Role of Computer Architecture

Design Aspects

- Instruction set design
- Functional organization
- Logic design
- Implementation

Implementation Areas

- Integrated circuit design
- Packaging
- Power management
- Cooling solutions

Required Knowledge

- Compilers
- Operating systems
- Logic design
- Packaging technologies

While instruction set architecture (ISA) was once considered the primary focus of computer architecture, the technical challenges in other aspects of design are often more significant than those in instruction set design.

Dimensions of ISA (1)(2)(3)

1 Class of ISA

Nearly all ISAs today are general-purpose register architectures. The two popular versions are register-memory ISAs (80x86) and load-store ISAs (ARM, MIPS).

2 Memory Addressing

Most computers use byte addressing. Some architectures like ARM and MIPS require aligned objects, while 80x86 does not require alignment but performs faster with aligned operands.

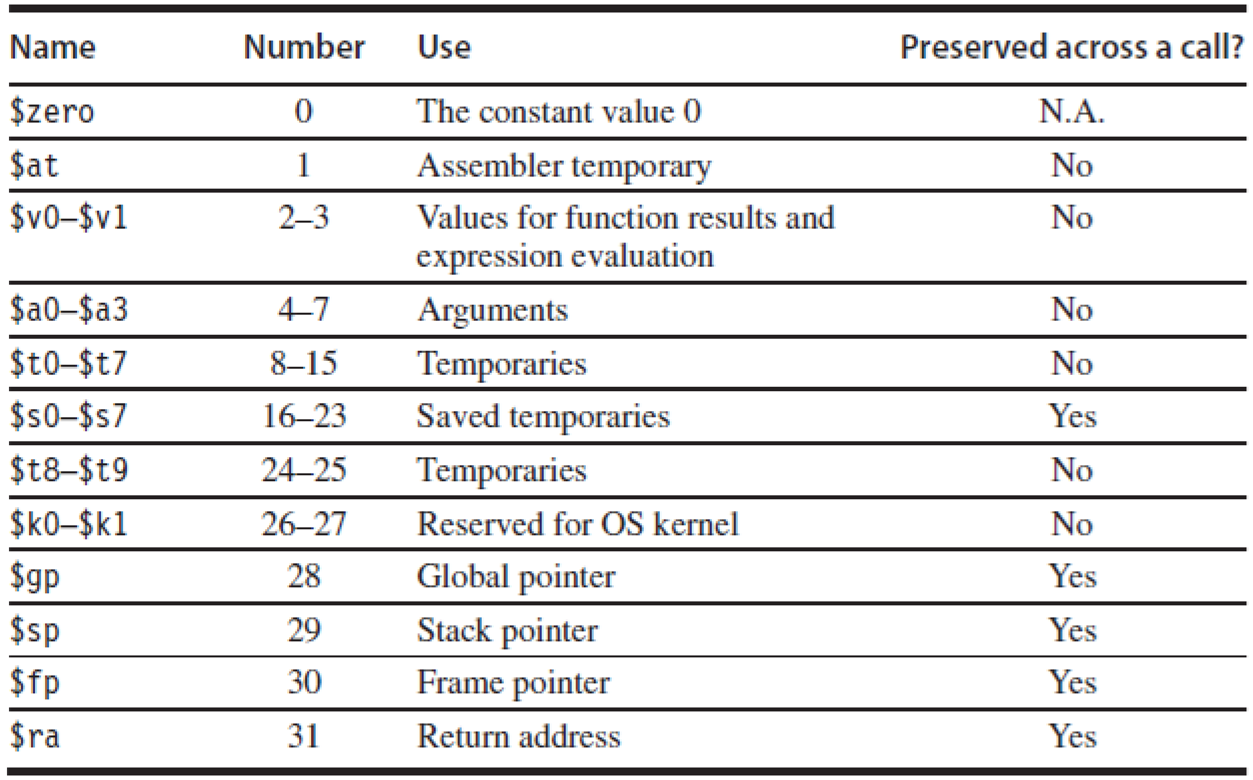
3 Addressing Modes

Different ways to specify the address of a memory object, including Register, Immediate, Displacement, PC-relative, and various combinations of registers with scaling factors.

