COL216 Computer Architecture
Lab Assignment 4

This assignment aims at designing ARM datapath modules and implementing these on the BASYS3 board. The following subset of ARM instructions has been chosen for implementation.

Arithmetic:	<add|sub|rsb|adc|sbc|rsc> {*cond*} {s}
Logical:	<and | orr | eor | bic> {*cond*} {s}
Test:	<cmp | cmn | teq | tst> {*cond*}
Move:	<mov | mvn> {*cond*} {s}
Branch:	<b | bl> {*cond*}
Multiply:	<mul | mla> {*cond*} {s}
Load/store:	<ldr | str> {*cond*} {b | h | sb | sh }

cond: <EQ|NE|CS|CC|MI|PL|VS|VC|HI|LS|GE|LT|GT|LE|AL>

Instructions excluded are as follows.

Co-processor:	cdp, mcr, mrc, ldc, stc
Branch and exchange:	bx
Load/Store multiple:	ldm, stm
Software interrupt:	swi
Atomic swap:	swp
PSR transfer:	mrs, msr
Multiply long:	mull, mlal

Instruction swi will be added later.

The building blocks are as follows.

ALU

Inputs: two 32-bit operands, operation to be performed, carry

Outputs: 32-bit result of arithmetic/logical operation, next flag values

Functionality:

It is a combinational circuit that performs the arithmetic/logical operations for the DP instructions. The “operation to be performed” input can come from the opcode field of DP instructions.

Its ability to add/subtract is also used by other instructions by giving appropriate inputs. This includes address offset addition/subtraction for DT and b|bl instructions as well as

addition required in mla instruction. The ALU need not inherently know whether the addition it is doing is for add or ldr or bl or mla.

Apart from the 32-bit result, the next values for the 4 flags are also output. Whether the flag values actually need to change is decided elsewhere.

Shifter

Inputs: 32-bit data to be shifted, shift type (LSL|LSR|ASR|ROR) and shift amount

Outputs: 32-bit data after shifting, shifter carry

Functionality:

It is a combinational circuit that performs the required shift operation by the specified number of bits. This serves the purpose for DP and DT instructions requiring shift. It also outputs shifter carry. Which carry (ALU or shifter) is to be used for updating the carry flag is decided elsewhere.

Multiplier

Inputs: Two 32-bit operands

Outputs: One 32-bit result

Functionality:

It is a combinational circuit that performs multiplication operation for mul and mla instructions. The addition required for mla instruction is done by ALU. For this assignment, a detailed design for multiplier is not required. Simply use multiply symbol in VHDL and leave the rest to the synthesizer.

Register File

Inputs: 32-bit data to be written, two 4-bit read addresses, one 4-bit write address, clock, reset, write enable

Outputs: two 32-bit data outputs, PC output

Functionality:

It has an array of sixteen 32-bit registers accessible through two read ports and one write port. It maintains a copy of register 15 out side the array for easy access as program counter. Clock and write enable have their usual meaning. Reset is used to initialize PC.

Processor-Memory path

Inputs: 32-bit data from processor, 32-bit data from memory, type of DT instruction, byte offset in memory address

Outputs: 32-bit data to processor, 32-bit data to memory, memory write enable signals

Functionality:

It is a combinational circuit that provides byte manipulation and sign/zero extension needed for ldrb, ldrh, ldrsb, ldrsh, strb and strh instructions. For byte addressable memory, byte level write enable signals are required. These are also generated by this module.