

System-on-Chip Design Engineering Foundations

Engineering workflows, scripting, electronics, HDL and manufacturing

FA-0797

Embedded Systems and Hardware · Software Development

Introduction

A modular engineering programme combining project delivery, Linux and Docker, Python and Tcl, digital logic, electronics, Verilog design, PCB workflows and manufacturing considerations. Develop foundational skills through selected simulations, scripting exercises and guided design projects.

The programme introduces tools and methods used around hardware development. It does not substitute for a complete advanced chip-design, sign-off or manufacturing qualification. Component tracks retain their individual allocations where specified.

Learning outcomes

- Plan engineering work and apply lean and project-management practices.
- Use Linux, Bash, Docker, Python and Tcl for engineering workflows.
- Model signals and control responses and create a small Python engineering project.
- Analyse basic circuits and design and simulate combinational and sequential logic.
- Create introductory Verilog modules, test examples and state-machine designs.
- Develop PCB schematics, layouts, checks and manufacturing outputs using a compatible prepared toolchain.
- Compare process-selection and design-for-manufacturing considerations.

Prerequisites

For electrical/electronics engineering learners and practitioners building design-tool skills. Familiarity with basic mathematics and electrical concepts is useful. Use prepared compatible software and lab or simulation environments for the selected tracks.

Training outline

1. Project management and lean – 2 days

Day 1

- Introduction to project management
- What is project management?
- The project management process
- The different types of project management roles
- Lean methodology
- What is lean methodology?
- The principles of lean methodology
- The tools and techniques of lean methodology
- Case study: applying lean methods to an electrical-engineering project

Day 2

- Project planning
- How to create a project plan
- How to estimate project costs and resources
- How to manage project risks
- Project execution
- How to communicate with project stakeholders
- How to resolve project conflicts
- How to measure project progress
- Project closeout

- How to document project lessons learned
- How to hand off project deliverables

2. Linux, Bash and Docker — 2 days

Day 1: Introduction to Linux

Session 1: Linux Basics

- Introduction to Linux: History and Importance
- Linux distributions: supported Ubuntu or Fedora releases; CentOS Linux as historical context
- Installation of Linux (Using a Virtual Machine)
- Basic Linux commands (ls, cd, mkdir, etc.)
- Exercise: Navigating through directories and creating files

Session 2: Advanced Linux Concepts

- File permissions and ownership
- Package management using distribution-appropriate tools such as apt or dnf
- Processes and services (using ps, top, systemctl)
- Exercise: Setting file permissions and installing packages

Session 3: Linux Networking and Shell Scripting

- Linux networking basics with ip, ss and ssh; recognising older ifconfig/netstat tools
- Introduction to Bash scripting
- Writing basic scripts (variables, loops, conditionals)
- Exercise: Creating and executing simple bash scripts

Session 4: Linux System Administration

- Disk and system monitoring tools (df, du, free)
- User and group management
- System logs and troubleshooting
- Exercise: User creation, monitoring system logs and resources

Day 2: Docker Introduction and Hands-On

Session 1: Docker basics

- What is Docker? Why Docker?
- Docker architecture (Images, Containers, Docker Daemon)
- Installing Docker on Linux
- Running your first Docker container
- Exercise: Running a simple Docker container

Session 2: Working with Docker Containers

- Docker Commands (run, ps, stop, rm)
- Interacting with Docker containers (exec, logs)
- Docker images (pull, rmi, build)
- Exercise: Interacting with Docker containers and images

Session 3: Dockerfile and Docker Networking

- Introduction to Dockerfile: creating a Dockerfile
- Building images with Dockerfile
- Docker networking basics (port mapping, networking modes)
- Exercise: Building a Docker image using a Dockerfile and setting up networking

Session 4: Docker Compose and practices

- Introduction to Docker Compose
- Running multi-container applications with Docker Compose
- Docker storage and volumes
- Docker best practices
- Exercise: Running a multi-container application using Docker Compose

3. Python for electrical engineering — 4 days

Day 1: Introduction to Python and Basics

- Introduction to Python
- What is Python? Why Python for Electrical Engineers?

- Installing Python
- Setting up the Development Environment (IDE)
- Python Basics
- Variables, Data Types and Operators
- Basic Input and Output
- Conditional Statements (if-elif-else) and Looping Constructs (for, while)
- Data Structures
- Lists, Tuples, Sets, and Dictionaries
- Introduction to NumPy
- What is NumPy? Why is it important for Electrical Engineers?
- Creating Arrays, Array Operations
- Basic Linear Algebra Operations with NumPy

Day 2: Functions, File Handling and Plotting

- Functions
- Defining Functions
- Function Arguments and Return Values
- File Handling
- Reading and Writing Files
- Parsing Data from CSV and Text Files
- Data Visualization with Matplotlib
- Creating basic plots (line, scatter, bar)
- Multiplots and customization
- Plotting Electrical Signals
- Introduction to SciPy
- Integration and Differential Equations
- Fourier Transform

Day 3: Advanced Topics and Project

- Object-Oriented Programming
- Classes and Objects
- Inheritance, Encapsulation, Polymorphism
- Signal Processing with SciPy
- Signal Generation
- Filtering and Convolution
- Control Systems using Python Control Systems Library
- Transfer Functions
- Step Response, Impulse Response
- Stability Analysis
- Project
- Application Project: Building a simple signal processing or control system project, utilizing concepts learned throughout the course.

