Lecture 1 — Course Introduction; Bits, Bytes and Binary

Table of contents

See csse2010[1] for staff, assessment, and course logistics — this note covers the technical content from Lecture 1.

Today's outline

- Course intro, organisation and admin
- Assessment details
- Technical content — binary numbers

Levels of abstraction

A computer can be understood at several levels of increasing abstraction (from Tanenbaum's *Structured Computer Organisation*):

1. **Level 0** — Digital Logic level (gates, circuits) — this course starts here
2. **Level 1** — Microarchitecture level
3. **Level 2** — Instruction set architecture level
4. **Level 3** — Operating system machine level
5. **Level 4** — Assembly language level
6. **Level 5** — Problem-oriented language level

[1] courses/csse2010/csse2010.pdf

The course roughly climbs this stack over the semester: hardware/digital logic first, then low-level (assembly) software, then high-level software.

Computers and binary numbers

Computers represent everything — inputs, instructions, and outputs — as binary data internally, e.g. `10100101011010100111101010100101110001101010 1`.

See binary-number-representations[2] for the full reference on bits, bytes, unsigned number representation, converting between decimal/binary/octal/hex, and radix notation conventions — all introduced in this lecture.

Reminders

- Sign up to a pair of lab sessions (P2, P1).
- Labs start from **Thursday** of week 1 (P1 sessions only — no P2 sessions in week 1).
- From week 2, both P2 and P1 sessions run.
- Lab 1 covers different binary representation formats for **signed** numbers (two's complement etc. — not covered in this lecture).
- A summary recording on binary numbers will be posted by the end of week 1.
- Lab sign-on issues: email eait.mytimetable@uq.edu.au.

[2] courses/csse2010/binary-number-representations.pdf