Dimensions of ISA (4)

4 Types and Sizes of Operands

Most ISAs support 8-bit (ASCII character), 16-bit (Unicode character), 32-bit (integer), 64-bit (long integer), and IEEE 754 floating point in 32-bit and 64-bit formats.



Name	Number	Use	Preserved across a call?
`$zero`	0	The constant value 0	N.A.
`$at`	1	Assembler temporary	No
`$v0–$v1`	2–3	Values for function results and expression evaluation	No
`$a0–$a3`	4–7	Arguments	No
`$t0–$t7`	8–15	Temporaries	No
`$s0–$s7`	16–23	Saved temporaries	Yes
`$t8–$t9`	24–25	Temporaries	No
`$k0–$k1`	26–27	Reserved for OS kernel	No
`$gp`	28	Global pointer	Yes
`$sp`	29	Stack pointer	Yes
`$fp`	30	Frame pointer	Yes
`$ra`	31	Return address	Yes

KOREA UNIVERSITY

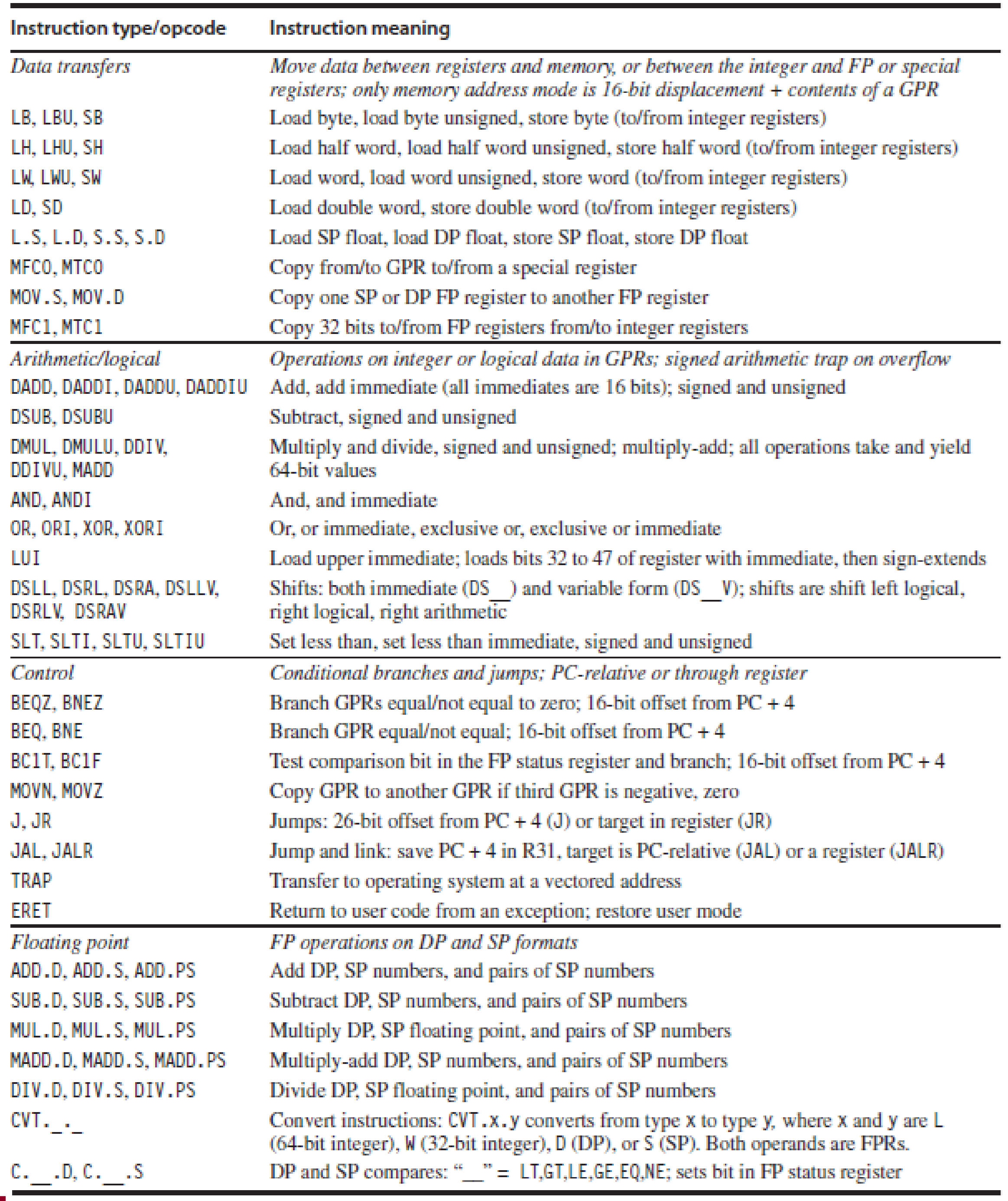
Dimensions of ISA (5)(6)

