# Section 3

# Course Delivery Details

## 3.1 Weekly Course Structure and Workload

| Sesssion | Time | Duration - Minutes | Content |
|---|---|---|---|
| Presentation | 4pm | 50 | Online Live Lecture |
| Presentation | 5pm | 50 | Online Live Lecture |
| Coffee | 6pm | 15 | Rest and informal live Q&A |
| Workshop | 6.15pm | 45 | **Elective** Workshop - Working together to design or code or understand an assignment – Instructor led. This is an elective for students that may need extra coaching/tuition to ascend the learning curve initially. |
| Study and Reading Time | Participant Chosen | 1 hours | Second reading of course material and/or viewing of recorded lecture. Involvement in technical discussion forum. |
| Design Work | Participant Chosen | 8 hours | Homework Assigments - hands on Experience of designing and coding and synthesis. |
| Assessment Feedback | 5 minutes per student | – | 1x1 Individual student feedback per week. |
| Total Student Workload | – | 11 hours | For average student - some will be less and some more. |
| Total Lecturer Workload | | 8 hours | 3 Hours – Online Lectures 2 Hours – Online Discussion 1.75 – Hours Marking/Grading Homework/Assignments, 7 minutes per student 1.25 Hours – 1x1 Feedback each week on average |

Table 3.1: Indicative Weekly Course Plan

An indicative weekly course delivery structure is given in table 3.1. The durations given are approximate and will vary depending on class size and class competence and the number of questions arising.

## 3.2 Course Content Outline

| Section Name | Content |
|---|---|
| Context and Background Material | State of the art in Digital Design Technology, CPU, SOC, ASIC, FPGA, Memory. Overview of Digital Systems Design, Flow and Tools.<br>Review of Digital Logic Basic Elements - Logic Gates, Multiplexor, Flip-Flop, decoders. Use of Timing Diagrams to test behaviour of logic citcuits. |
| Verilog1 – Essentials | Introduction to Basic Verilog Modelling of combinational logic and synchronous logic. Different coding methods. |
| Verilog2 – FSM Coding | Verilog Coding of FSMs - different styles of coding. |
| Verilog3 – Testbenches | Coding of a basic functional testbench. Correct Application of stimulus to UUT. Self-Checking testbench. Building up testbench complexity. |
| Verification1 | Test Specifications. Unit testing and system testing. Use of Linting. Testing Silicon - Functional testing, BIST, SCAN and production testing. Description of Verification Methodologies - UVM. Verilog and UPF in verification. |
| Digital Design1 – FSMs | Introduction to Finite State Machines (FSMs). Logic derivation from FSMs. Application of FSMs to design solutions to various problems e.g. simple UART receive and transmit, Sequence Detect, Memory Controller, Memory Arbitration. |
| Digital Design2 – Top Down1 | Introduction to Top Down Design Methodology and application to Design of 16x8 FIFO from Flips-Flops or similar. |
| Digital Design3 - Bus Protocols | ARM and AMBA, APB, AHBLite and AXI4 buses, how they work and compare. Design of AMBA bus DMA controller using one of the AMBA protocols. |
| Digital Design4 - Timing Analysis1 | Timing Analysis(STA) of digital blocks. Minimum and maximum time calculations. Chip I/O timing and PLL application. |
| Digital Design5 - Asynchronous1 | Single bit Asynchronous Inputs to FSMs and synchronization methods. Metastability and Reliability of single bit and double synchronisation methods. Design Method to safely facilitate bus crossing asynchronous interface – method 1. |
| Digital Design6 - Asynchronous2 | Efficient Data Bus Crossing of asynchronous interfaces using FIFO. |
| Digital Design7 - Top Down2 | Design of a simple Microprocessor or similar project. |
| Digital Design8 - Realisation1 | FPGAs - Overview of the architecture and building blocks of XILINX, Altera(Intel) and Microsemi FPGAs. Comparison of FPGA to ASIC development. IO Standards, DDR Memory and DDR interfacing. |

Table 3.2: Outline of Course Lecture Content

### 3.2.1   Indicative Assessment Content

This module will be assessed using 100% Continuous Assessment(CA). Some of the indicative homework and type of assignments/assessments are shown in table 3.3.

| Section Name | Content |
| --- | --- |
| Assignment1 | **Instructor Led** – Verilog Coding of a logic schematic with switches and LEDs. Writing verilog self-checking testbench to test schematic. Synthesis(with basic constraints), Place and route(PAR) and download to FPGA (using Vivado). |
| Assignment2 | **Participant Led** – Design an FSM to solve simple problem e.g. Transmit UART. Coding and self-checking testbench to test this design. |
| Assignment3 | Design, Document, Code and test of 16x8 FIFO. |
| Assignment4 | Design, Document, Code and Test of AXI4 DMA Controller. |
| Assignment5 | Design, Document, Code and Test of memory controller to simple asynchronous SRAM. Code and test this design including bus functional model (BFM) of the SRAM. |
| Assignment6 | Design, Document, Code and Test of multi-master memory controller to simple asynchronous SRAM. Arbitration using priority and round robin basic arbiters |
| Assignment7 | Design, Document, Code and Test of asynchronous interface clock crossing with a simple bus crossing method |
| Assignment8 | Capstone Project - Design, Code and test of a simple Microprocessor or similiar |

Table 3.3:   Indicative - Course Assessment Content