

# CS132-15 Computer Organisation & Architecture

**24/25**

**Department**

Computer Science

**Level**

Undergraduate Level 1

**Module leader**

Richard Kirk

**Credit value**

15

**Module duration**

10 weeks

**Assessment**

Multiple

**Study location**

University of Warwick main campus, Coventry

---

## Description

### Introductory description

The module focuses on computer system hardware and how it is used to facilitate the execution of software, including topics such as data representation, digital logic, assembler, memory systems, I/O and microprocessor architecture.

### Module aims

The aim of the module is to provide students with an understanding of the functional components of a computer system and how they are organised to facilitate execution. Student will also gain a range of practical skills in the application and construction of computer components that are capable of interfacing with microprocessors.

### Outline syllabus

This is an indicative module outline only to give an indication of the sort of topics that may be covered. Actual sessions held may differ.

Data Representation: Analog and digital data; number systems including number representations

and conversions; binary variables including logic expressions and simplifications of binary variables; binary arithmetic, signed numbers; arithmetic structures, multiple precision arithmetic, floating point representation and manipulation.

Logic: Logic expressions, logic gates; Boolean algebra, logical expression simplification; combinatorial logic including implementation of adders/subtractors, multiplexers/de-multiplexers, encoders/decoders; sequential logic including implementation of storage elements, registers, shift-registers; bus structures.

Assembler: Microprocessor architecture; instruction cycles, register transfer language (RTL); instruction set and addressing modes; assembly and machine code programming.

Input/Output Mechanisms: Parallel I/O; memory-mapped and programmable I/O devices, eg VIA; I/O scheduling including programmed I/O, interrupt driven I/O, direct memory access (DMA); analog I/O.

Memory systems: Motivation for memory hierarchy; devices including disk formats, hard disks, optical data storage; elementary error detection and correction.

Processor Architecture: Example combination of ALU, registers, and CPU; instruction set construction; control units including hardwired and microprogrammed control units; performance of microprocessors; RISC/CISC architectures.

## **Learning outcomes**

By the end of the module, students should be able to:

- Understand the role data representation in facilitating computation
- Understand the operation of electronic logic elements
- Understand the organisation of a computer system in terms of its components
- Understand the detailed operation of a microprocessor
- Understand the different characteristics of processor architectures
- Understand mechanisms for input/output (I/O)
- Understand the role and operation of the system memory hierarchy
- Have practical experience of prototyping digital circuits
- Be able to interface digital circuits to microprocessor systems

## **Indicative reading list**

Please see Talis Aspire link for most up to date list.

[View reading list on Talis Aspire](#)

## **Research element**

Students are required to conduct independent research as part of their laboratory work and documentation. In addition, 20-40% of Coursework 1 is based on research and working with academic sources.

## **Subject specific skills**

Systems analysis and design.

Systems-level programming.  
Software documentation and maintenance.  
Logic circuit design.  
Microprocessor interfacing and design.

## Transferable skills

Technical - Expertise in the analysis and design, operation of computer system hardware. An understanding of systems programming, including the ability to use a systems programming language, and how to interface with a microprocessor.

Communication - Lecture listening. Technical report writing. Technical document comprehension and analysis. Documenting software solutions.

Critical Thinking - Systems analysis and technical problem solving. Quantitative performance analysis based.

Multitasking - Management of competing deadlines and priorities. Documentation- and test-driven development.

Teamwork - Working as part of a technical team in contributing to the development and documentation of a solution.

Creativity - Developing an original solution to a problem with little guidance on the nature or form of the approach that should be taken.

Leadership - Combining teamwork, critical thinking and technical understanding in the development of a software solution.

---

## Study

### Study time

Type	Required	Optional
Lectures	30 sessions of 1 hour (20%)	
Practical classes	8 sessions of 2 hours (11%)	8 sessions of 2 hours
Private study	104 hours (69%)	
Total	150 hours	

### Private study description

Background reading:

Clements A, The Principles of Computer Hardware, Oxford, 2006.

