fetch

We have made an array in which the instruction is stored at the index which is the address for the particular instruction.

At the time of fetch the current instruction is updated. Finally, the Program Counter is increased by 4.

Decode

All the components of an instruction are extracted using bit shifting. Then according to that, the flag, immediate values, opcode, offset, destination and the operand values are updated.

We know that an instruction can have 3 types of Flag.

If Flag is 0: Then it may have immediate value 0 or 1.If it is 0 then we have Register numbers stored in Operand 1 and Operand 2 and Destination Register. And the corresponding instructions associated would be ADD,

XOR, SUB, ADD, ADC, CMP, ORR, MOV, MVN.

If the immediate value is 1, then operand 2 stores the immediate value rather than register no. in the last 12 bits of the instruction. Instructions can be ADDI,

XORI, SUBI, ADDI, ADCI, CMPI, ORRI, MOVI, MVNI.

If Flag is 1: These type of instructions deal with memory.

They can be LDR, STR.

If Flag is 2: These are the branch instructions. Instructions can be BLE, BEQ, BNE, BLT, BGT, BAL(Branch Always) etc.

Execute

This function calculates the desired result.

If it is of LDR or STR type, then the value is either retrieved or stored to a location in the memory.

The memory is simulated as a 2D matrix where each row corresponds to an array in the memory and offset is the column index that can be used to access the ith element of the particular array.

If it is a Branch instruction, we check the first bit of the offset and accordingly extend the sign bits to make it a 32-bit number. According to the type of branch instruction and the result of previous compare instruction(stored in flag Z and N), we jump to the desired location.

Memory

This stage prints about all the updates in memory.

WriteBack

This stage writes the result calculated in execute stage in the destination.

Instruction Format

Instructions in ARM Simulator are designed in the following manner:

FOR DATA PROCESSING INSTRUCTIONS:

Condition	Flag	Immediate	Opcode	S	Operand1	Destination	Operand2
4 bits	2 bits	1 bit (0 if no immediate operand, 1 otherwise)	4 bits	1 bit	4 bits	4 bits	12 bits (more bits to accommodate immediate values)

The above bit representation was used in the following instructions:

- ADD: Add Destination(Rd), Operand1(Rs), Operand2(Rn) / Add Destination(Rd), Operand1(Rs), Operand2(immediate value)
- SUB: Sub Destination(Rd), Operand1(Rs), Operand2(Rn) / Sub Destination(Rd), Operand1(Rs), Operand2(immediate value)
- ADC: Adc Destination(Rd), Operand1(Rs), Operand2(Rn) / Adc Destination(Rd), Operand1(Rs), Operand2(immediate value)
- CMP: Cmp Destination(Rd), Operand1(Rs), Operand2(Rn) / Cmp Destination(Rd), Operand1(Rs), Operand2(immediate value)
- AND: And Destination(Rd), Operand1(Rs), Operand2(Rn) / And Destination(Rd), Operand1(Rs), Operand2(immediate value)

- ORR: Orr Destination(Rd), Operand1(Rs), Operand2(Rn) / Orr Destination(Rd), Operand1(Rs), Operand2(immediate value)
- EOR: Eor Destination(Rd), Operand1(Rs), Operand2(Rn) / Eor Destination(Rd), Operand1(Rs), Operand2(immediate value)
- MOV: Mov Destination(Rd), Operand2(Rn) /Mov Destination(Rd), Operand2(immediate value)
- MVN: Mvn Destination(Rd), Operand2(Rn) /Mvn Destination(Rd), Operand2(immediate value)

FOR BRANCH INSTRUCTIONS:

Condition	Flag	Opcode	Offset
4 bits	2 bits	2 bits	24 bits (Immediate values)

The above representation was used for the following instructions:

- BEQ
- BNE
- BGE
- BGT
- BLE
- BLT
- BAL

The branch methods are called after a CMP instruction, which sets the Z flag to the respective value based on the comparison. After this, the Branch method need simply read this flag and decide to take the branch or not accordingly.

FOR DATA TRANSFER INSTRUCTIONS:

Condition	Flag	Opcode	Operand1(Rn)	Destination(Rd)	Offset(12)
4 bits	2 bits	6 bits	4 bits	4 bits	12 bits (immediate value)

The above representation was used for the following instructions:

- LDR: Ldr Destination(Rd), [Operand1(Rn), #immediate]
- STR: Str Destination(Rd), [Operand1(Rn), #immediate]

Team members: Aarushi Agarwal (2016216)

Arushi Chauhan (2016019)

Pragya Prakash (2016067)