

Indian Institute of Technology (IIT-Bombay)

AUTUMN Semester, 2025

COMPUTER SCIENCE AND ENGINEERING

CS230: Digital Logic Design and Computer Architecture

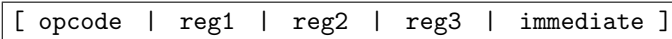
Tutorial - III

Full Marks: 0

Time allowed: ∞ hours

1. Consider a processor with 128 registers and an instruction set of size 20. Each instruction has five distinct fields, namely, opcode, two source register identifiers, one destination register identifier, and a sixteen-bit immediate value. Each instruction must be stored in memory in a byte-aligned fashion. If a program has 250 instructions, the amount of memory (in bytes) consumed by the program text is _____.
2. Assuming the system has 32-bit integers, answer the following questions.
Put the byte with the lowest address on the left (and the byte with the highest address on the right)
 - Write the decimal number 13 in Binary (Base-2) as a 32-bit Big Endian Int
 - Write the decimal number 13 in Hexadecimal (Base-16) as a 32-bit Big Endian Int
 - Write the decimal number 13 in Hexadecimal (Base-16) as a 32-bit Little Endian Int
 - Write the decimal number 13 in Hexadecimal (Base-16) as a 32-bit Little Endian Int
3. For the following, assume that values A, B, and C reside in memory. Also assume that instruction operation codes are represented in 8 bits, memory addresses are 64 bits, register addresses are 6 bits and data values are 32-bit integers.
Write down how many addresses, or names, appear in each instruction for the code to compute $C = A + B$, and what is the total code size for each of the following Instruction Set Architectures?
 - Stack
 - Accumulator
 - Register-Memory
 - Register (load-store)
4. We have 32-bit ISAs, 64-bit ISAs, well, there are also 16-bit ISAs. It's been two decades since the use of 64-bit ISAs.
 - a) What does 16-bit/32-bit/64-bit mean here?
 - b) Why not a leap towards 128-bit ISA? Yes/No? Give reasons.

5. The value represented by the hexadecimal number 4B45 5942 4F41 5244 is to be stored in an aligned 64-bit double word. The memory is byte-addressed.
- Write the value to be stored using Big Endian byte order.
 - Write the value to be stored using Little Endian byte order.
 - What are the hexadecimal values of all misaligned 2-byte words that can be read from the given 64-bit double word when stored in Big Endian byte order?
 - What are the hexadecimal values of all misaligned 4-byte words that can be read from the given 64-bit double word when stored in Little Endian byte order?
6. Consider the case of a processor with an instruction length of 12 bits and with 32 general-purpose registers, so the size of the address fields is 5 bits. Is it possible to have instruction encodings for the following?
- 3 two-address instructions
 - 30 one-address instructions
 - 45 zero-address instructions
7. Consider a **32-bit hypothetical CPU** which supports **1-word long instructions** stored in a **32KB memory**. Each instruction contains:



- The opcode field is **6 bits**.
 - There are **3 register operands**.
 - Each register field must encode **64 registers**.
 - The remaining bits go to the **immediate constant field**.
- What is the size of the immediate field?
 - What is the **largest unsigned constant** that can be represented in this instruction format?

Step 1: Word and instruction size. Word size = 32 bits. Instruction size = 1 word = 32 bits. **Step 2: Field sizes.**

- Opcode = 6 bits (given).
- Each register field must encode 64 registers: $\lceil \log_2 64 \rceil = 6$ bits.
- Three register fields $\Rightarrow 3 \times 6 = 18$ bits.

Step 3: Immediate field size.

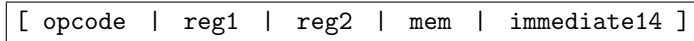
$$\text{Immediate size} = 32 - (6 + 18) = 32 - 24 = \boxed{8 \text{ bits}}$$

Step 4: Largest unsigned constant. An n -bit immediate can represent unsigned values from 0 to $2^n - 1$:

$$0 \text{ to } 2^8 - 1 = 255$$

$$\Rightarrow \boxed{255}$$

8. A hypothetical CPU supports **64 opcodes**, **256 registers**, and **32K memory cells**. Every instruction is fixed-format:



A program contains **200 instructions**. Instructions must be stored *aligned to the memory cell size*. **Note:** The immediate field is explicitly specified to be **14 bits**.

- a) Compute the *field widths* using $\lceil \log_2 N \rceil$ and hence the *instruction size in bits*.
- b) For each memory organization, give the *program size* and *internal fragmentation* (padding) **per instruction** and **in total**:
 - (i) **Byte-addressable** memory (cell = 8 bits),
 - (ii) **Word-addressable** memory (cell = 16 bits),
 - (iii) **Cell size equals the instruction size**.