Day 4: Signal Processing, Control Systems, and Wavetable Synthesis

- Signal Processing with SciPy
- Signal Generation
- Filtering and Convolution
- Control Systems using Python Control Systems Library
- Transfer Functions
- Step Response, Impulse Response
- Stability Analysis
- Introduction to Wavetable Synthesis
- Understanding Wavetable Synthesis
- Creating basic waveforms (sine, square, sawtooth, etc.)
- Building a Wavetable Synthesizer in Python
- Generating and manipulating wavetables

- Adding features like amplitude envelope, frequency modulation
- Saving synthesizer output to audio files

4. Tcl programming — 1 day

Session 1: Introduction to Tcl

- Overview of Tcl
- What is Tcl?
- History of Tcl
- Where and Why is Tcl Used?
- Tcl Installation and Setup
- Interactive Tcl Shell
- Writing and Running a Tcl Script

Session 2: Tcl Basics

- Tcl Variables and Data Types
- Variables
- Strings
- Numbers
- Basic Syntax
- Comments
- Quotes and Braces
- Commands and Arguments
- Control Flow in Tcl
- If, else, elseif
- For loops
- While loops

Session 3: Tcl Advanced Features

- Procedures in Tcl
- Defining and Calling Procedures
- Arguments and Return Values
- Lists and Arrays
- Creating and Manipulating Lists
- Array Usage and Operations
- String Manipulation
- String Comparison
- Pattern Matching and Regular Expressions

Session 4: File I/O and Tcl Libraries

- File I/O Operations in Tcl
- Opening and Closing Files
- Reading and Writing Files
- Tcl Libraries and Packages
- Loading Libraries
- Using Common Libraries like Tcllib
- Exception Handling in Tcl
- Catch Command
- Error Command

Session 5: Hands-On Exercises and Wrap Up

- Hands-On Exercise: Building a small Tcl application integrating the concepts learned
- Best Practices in Tcl Programming
- Wrap up and Q&A

5. Digital electronics and logic design — 3 days

Day 1: Number Systems and Boolean Algebra

- Session 1: Number Systems
 - Introduction to Number Systems: Binary, Decimal, Octal, Hexadecimal
 - Conversion between Number Systems
 - Signed Numbers: One's Complement, Two's Complement
 - Binary Arithmetic: Addition, Subtraction, Multiplication, Division
- Session 2: Boolean Algebra
 - Introduction to Boolean Algebra
 - Boolean Operators: AND, OR, NOT, NAND, NOR, XOR, XNOR
 - Boolean Laws and Properties
 - Canonical and Standard Forms

Day 2: Boolean Functions, K-Maps and Logic Gates

- Session 1: Boolean Functions
 - Representation of Boolean Functions
 - Simplification of Boolean Functions using Boolean Algebra
 - Truth Tables
- Session 2: K-Maps
 - Introduction to K-Maps
 - Simplifying Boolean Functions using K-Maps
 - Handling Don't-Care Conditions
- Session 3: Logic Gates
 - Introduction to Logic Gates: AND, OR, NOT, NAND, NOR, XOR, XNOR
 - Truth Tables and Logic Symbols
 - Implementing Boolean Functions using Logic Gates
- Session 4: Designing Circuits in Logisim
 - Building Basic Circuits with Logic Gates in Logisim

- Simulating and Testing Circuits in Logisim
- Creating and Using Subcircuits in Logisim

Day 3: Combinational and Sequential Logic Circuits

- Session 1: Combinational Logic Circuits
- Introduction to Combinational Logic Circuits
- Design of Basic Combinational Circuits: Adders, Subtractors, Multiplexers, Decoders, Encoders
- Session 2: Sequential Logic Circuits
- Introduction to Sequential Logic Circuits
- Flip-Flops: SR, D, JK, T
- Design of Counters, Registers, and Memory Units
- Session 3: Hands-On Design and Testing
- Designing Combinational and Sequential Logic Circuits using a logic simulator (such as Logisim or similar)
- Debugging and Testing of Logic Circuits

6. Electronics, circuits and integration foundations – 5 days

Day 1: Introduction to Electronics

- Session 1: Basics of Electronics
- Basic Electronic Components (Resistors, Capacitors, Diodes, Transistors)
- Ohm's Law and Kirchhoff's Laws
- Series and Parallel Circuits
- Session 2: Introduction to Electrical Engineering
- The role of Electrical Engineering in Technology
- The scope of Electrical Engineering
- Different fields within Electrical Engineering

