Slides for lecture 5

CE/CZ 3001:
Advanced Computer Architecture

(Module 3: Data-path and Control Design)

Dr Smitha K. G.
School of Computer Science
And Engineering

Topics covered till now

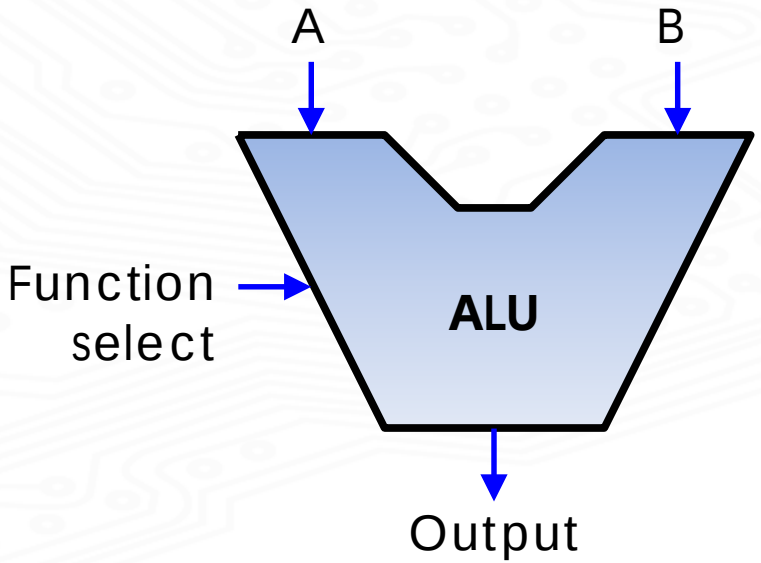
- LEGv8 ISA:
  - Instruction format
    - R-format, D-format, I- format, CB format and B-format
  - Addressing modes
    - Register addressing
    - Base addressing
    - Immediate addressing
    - PC-relative addressing
  - Functionality
    - ALU instructions
    - Data transfer instructions
    - Conditional and unconditional instructions

Outline

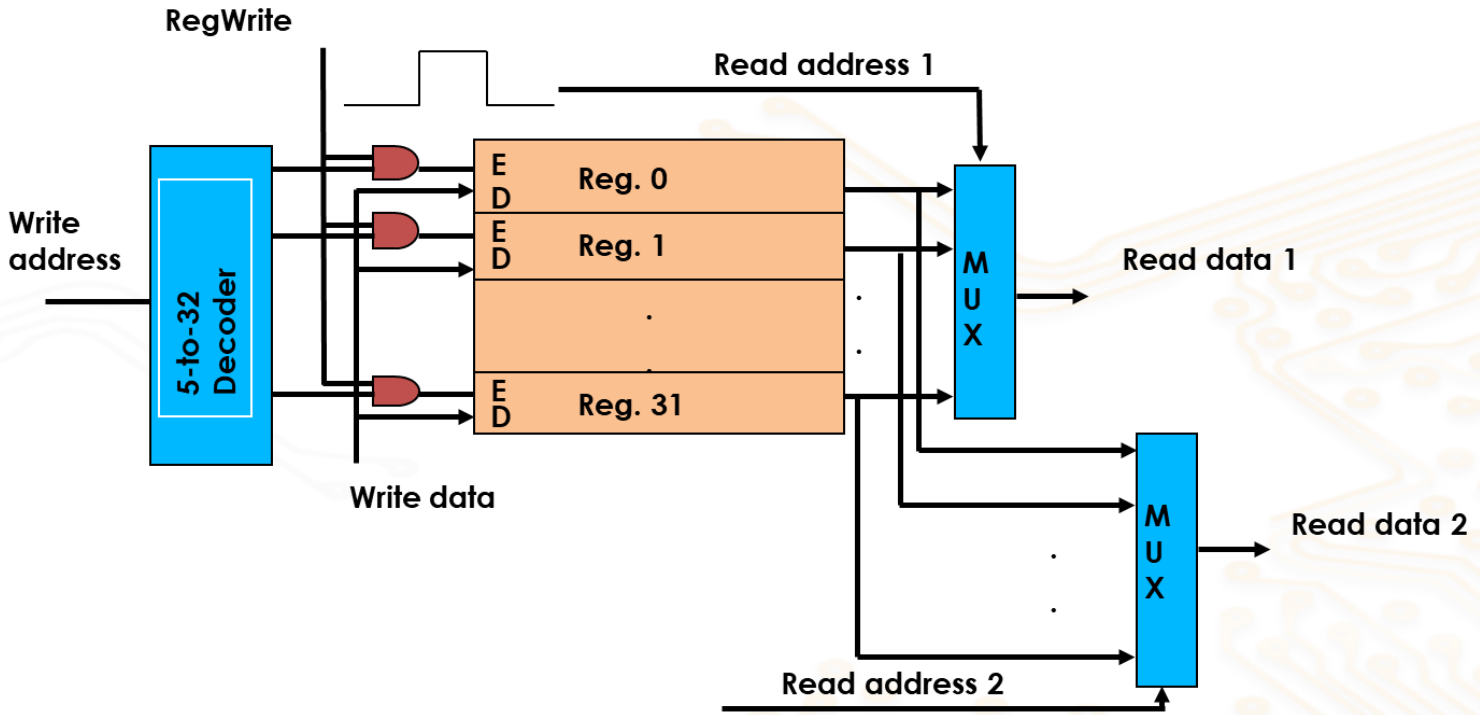
- Review of basic logic devices and register file
- Parts of datapath design
- Single-cycle datapath design
  - R-type
  - D-type
  - Combined R and D type