Burrell M, Fundamentals of Computer Architecture, Palgrave Macmillan, 2004.

Tanenbaum AS, Structured Computer Organization, Prentice Hall, 2006.

Möldner T, C for Java Programmers, Addison Wesley, 2000.

Coursework-related activities:

Reading, programming, systems design and project management.

## Self-tests:

- Data Representation
- Digital Logic
- Assembler
- Input / Input Mechanisms
- Memory Systems
- Microprocessor Architecture

## Revision:

Data Representation: Analog and digital data; number systems including number representations and conversions; binary variables including logic expressions and simplifications of binary variables; binary arithmetic, signed numbers; arithmetic structures, multiple precision arithmetic, floating point representation and manipulation.

Digital Logic: Logic expressions, logic gates; Boolean algebra, logical expression simplification; combinatorial logic including implementation of adders/subtractors, multiplexers/de-multiplexers, encoders/decoders; sequential logic including implementation of storage elements, registers, shift-registers; bus structures.

Assembler: Microprocessor architecture; instruction cycles, register transfer language (RTL); instruction set and addressing modes; assembly and machine code programming.

Input/Output Mechanisms: Parallel I/O; memory-mapped and programmable I/O devices, eg VIA; I/O scheduling including programmed I/O, interrupt driven I/O, direct memory access (DMA); analog I/O.

Memory Systems: Motivation for memory hierarchy; devices including magnetic tape, floppy disks, disk formats, hard disks, optical data storage; elementary error detection and correction.

Processor Architecture: Example combination of ALU, registers, and CPU; instruction set construction; control units including hardwired and microprogrammed control units; performance of microprocessors; RISC/CISC architectures.

## Costs

No further costs have been identified for this module.

---

## Assessment

You do not need to pass all assessment components to pass the module.

Students can register for this module without taking any assessment.

### Assessment group D4

Weighting	Study time
Coursework 2 - Group Technical Report 25%	
Coursework 2 Group Technical Report - This assignment is a group technical report based on the	

## Weighting

## Study time

work done within labs. This assignment is worth more than 3 CATS, so is not eligible for self-certification.

Coursework 1 - Problem Sheet 15%

Coursework 1 Problem Sheet - This assignment is a problem sheet based on some of the key knowledge gained in the first half of the module. This assignment is worth less than 3 CATS, so is eligible for self-certification via extension.

In-person Examination 60%  
CS132 Exam

- 
- Answerbook Pink (12 page)
  - Students may use a calculator

## Assessment group R3

## Weighting

## Study time

In-person Examination - Resit  
CS132 Resit Exam

100%

- 
- Answerbook Pink (12 page)

## Feedback on assessment

Feedback for coursework via Tabula and in lab sessions.

[Past exam papers for CS132](#)

## Availability

## Post-requisite modules

If you pass this module, you can take:

- ES2E3-15 Digital Systems Design

## Courses

This module is Core for:

- UCSA-G500 Undergraduate Computer Science
  - Year 1 of G500 Computer Science
  - Year 1 of G500 Computer Science
- UCSA-G503 Undergraduate Computer Science MEng
  - Year 1 of G500 Computer Science
  - Year 1 of G503 Computer Science MEng
  - Year 1 of G503 Computer Science MEng
- Year 1 of UCSA-I1N1 Undergraduate Computer Science with Business Studies
- Year 1 of UCSA-G406 Undergraduate Computer Systems Engineering
- Year 1 of UCSA-G408 Undergraduate Computer Systems Engineering

This module is Option list B for:

- UCSA-G4G1 Undergraduate Discrete Mathematics
  - Year 1 of G4G1 Discrete Mathematics
  - Year 1 of G4G1 Discrete Mathematics
- Year 1 of UCSA-G4G3 Undergraduate Discrete Mathematics