5 **Operations**

General categories include data transfer, arithmetic logical, control, and floating point. MIPS represents a simple RISC architecture, while 80x86 has a richer and larger set of operations.

6 **Control Flow Instructions**

All ISAs support conditional branches, unconditional jumps, procedure calls, and returns, with small implementation differences between architectures.



Instruction type/opcode	Instruction meaning
Data transfers	*Move data between registers and memory, or between the integer and FP or special registers; only memory address mode is 16-bit displacement + contents of a GPR*
LB, LBU, SB	Load byte, load byte unsigned, store byte (to/from integer registers)
LH, LHU, SH	Load half word, load half word unsigned, store half word (to/from integer registers)
LW, LWU, SW	Load word, load word unsigned, store word (to/from integer registers)
LD, SD	Load double word, store double word (to/from integer registers)
L.S, L.D, S.S, S.D	Load SP float, load DP float, store SP float, store DP float
MFC0, MTC0	Copy from/to GPR to/from a special register
MOV.S, MOV.D	Copy one SP or DP FP register to another FP register
MFC1, MTC1	Copy 32 bits to/from FP registers from/to integer registers
Arithmetic/logical	*Operations on integer or logical data in GPRs; signed arithmetic trap on overflow*
DADD, DADDI, DADDU, DADDIU	Add, add immediate (all immediates are 16 bits); signed and unsigned
DSUB, DSUBU	Subtract, signed and unsigned
DMUL, DMULU, DDIV, DDIVU, MADD	Multiply and divide, signed and unsigned; multiply-add; all operations take and yield 64-bit values
AND, ANDI	And, and immediate
OR, ORI, XOR, XORI	Or, or immediate, exclusive or, exclusive or immediate
LUI	Load upper immediate; loads bits 32 to 47 of register with immediate, then sign-extends
DSLL, DSRL, DSRA, DSLLV, DSRLV, DSRAV	Shifts: both immediate (DS__) and variable form (DS__V); shifts are shift left logical, right logical, right arithmetic
SLT, SLTI, SLTU, SLTIU	Set less than, set less than immediate, signed and unsigned
Control	*Conditional branches and jumps; PC-relative or through register*
BEQZ, BNEZ	Branch GPRs equal/not equal to zero; 16-bit offset from PC + 4
BEQ, BNE	Branch GPR equal/not equal; 16-bit offset from PC + 4
BC1T, BC1F	Test comparison bit in the FP status register and branch; 16-bit offset from PC + 4
MOVN, MOVZ	Copy GPR to another GPR if third GPR is negative, zero
J, JR	Jumps: 26-bit offset from PC + 4 (J) or target in register (JR)
JAL, JALR	Jump and link: save PC + 4 in R31, target is PC-relative (JAL) or a register (JALR)
TRAP	Transfer to operating system at a vectored address
ERET	Return to user code from an exception; restore user mode
Floating point	*FP operations on DP and SP formats*
ADD.D, ADD.S, ADD.PS	Add DP, SP numbers, and pairs of SP numbers
SUB.D, SUB.S, SUB.PS	Subtract DP, SP numbers, and pairs of SP numbers
MUL.D, MUL.S, MUL.PS	Multiply DP, SP floating point, and pairs of SP numbers
MADD.D, MADD.S, MADD.PS	Multiply-add DP, SP numbers, and pairs of SP numbers
DIV.D, DIV.S, DIV.PS	Divide DP, SP floating point, and pairs of SP numbers
CVT._._	Convert instructions: CVT.x.y converts from type x to type y, where x and y are L (64-bit integer), W (32-bit integer), D (DP), or S (SP). Both operands are FPRs.
C.__.D, C.__.S	DP and SP compares: "__" = LT,GT,LE,GE,EQ,NE; sets bit in FP status register