Basic datapath components

Arithmetic and logic unit (ALU)

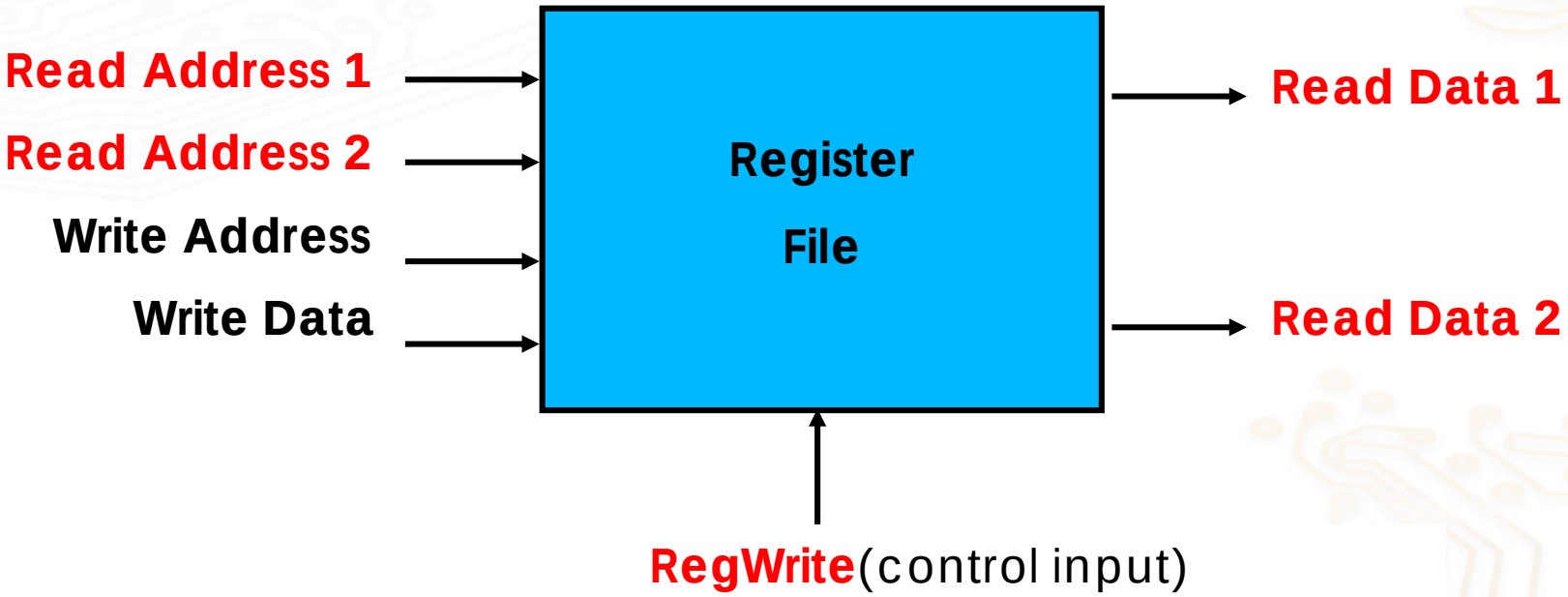


Register File



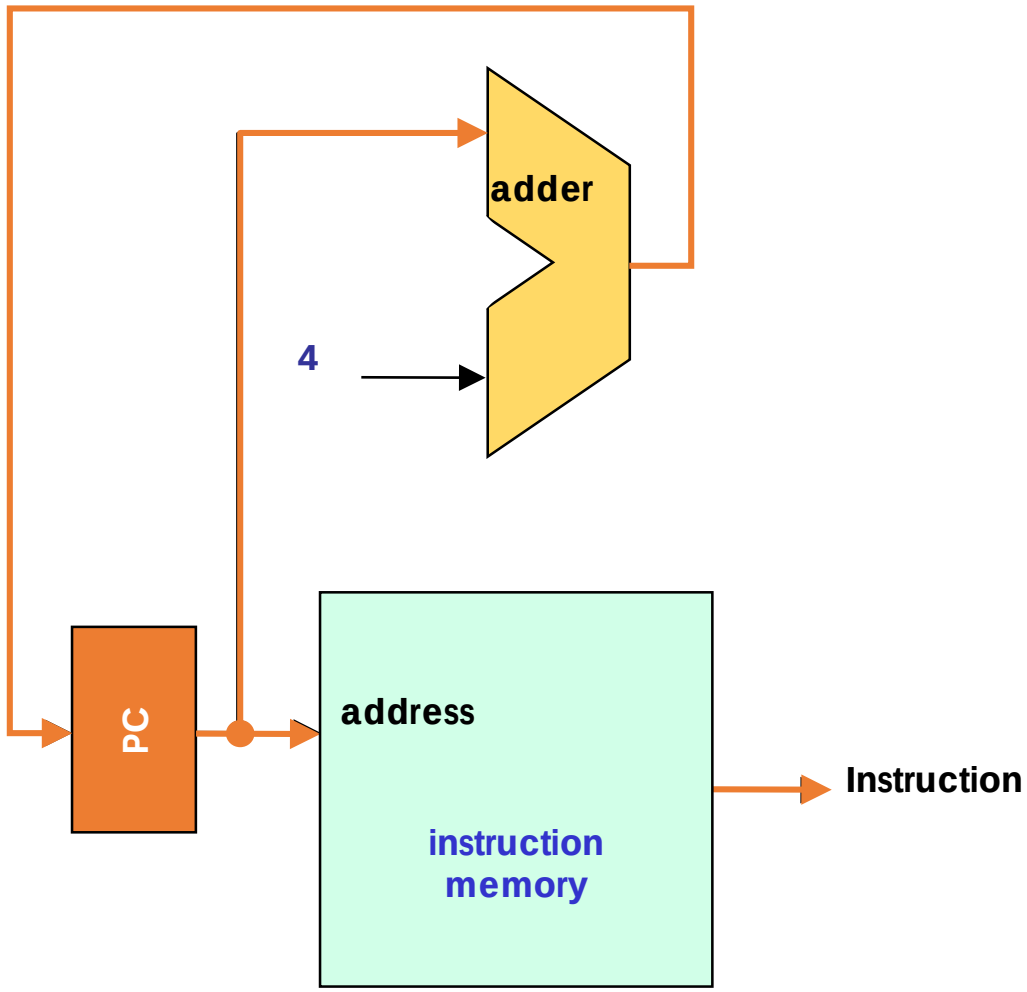
Register File

- Consists of a set of registers that can be read and written by supplying a register number to be accessed
- Can be implemented with a multiplexer for each read and a decoder to write and array of registers built from D-FF



