

# Logic Design and Computer Introduction

83-253

Shmuel wimer

## Course format: Lectures and Training

First Semester 2016/17 **Weekly hours:** 2 lecture + 1 training + 1 lab

### 1) **Course objectives:**

The digital computing is a fundamental component in almost every technological system. This course provides a basic knowledge of how an ordinary digital computer is working, its internal computational processes, their control, its programming and how the computer interfaces with the world. An emphasis is put on engineering, design and performance considerations.

### 2) **Course format:**

Frontal lectures – in the classroom.

Frontal training – in the classroom.

Verilog workshop – in lab.

### 3) **Course content:**

Week	Topic	Reading
1	Principles of system design, the digital abstraction, combinational and digital systems, timing.	
2	Pipeline, performance, latency, throughput, precedence, retiming, the pipeline graph.	
3	Regular structures, switches and gates, tri-state, busses, PLAs, ROM, RAM, faults detection.	
4	Data-path and control, components of data-path, timing in data-path, design examples, busses and switches, synchronization, precedence, serialization, communication, handshaking.	
5	Simple CPU, registers, RTL, instruction set, assembly language and programming, the boot program, microinstructions, control signals generation.	
6	MIPS instructions, high-level programming, assembly and machine code, MIPS registers, instructions format, R-type, I-type, memory organization, the program counter, branches and jump instructions, ALU internals.	
7-8	Single cycle MIPS, the data-path, register-file internals, instruction stages, fetching, decoding, memory instructions, jumps, ALU control, instruction execution and timing, critical	

	paths and performance.	
9-10	Multi-cycle MIPS, data-paths, registers, control, microinstructions, performance analysis, control's state machine, ROM and PLA implementations, control logic, microcode and microprogramming.	
11	Pipelined MIPS, combining single cycle and multi cycle, the pipe, pipeline registers, fetch, decode, execute, memory, write-back, register support for control signals.	
12	Data hazards, bypass, forwarding, NOPs and bubbles, control hazards, flushing.	
13	Exceptions.	
14	Memory hierarchy, caches.	

4) **Prerequisites:**

Digital Logic Systems.

5) **Course requirements:**

Biweekly homework assignments. Delivery of all assignments minus one is mandatory and must be completed before term A exam.

Delivery of Verilog assignments is mandatory. Each undelivered assignment will trip 2 points from the final grade.

6) **Grading:**

Final exam: 80% ; Homework exercises: 20% ; Pass grade (60) in the final exam is mandatory.

7) **Textbooks and supplementary reading:**

D. A. Patterson and J. L. Hennessy, Computer Organization and Design, 3<sup>rd</sup> Edition, Morgan Kaufmann.

D. M. Harris and S. L. Harris, Digital Design and Computer Architecture, 2<sup>nd</sup> Edition, Morgan Kaufmann.

All lectures are on-line available.