Sep - 27

Note Title 27-09-2012

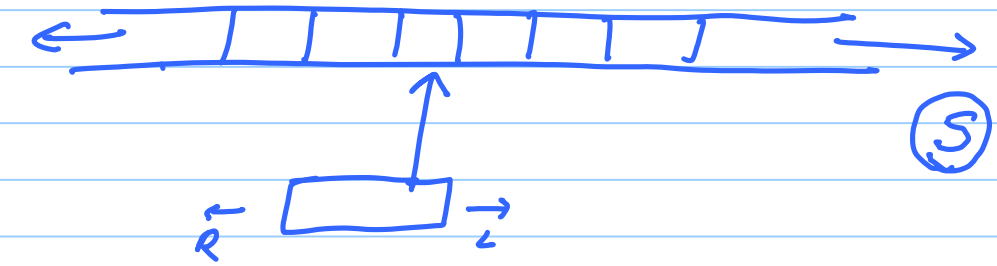
SUBN (Subtract and Branch if negative)

$A + B = A - (0 - B)$

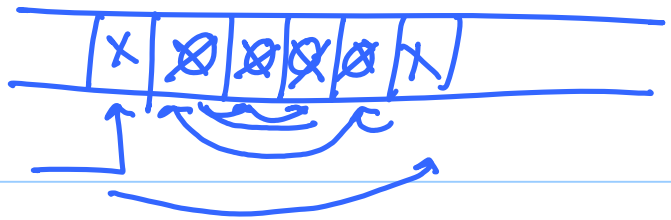
Sub is inherently more powerful than an add.

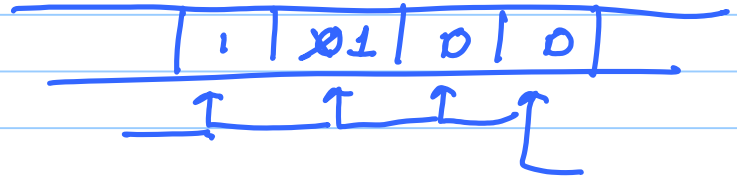
OISC – One instruction set computer

Turing Machine



Church-Turing Thesis





(1)

Properties of an instruction set.

- Complete {Necessary}

Desirable: 1) finite and preferably few instructions

[CONCISE]

ARM. MIPS. x86 (INTEL)

2) GENERIC — All common ops. should be on inst. assigned to them

3) SIMPLE —

4) REGULAR

	COMPLETE	GENERIC	SIMPLE	REGULAR
ARM	✓	✓	RISC ✓	✓
MIPS	✓	✓	RISC ✓	✓
x86	✓	✓	CISC X	X

MIPS Instruction Set

32 registers: 0 31

0 → special register
(always set to zero)

No register for holding the pc.

sp → stack pointer

ra → (lr in ARM)

No, conditional instructions

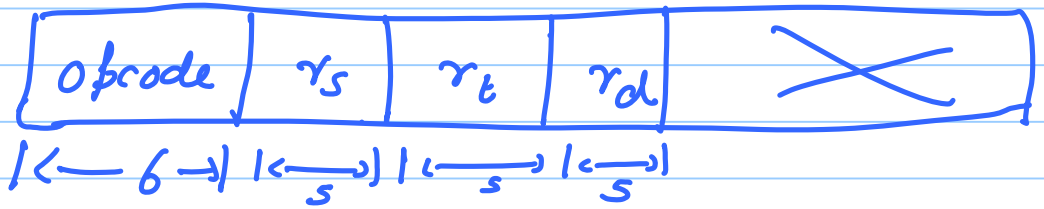
Arith- logical.

add r_1, r_2, r_3 $(r_1 = r_2 + r_3)$

sub/ r_1, r_2, r_3 $(r_1 = r_2 - r_3)$

and/or/nor sll → shiftleft srl → shift right.

representation : R- format



r_d - destination register

r_s → src1

r_t → src2

I-format Instructions

addi / subi $r_1, r_2, 30$

Instructions that take an immediate as the second operand have a different opcode.

