CISC Design

Hardware Flowchart

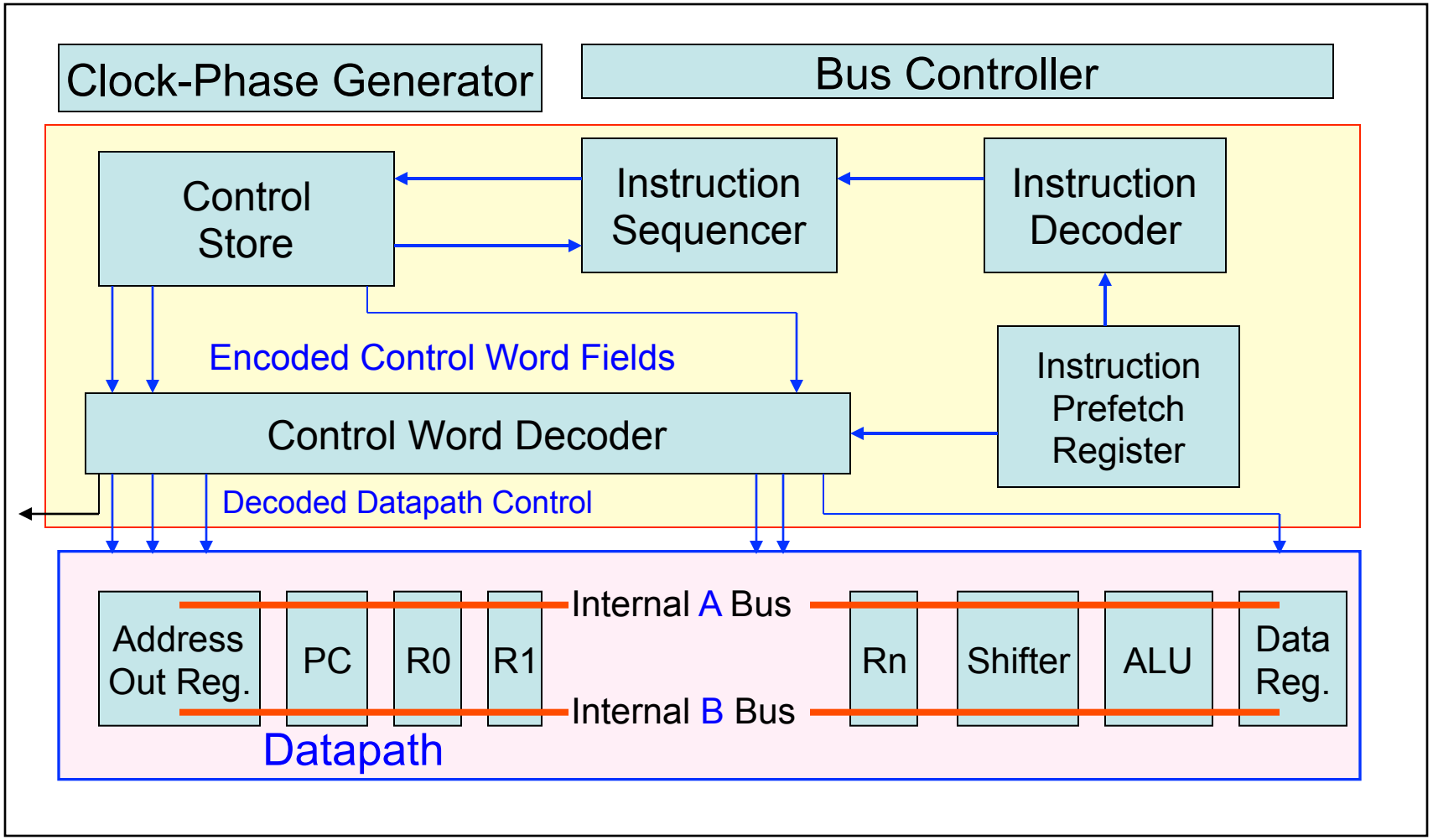
Virendra Singh
Computer Architecture and Dependable Systems Lab
Department of Electrical Engineering
Indian Institute of Technology Bombay
http://www.ee.iitb.ac.in/~viren/
E-mail: viren@ee.iitb.ac.in

EE-309: Microprocessors

Lecture 23 (15 Sep 2015)