Day 2: Advanced Circuit Analysis Techniques and Tools

- Session 1: Advanced Circuit Analysis Techniques

- Thevenin's, Norton's, and Maximum Power Transfer Theorems
- Superposition Principle
- AC circuit analysis, Phasors
- Session 2: Circuit Analysis Tools
- Introduction to Circuit Simulation Tools
- Simulating and Analyzing Circuits with SPICE or other similar tools

Day 3: Introduction to Digital Logic Systems, Boolean Algebra, Timing Diagrams, and TTL

- Session 1: Digital Logic Systems and Boolean Algebra
- Introduction to Digital Logic
- Boolean Algebra and Logic Gates
- Logic Simplification using K-maps
- Session 2: Timing Diagrams and TTL (Transistor-Transistor Logic)
- Understanding and Creating Timing Diagrams
- Introduction to TTL
- Working with TTL Logic Gates

Day 4: Taking Digital to the Next Level with Small, Medium, and Large Scale Integration

- Session 1: Integrated Circuits and SSI
- Introduction to Integrated Circuits (ICs)
- Understanding Small-Scale Integration (SSI)
- Practical Applications of SSI
- Session 2: Medium and Large Scale Integration
- Introduction to Medium-Scale Integration (MSI)
- Introduction to Large-Scale Integration (LSI)
- Comparing SSI, MSI, and LSI

Day 5: Printed Circuit Board Design and Technology with CircuitMaker

- Session 1: Introduction to PCB Design and CircuitMaker
 - Basics of PCB Design
 - Introduction to CircuitMaker
 - Navigating CircuitMaker Interface
- Session 2: CircuitMaker Fundamentals and Real-World Projects
 - Creating Schematics in CircuitMaker
 - Designing PCB Layout in CircuitMaker
- Guided project: creating a small schematic and PCB design using CircuitMaker

7. Verilog HDL design and verification

Day 1: Introduction and Designing with Verilog

- Session 1: Introduction to Verilog HDL and Verilog Data Types and Operators
 - Brief history and application of Verilog HDL
 - Understanding various data types and operators in Verilog
- Session 2: Verilog Module and Verilog Design Styles
 - Structure and importance of a Verilog module
 - Different design styles and when to use them
- Session 3: Verilog Structural Design
 - Basics of structural design in Verilog
 - Designing example digital circuits using structural modelling
- Session 4: Verilog Combinational Design
 - Basics of combinational design in Verilog
 - Hands-on session: Designing a combinational circuit
- Session 5: Verilog Sequential Design
 - Basics of sequential design in Verilog
 - Hands-on session: Designing a sequential circuit

Day 2: Advanced Design and Verification with Verilog

- Session 1: Verilog Functions and Tasks
 - Understanding functions and tasks in Verilog
 - Application of functions and tasks in design verification
- Session 2: Verilog Memory Design
 - Understanding memory components in Verilog
 - Hands-on session: Designing a memory component
- Session 3: Verilog State Machines
 - Introduction to state machines in Verilog
 - Hands-on session: Designing a state machine
- Session 4: Verilog Design Examples
 - Real-world examples of digital designs using Verilog
 - Participants to practice designs under guided supervision
- Session 5: Conclusion and Q&A
 - Recap of Verilog HDL fundamentals and digital design techniques
 - Open session for questions, discussion, and feedback

8. PCB layout and fabrication extension

- Session 1: Assigning Footprints to Schematic Components
 - Linking Footprints to Components in the Schematic
 - Checking Footprint Assignments
- Session 2: Placement & Routing in PCB Layout
 - Importing Schematic to PCB Layout
 - Component Placement Strategies
 - Manual and Auto-Routing Techniques
 - Design Rule Checking (DRC)
- Session 3: Generating BOM and Gerber Files
 - Introduction to BOM (Bill of Materials)

- Generating and exporting a BOM using the selected compatible PCB design tool
- Generating Gerber Files for Fabrication
- Session 4: Optimizing the Board and Generating Complex Footprints
- Evaluating Design for Potential Improvements
- Implementing Design Optimizations
- Creating a Footprint for a Complex Component

9. Manufacturing process selection and DFM – 2 days

Day 1: Manufacturing Process Selection

- Session 1: Introduction to Manufacturing Process Selection
- Basics of process selection
- Factors influencing process selection
- Session 2: Process Selection Criteria
- Technical feasibility
- Economic feasibility
- Environmental impact
- Quality implications
- Session 3: Case Study 1 in Process Selection
- Study and discussion of an engineering process-selection example
- Session 4: Case Study 2 in Process Selection
- Study and discussion of a second process-selection example
- Session 5: Discussion and Wrap-up
- Open discussion on the day's topics
- Recap and preparation for the next day

Day 2: Design for Manufacturing

- Session 1: Introduction to Design for Manufacturing (DFM)
- Overview of DFM

- Importance of DFM in the manufacturing process
- Session 2: Design for Forging
- Specifics of designing for the forging process
- Considerations and best practices
- Session 3: Design for Casting
- Specifics of designing for the casting process
- Considerations and best practices
- Session 4: Design for Injection Molding
- Specifics of designing for the injection molding process
- Considerations and best practices