Computer Architecture Practice
20 May 2020
Instruction Set Architecture for 16-bit Processor

1) **Instruction Format and Size: F D E**
   This design considers the Instruction size as 16-bits.
   The processor has 16 registers ranging from 0000 to 1111 (ie., R0, R1, ...R15.)
   CF = Carry Flag
   Rdst = Destination register
   Rsrc1 = Source Register 1
   Rsrc2 = Source Register 2
   { } = Concatination
   [] = Address Value of Oprand

2) **Register based Instructions:**

Opcode	Rdst	Rsrc2	Rsrc1
		4 bits	4 bits

Opcode : 0000 to 1111

3) **Load Instruction:**

Opcode	Rdst	Address
4 bits	4 bits	8 bits

Opcode : 1111

4) **Store Instruction:**

Opcode	Rdst	Address
4 bits	4 bits	8 bits

Opcode : 1110

5) Opcode (Operation Code) Encoding:

***Assumed each stage of addition and subtraction as one pipelining stage due to equal delay for the logical operation.

OPCODE	Description	Usage	Explanation	Machine Cycle
0000	Addition	ADD Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 + Rsrc1	F D E(1-6) = 8
0001	Addition with Carry	ADC Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 + Rsrc1 + CF	F D E(1-6) = 8
0010	Subtraction	SUB Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 - Rsrc1	F D E(1-6) = 8
0011	Subtraction With Barrow	SBB Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 - Rsrc1 -1	F D E(1-6) = 8
0100	Multiplication	MUL Rdst, Rsrc2, Rsrc1	{R15, Rdst} = Rsrc2 * Rsrc1	F D E(1-11) = 13
0101	Floating Point Addition	FADD Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 + Rsrc1	F D E(1-21) = 23
0110	Floating Point Subtraction	FSUB Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 - Rsrc1	F D E(1-21) = 23
0111	Floating Point Multiplication	FMUL Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 * Rsrc1	F D E(1-24) = 26
1000	Halt	HLT	Halt	F D E(1-4) = 6
1001	Register Complement	CMP Rdst, Rsrc1	Rdst = ~Rsrc1	F D E = 3
1010	Logical Bit Wise XOR	XOR Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 ^ Rsrc1	F D E = 3
1011	Logical Bit Wise NAND	NAND Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 ~& Rsrc1	F D E = 3
1100	Shift Right	SHR Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 >> Rsrc1 If Rsrc1 = 2 then shift the Rsrc2 in 2bit towards right	F D E(1-4) = 6
1101	Shift Left	LHR Rdst, Rsrc2, Rsrc1	Rdst = Rsrc2 << Rsrc1 If Rsrc1 = 2 then shift the Rsrc2 in 2bit towards Left	F D E(1-4) = 6
1110	Store	STR Rdst, 8-bit Address	[8-bit Address] = Rdst	F D E(1-2) = 4
1111	Load	LDR Rdst, 8-bit Address	Rdst = [8-bit Address]	F D E(1-2) = 4