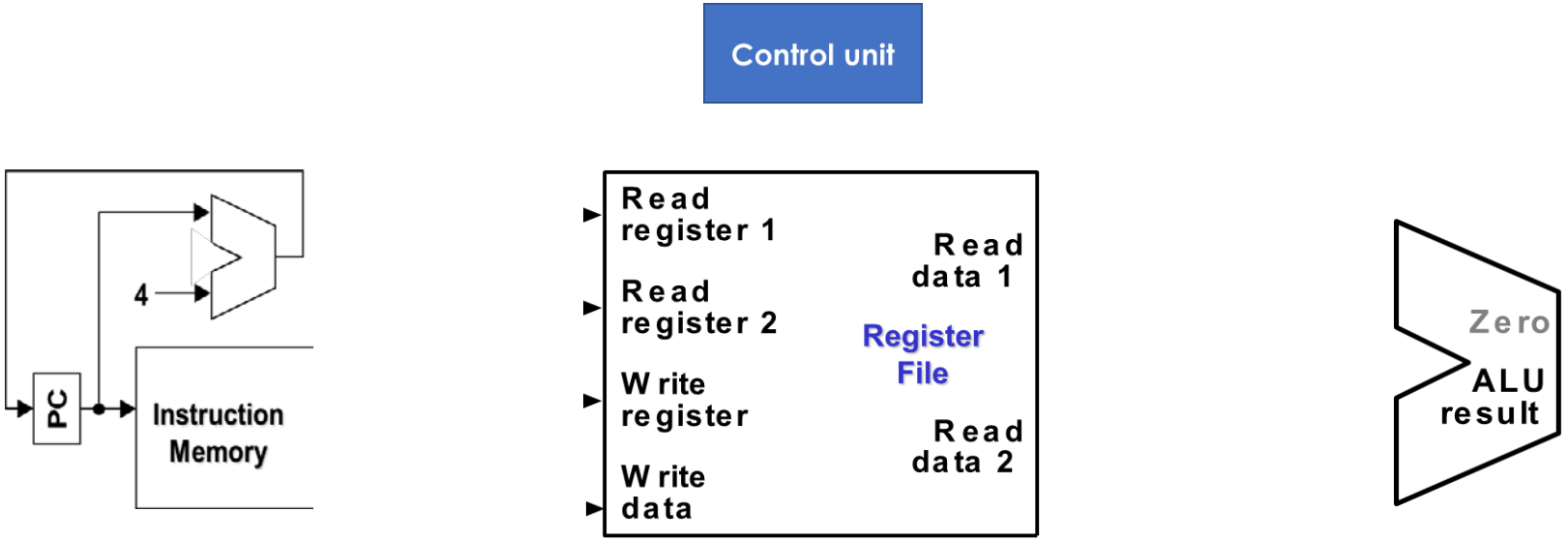
- For a regiter file with 64 registers each register having a width of 32, how much will be its read address size?