CADSL

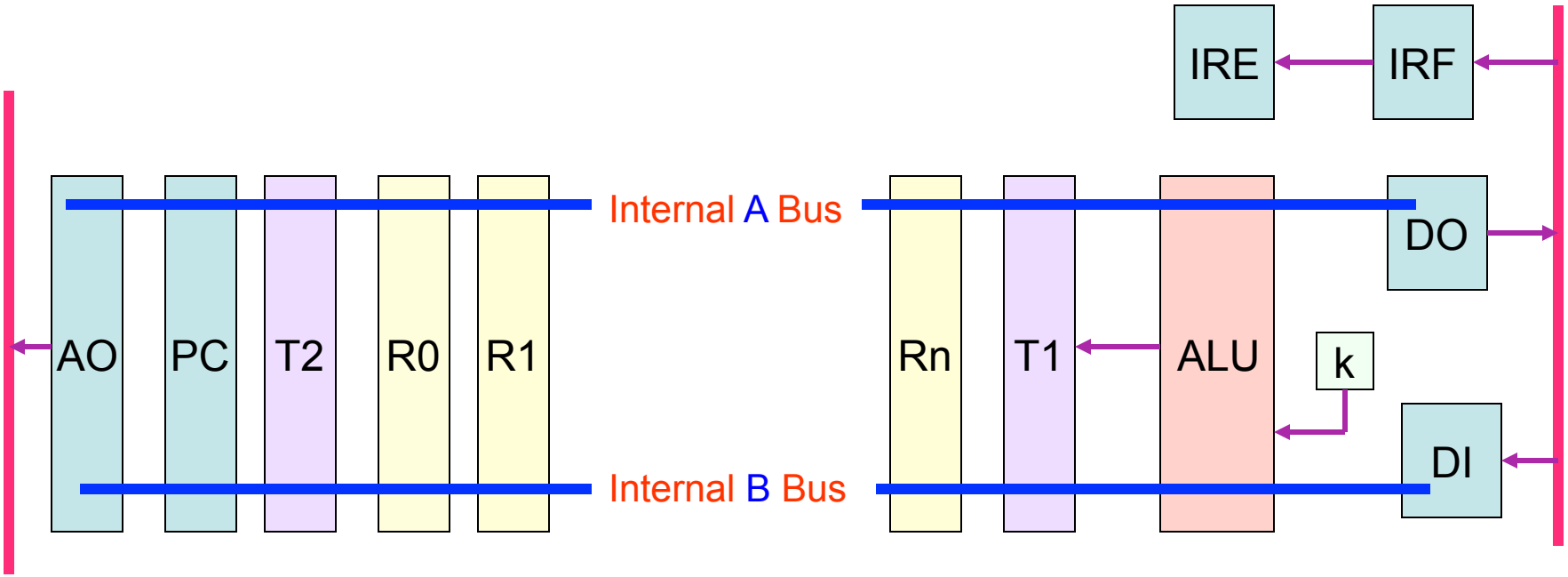
Micro-coded Implementation



MIN Instruction Set

- ✓ ADD
- ✓ AND
- ✓ BZ – Branch if zero bit is set. (Register Indirect only)
- ✓ LOAD – Second operand is source and Rx is destination
- ✓ POP – Postincrement with register indirect only
- ✓ PUSH – Predecrement with register indirect only
- ✓ STORE
- ✓ SUB
- ✓ TEST

MIN Datapath



Flowcharts

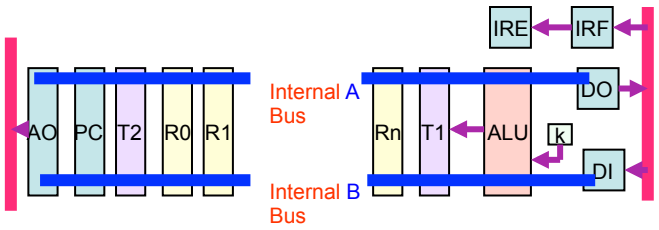
ADD RX AR RY

Register-to-Register

R → R ADD

rx → a → alu
ry → b → alu

t1 → b → ry



ADD RX AI (RY)