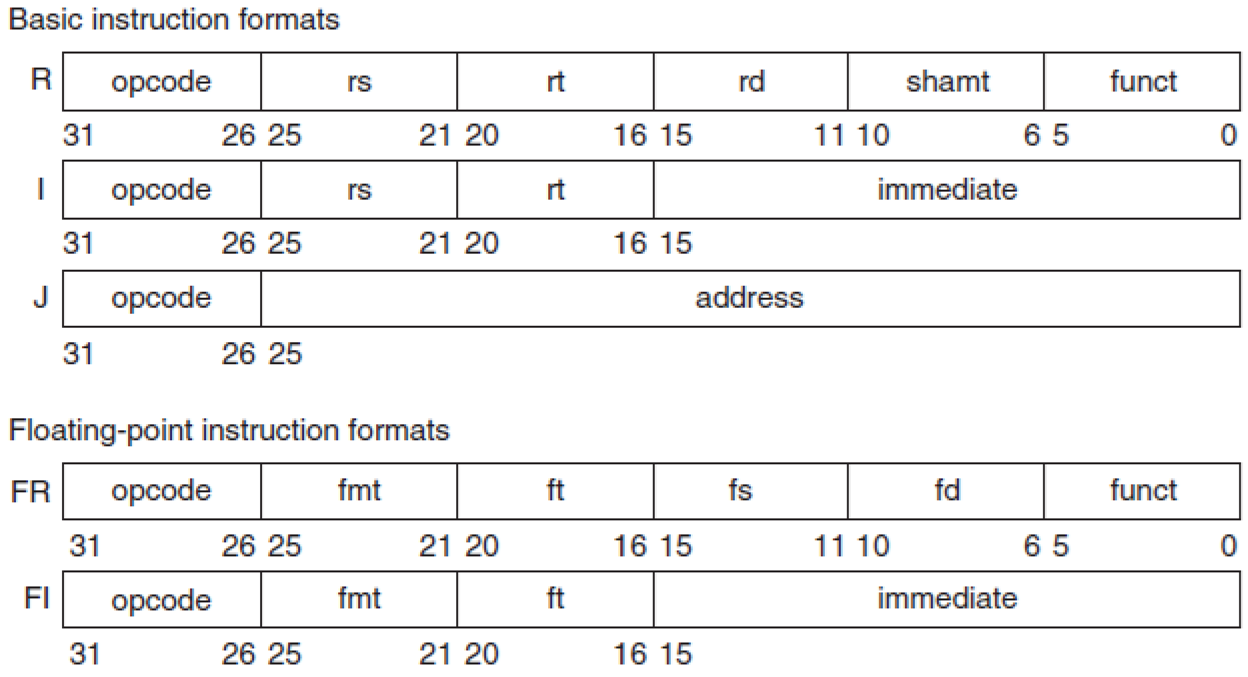
KOREA UNIVERSITY

Dimensions of ISA (7)

7

Encoding an ISA

Fixed length (ARM, MIPS: 32 bits) vs. variable length (80x86: 1-18 bytes). Variable-length instructions typically result in smaller program size.