Introduction to datapath – Instruction fetch

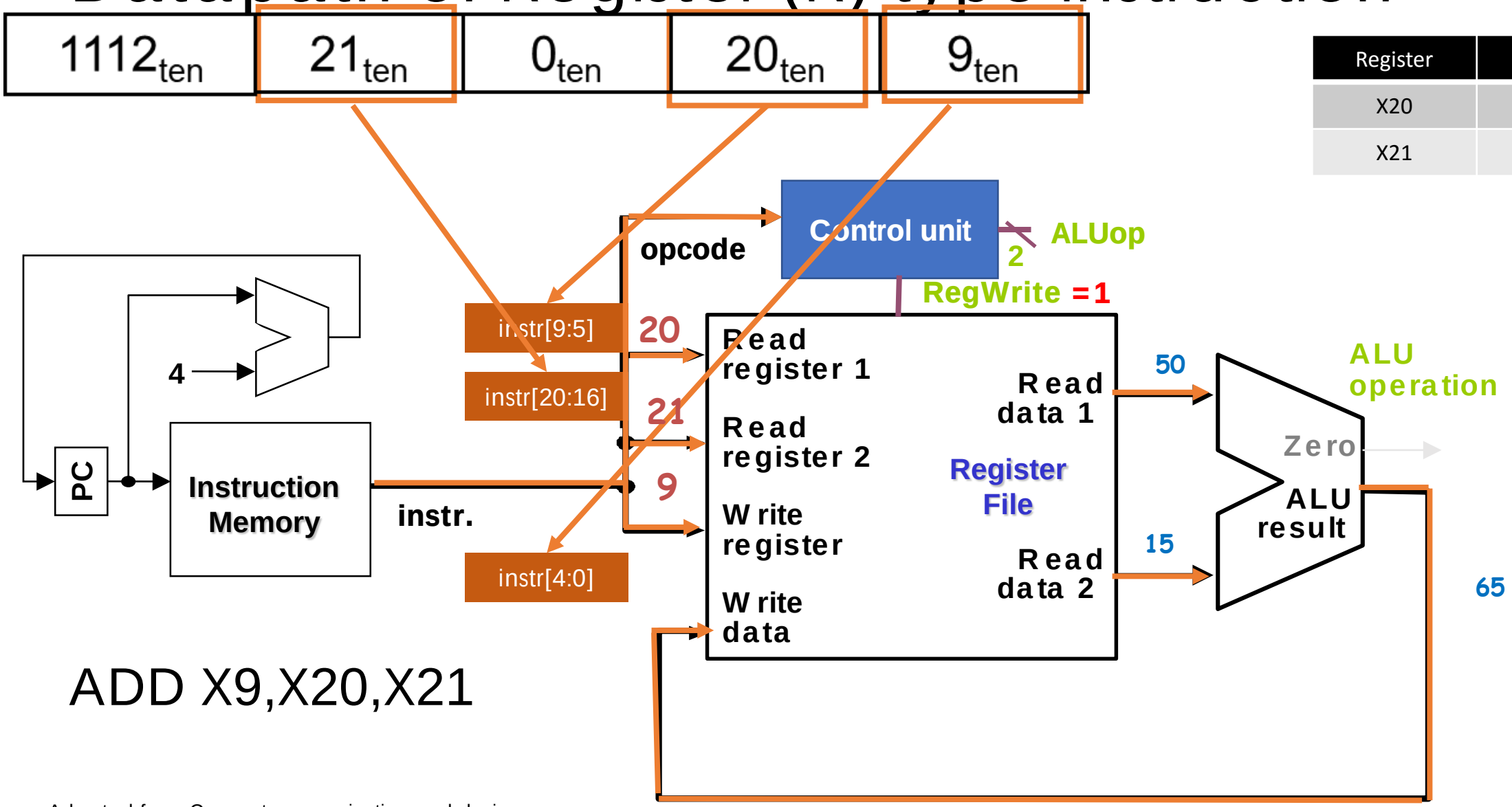


Datapath for R type instructions: (register addressing)

opcode	Rm	shamt	Rn	Rd
11 bits	5 bits	6 bits	5 bits	5 bits



Datapath of Register (R) type instruction



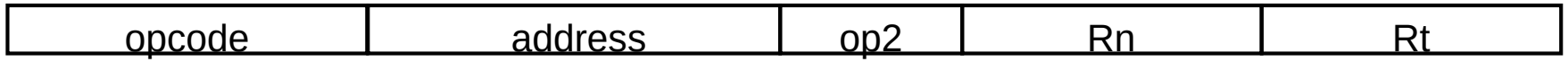
Register	Value
X20	50
X21	15

ADD X9,X20,X21

Adapted from Computer organization and design:
the hardware/software interface ARM edition , by D. A. Patterson, J. L. Hennessy & P. Alexander, 2017, Morgan Kaufmann.

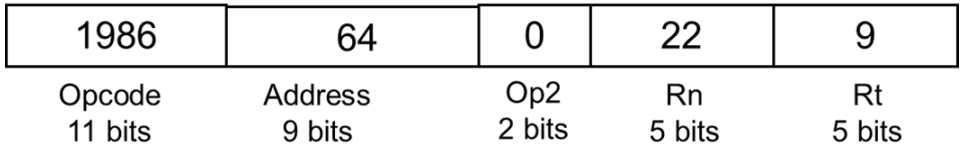
- Can we have a similar instruction in LEGv8 resembling the “MOV” instruction?