* The immediate in MIPS is 16 bits long

load a 32 bit constant

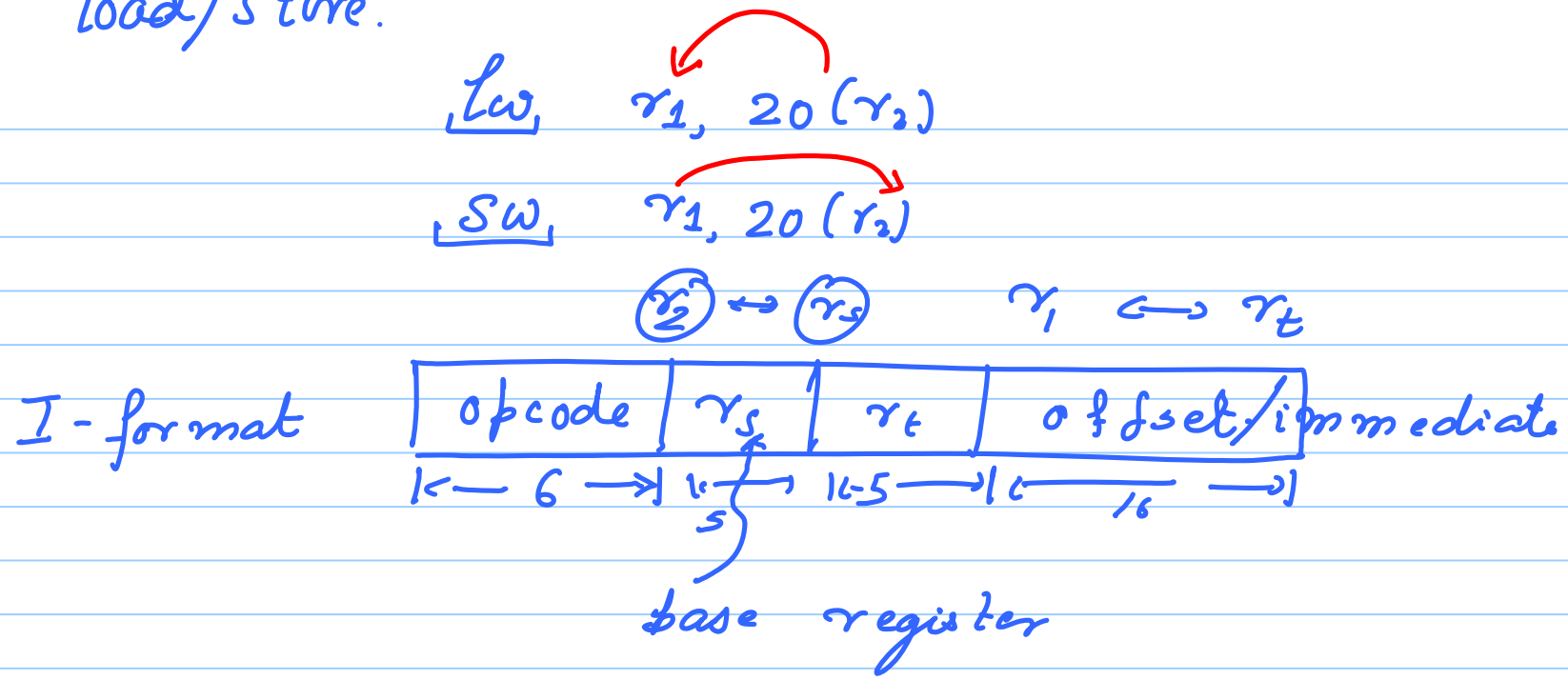
r_0 = 0

addi $r_1, r_0,$ <16 bits>

lui $r_1,$ <16 bits>

[load upper immediate]

load/store.


lw r_1, 20(r_2)
sw r_1, 20(r_2)
r_2 ↔ r_s
r_1 ↔ r_t
I-format
opcode
r_s
r_t
offset/immediate
6
5
5
16
base register

Branches in MIPS : (NO CPSR)

(I- format)

beq r_1, r_2, 200

bne r_3, r_4, 300

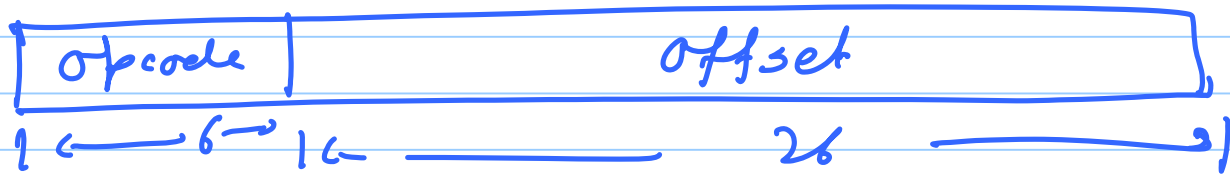
beq : if (value(r_1) == value(r_2))

pc += 4 + offset x 4

Jump Instructions.

	j	jr (I)	jal
	(jump)	(jump to value stored in register)	(Same as bl in ARM)
ARM Equivalent	B	MOV PC, R_x	BL

J - Format (j, jal)



Summary:

1) Go to Wikipedia: Study the One Instruction Set Computer.

2) How do you know that an ISA can represent all programs?

Prove that it is equivalent to a Turing Machine.

3) Turing Machine: Mathematical machine with a tape and tape-head.

4) MIPS ISA: (Not very different from ARM)

Differences:

	MIPS	ARM
Registers	32	16
Special Regs.	sp, ra (lr), zero	sp, lr, pc, cpsr
Imm. Size	16 bits	12 bits.

Imm. Inst.	Separate	Same.
Formats:	R, I, J	Slightly Complex
Branches:	beq, bne (NO CPSR) | Two regs.	No reg. as argument