Terminology

Organization/Microarchitecture

Includes high-level aspects of computer design:

- Memory system
- Memory interconnect
- Internal processor design (CPU)

Example: AMD Opteron and Intel Core i7 implement the same x86 instruction set but have different organizations.

Hardware

Refers to the specifics of a computer:

- Detailed logic design
- Packaging technology

Example: Intel Core i7 and Intel Xeon 7560 have nearly identical organizations but differ in clock rates and memory systems, making the Xeon more effective for servers.

The term "architecture" covers all three aspects: instruction set architecture, organization/microarchitecture, and hardware. With the shift to multiple processors per chip, the term "multicore" has replaced "multiprocessor microprocessor."

Requirements for Computer Design

1

Application Area

- Personal mobile device: Real-time performance, energy efficiency
- General-purpose desktop: Balanced performance for various tasks
- Servers: Database support, reliability, availability, scalability
- Clusters/warehouse-scale: Throughput performance, error correction
- Embedded computing: Application-specific extensions, power limitations

2

Software Compatibility

- Programming language level: Most flexible, requires new compiler
- Object code/binary compatible: Fully defined ISA, no software investment

3

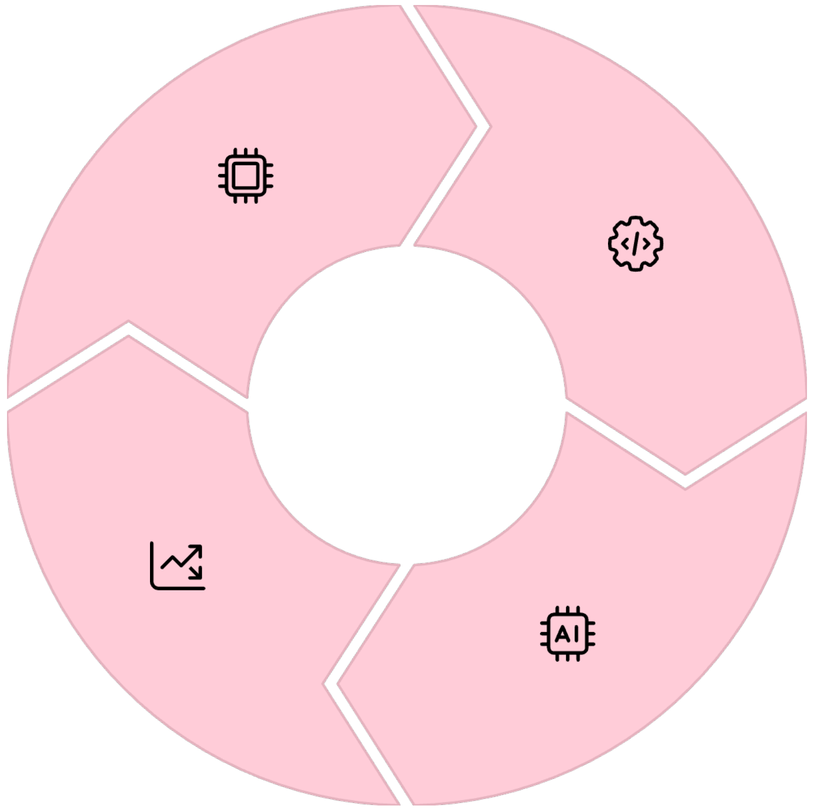
OS Requirements

- Address space size: May limit applications
- Memory management: Paged or segmented
- Protection: Page vs. segment, virtual machines

Computer architects must design systems to meet these functional requirements while balancing price, power, performance, and availability goals.

KOREA UNIVERSITY

Comprehensive View



Instruction Set Architecture

The programmer-visible instruction set that serves as the boundary between software and hardware

Organization/Microarchitecture

High-level aspects including memory system, interconnects, and processor design

Technology Trends

Awareness of important trends affecting future cost and architecture longevity

Hardware Implementation

Detailed logic design and packaging technology of the computer

Computer architecture encompasses all these aspects, going far beyond just instruction set design. Architects must balance functional requirements with price, power, performance, and availability goals while staying aware of technology trends.