Data transfer (D)type –Memory related instructions



LDUR Rt, [Rn, #address]

11 bits | 9 bits | 2 bits | 5 bits | 5 bits

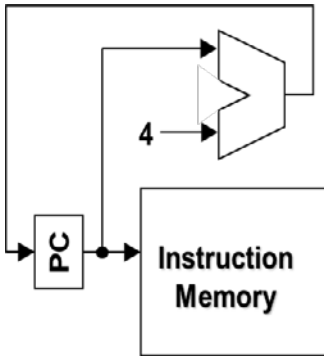


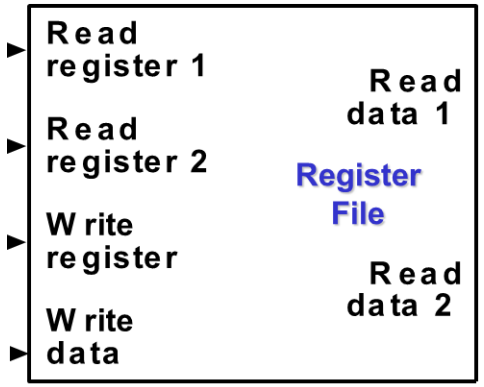
- LDUR X9, [X22, #64] //[X9] ←mem[[X22] +64]

Register	Value
22	26

Memory[90]
50

Control unit





Data mem

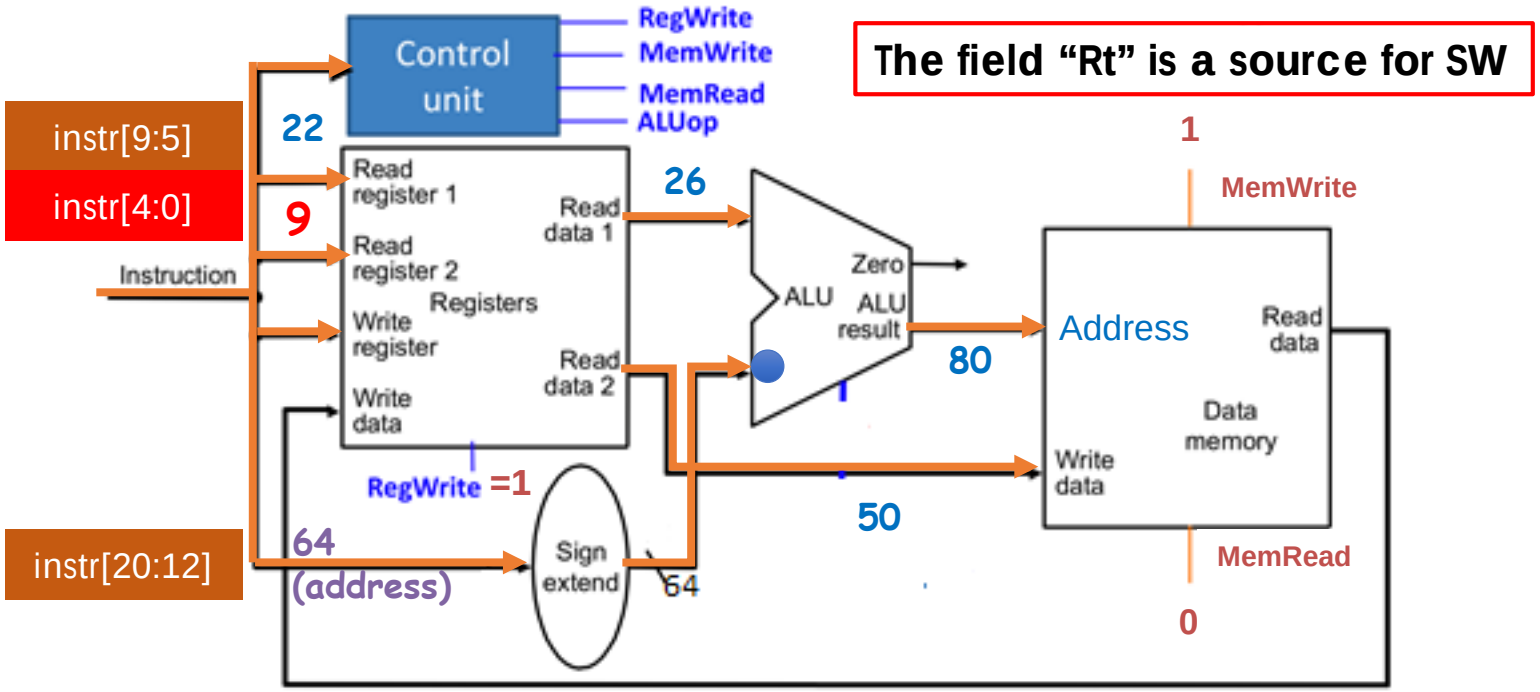
Data transfer (D)type – Base addressing)

