CS4403 Wednesday 8th January 2014

Chapter 2 - Instruction Set Architecture

Common word length = 32 bits
4 x 8 bit bytes (eg ASCII chars)

Byte - addressable Memory : assigns an address to each byte.
Word length = 0, 4, 8 addresses (instead of 0, 1, 2)

Big - Endian
Little - Endian (LSB)

2 Bytes per word → 0, 2, 4
8 " " " → 0, 4, 8

Instructions & Sequencing

Register Transfer Notation (RTN)
R4 ← [R2] + [R3]

Add R2 + R3 and put result in Register 4

△ Left - hand location } Always
Right - hand value . }

Load R2, LOC
Add R4, R2, R3
operation
destination
source

LD = Load

ST = Store

ADD

RISC & CISC Instruction Sets

One word instructions	Multi-word instructions
eg LD & ST	
LOAD/STORE Architecture	
Load proc_register, mem_location	
RTN = C ← [A] + [B]	

IR = Instruction Register

Two phase procedure - fetch & execute

Branching

Program adding a list of numbers - repetition!

Branch_if_[R2] > 0 LOOP

-Go to LOOP if [R2] > 0

Branch_if_[R4] > [R5] LOOP

OR

BGT R4, 25, Loop

Cannot store full address as it's greater than 32 bits so increment inside the loop the address of the next address. (Ri)

EA = Effective Address

Add R4, R6, #200

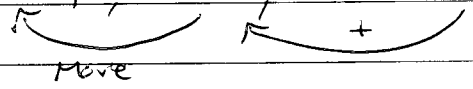
is used to specify that it is an immediate value not an address.

∴ So Add R6 + 200 and put result in R4

! not Add R6 + Mem-loc 200 " " " " "

A MOVE R4, #NUM 1

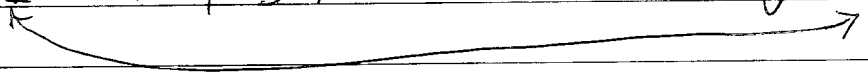
B ADD R4, R0, ## NUM 1



R0 is usually always 0, so to achieve A we do B.

pseudo instruction

ADDI R2, R3, 5 (instead of #5)



I takes place as it represents immediate addressing

Ensure Names, commas, values etc are in a straight line down.

CS4403 Wednesday 15th January 2014

Condition Codes (CISC Machines)

N
Z
V
C - carry

CISC branches check condition code flags.
- Branch - if - overflow
- Branch - if - carry

Move R4, #NUM1
↑
use the immediate value
if it was MOVE R4, NUM1, it would move the value from NUM1 into R4.

Add R3, (R4)+ ← use register 4 as an indirect address, + = post increment. (+ = increment of the pointer)

SUBTRACT R2, #1 ← subtract 1 from Register 2

How are instructions used?
Exam - not required to know program off by heart just how it works.

MoveByte R8, (R6)+

The above is the first time the program accesses memory, thus doesn't happen in RISC, so this program is CISC.

Machine Code = Assembly Code

CS4403 Monday 3rd February 2014

Chapter 5 - Basic Processing Unit.

A digital Processing System.

- Chop up the job into smaller jobs.
- Pipelining

5 Stages Implementation of a RISC Processor

Instruction 1 - Memory (LOAD + STORE)

Load R5, X(R7)

- Computational Instruction

Add R3, R4, R5

Summary => A RISC approach (RISC)

3. Perform an ALU operation, on registers and we'll go through a whole cycle to get at memory.

★ STAGE 1 => FETCH, this is part of a different path, so it does not appear in diagram.

The datapath - Stages 2 to 5

CS4403 Tuesday 4th Feburay 2014

Cos. signal gen without box / interface
Wave forms using Arduino

Tools -> Board ->
Tools -> Serial Port -> COM1 / COM7

Attempt 2:

Sketch → add file CS4403 Lab 2

// ~~constream~~ =

// sample = constream (t-sample, 0, one Hz Sample);

removed the isave line from the script and also created the .h file in notepad and added it.