Register-to-Memory

R → M ADD

edb → di
ry → b → ao

State

di → b → alu
rx → a → alu

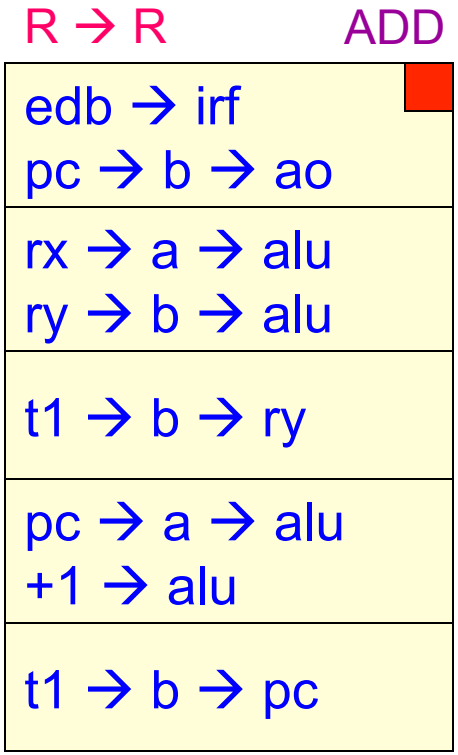
ry → b → ao
t1 → a → do

Sequence

Flowcharts

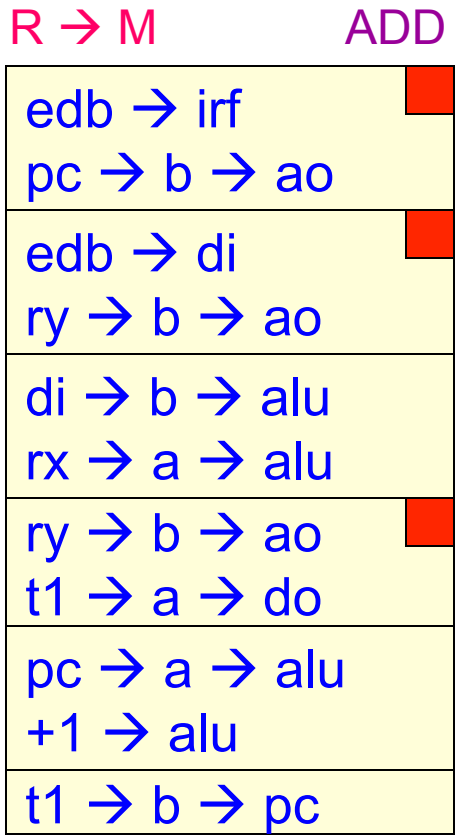
ADD RX AR RY

Register-to-Register



ADD RX AB (RY)

Register-to-Memory



- Execution Speed

Level 1 Flowchart - ADD

ADD RX AR RY

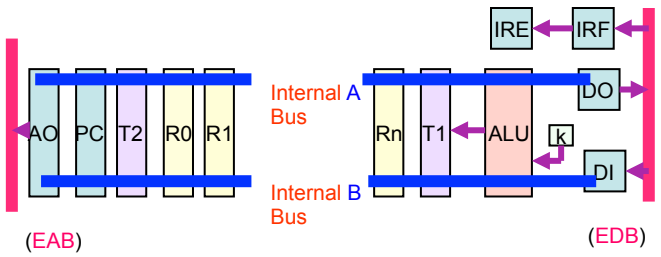
Register-to-Register

R → R ADD

rx → a → alu ry → b → alu	edb → irf pc → b → ao
	pc → a → alu +1 → alu
t1 → b → ry	irf → ire t1 → b → pc

Operation tasks

Housekeeping tasks



Level 1 Flowchart - ADD

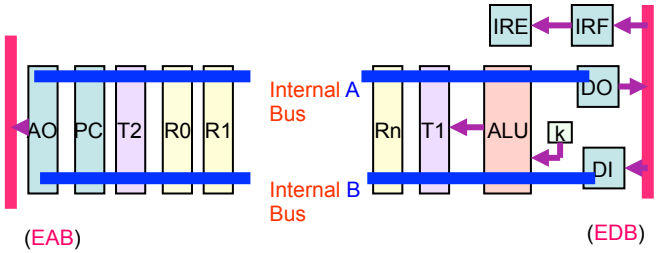
ADD RX AI (RY)
Register-to-Memory

R → M ADD

edb → di ry → b → ao	edb → irf pc → b → ao
di → b → alu rx → a → alu	pc → a → alu +1 → alu
ry → b → ao t1 → a → do	irf → ire t1 → b → pc

Operation tasks

Housekeeping tasks



Merged Level 1 Flowchart - ADD

ADD RX AR RY

Register-to-Register

R → R ADD

rx → a → alu
ry → b → alu

edb → irf
pc → a → alu, ao
t1 → b → ry
+1 → alu

irf → ire
t1 → b → pc

ADD RX AI (RY)

Register-to-Memory

R → M ADD

edb → irf
pc → a → alu, ao
+1 → alu

edb → di
ry → b → ao, t2
t1 → a → pc

di → b → alu
rx → a → alu

irf → ire
t1 → a → do
t2 → b → ao

Merger

- Speed
- Identical states

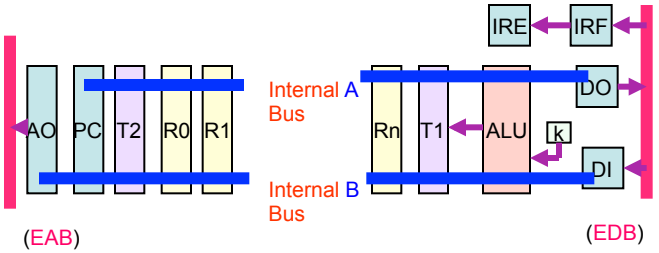
Merged Level 1 Flowchart - ADD

ADD RX AR RY
Register-to-Register

when AO is connected to
B (internal) bus only

Do level2 flowchart of the fastest instruction

Point out inadequacy in Datapath



R → R ADD

rx → a → alu ry → b → alu
edb → irf pc → b → ao t1 → a → ry
pc → a → alu +1 → alu
irf → ire t1 → b → pc

Level1 Flowchart

Level1 Flowchart

- At the beginning of instruction execution, IRE is assumed to contain the current instruction
- Instruction execution begins with the address mode sequence
- The execution sequences for Register-to-Register instructions cannot be shared
- The execution sequences for standard dual operand instructions are identical

Thank You