- Operation **STUR Rt, [Rn, #address]**
- STUR X9, [X22, #64]] //[X9] →mem[[X22] +64]

1984	64	0	22	9
Opcode 11 bits	Address 9 bits	Op2 2 bits	Rn 5 bits	Rt 5 bits

Register	Value
X9	50
X22	26

Memory[80]
50



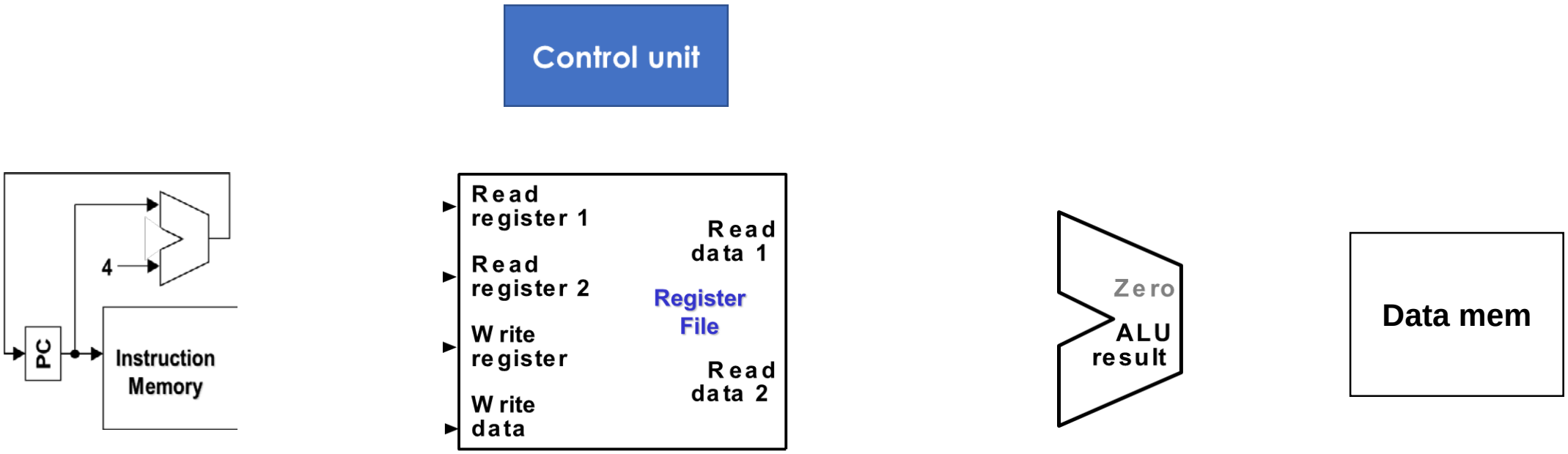
Adapted from Computer organization and design:
the hardware/software interface ARM edition , by D. A. Patterson, J. L. Hennessy & P. Alexander, 2017, Morgan Kaufmann.

How can we combine the data path for R and D type instructions?

ADD X9,X20,X21

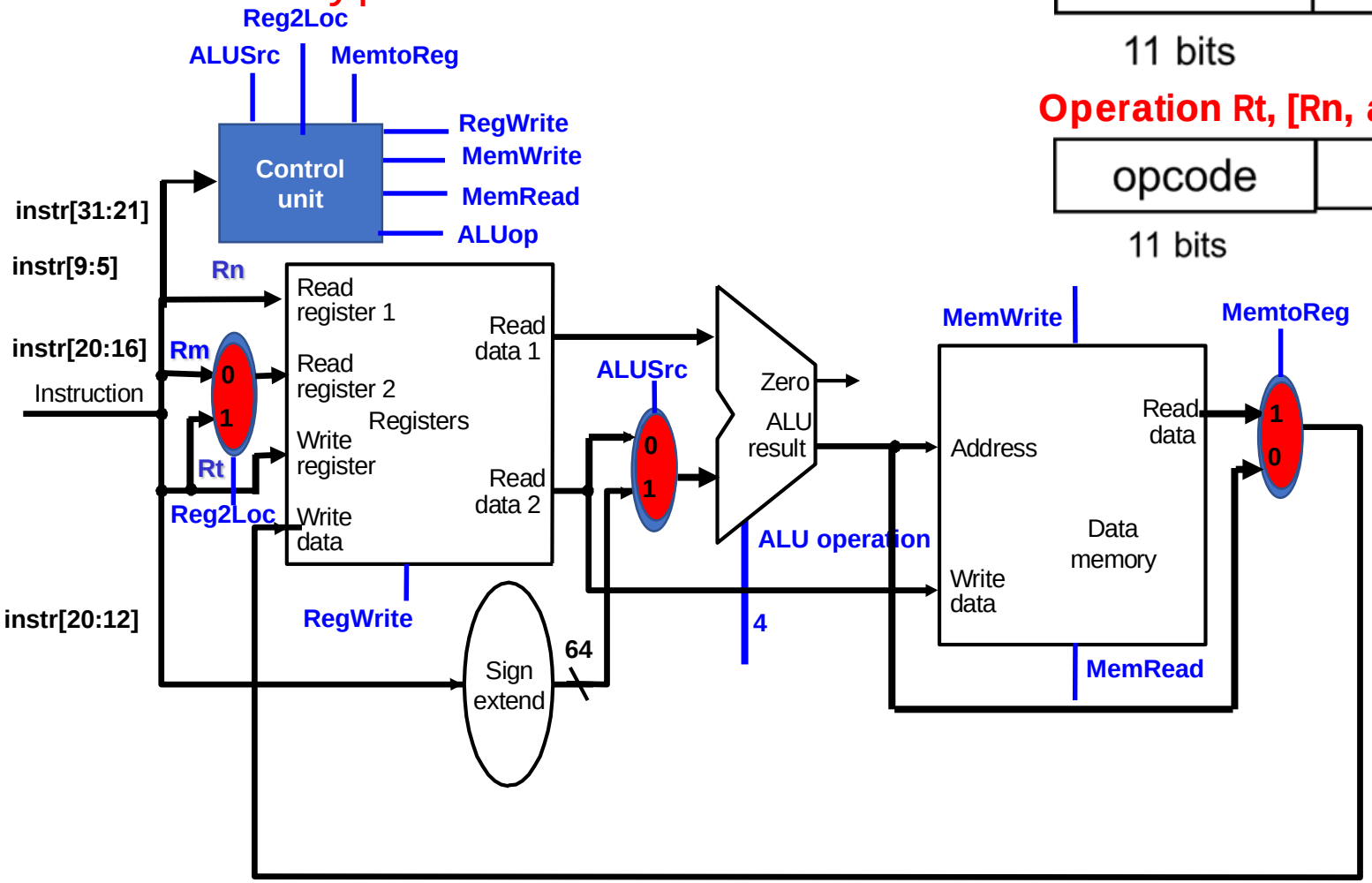
LDUR X9, [X22, #64]

STUR X9, [X22, #64]



Datapath for R and D-type Instructions

How can we combine the data path for R and D type instructions?



Operation Rd, Rn, Rm

opcode	Rm	shamt	Rn	Rd
11 bits	5 bits	6 bits	5 bits	5 bits

Operation Rt, [Rn, address]

opcode	address	op2	Rn	Rt
11 bits	9 bits	2 bits	5 bits	5 bits

"Rn" -Source for R and D type
"Rd" -Destination for R-type
"Rt" -destination/source for D-type

Extra mux needed

"Reg2Loc" -Selects between "Rt" and "Rm" as the source register address
"ALUSrc" -Selects between "read data2" and "address" as the source to ALU
"MemtoReg"-select the result from memory or from ALU

Adapted from Computer organization and design:
the hardware/software interface ARM edition , by D. A. Patterson, J. L. Hennessy & P. Alexander, 2017, Morgan Kaufmann.