



UNIVERSITÀ DEGLI STUDI DI PALERMO

DEPARTMENT	Ingegneria
ACADEMIC YEAR	2023/2024
BACHELOR'S DEGREE (BSC)	COMP'UTER ENGINEERING
SUBJECT	COMPUTER ARCHITECTURES
TYPE OF EDUCATIONAL ACTIVITY	A
AMBIT	50283-Matematica, informatica e statistica
CODE	01477
SCIENTIFIC SECTOR(S)	ING-INF/05
HEAD PROFESSOR(S)	PERI DANIELE Ricercatore Univ. di PALERMO
OTHER PROFESSOR(S)	
CREDITS	9
INDIVIDUAL STUDY (Hrs)	144
COURSE ACTIVITY (Hrs)	81
PROPAEDEUTICAL SUBJECTS	
MUTUALIZATION	
YEAR	1
TERM (SEMESTER)	1° semester
ATTENDANCE	Not mandatory
EVALUATION	Out of 30
TEACHER OFFICE HOURS	PERI DANIELE Wednesday 15:00 16:00 Ricevimento in modalita a distanza sulla piattaforma MS Teams

PREREQUISITES	Nessuno
LEARNING OUTCOMES	Conoscenza e capacita di comprensione Lo studente acquisira' conoscenza sia teorica che di tipo progettuale sulle architetture dei calcolatori e dei loro fondamenti nella logica digitale combinatoria e sequenziale. Conoscera' gli elementi di rappresentazione delle informazioni nei calcolatori e le metodologie di base per la progettazione e l'analisi di architetture di calcolatori a partire dalle componenti logiche combinatorie e sequenziali e i concetti di base della programmazione in linguaggio macchina. Capacita di applicare conoscenza e comprensione Lo studente sara' in grado di affrontare semplici problemi di rappresentazione binaria delle informazioni, di progettare a livello funzionale circuiti logici per la soluzione di semplici problemi e sara' in grado di scrivere semplici programmi nel linguaggio di assemblatore dell'architettura ARM e di codificare e decodificare sequenze di istruzioni in linguaggio macchina. Autonomia di giudizio Lo studente sara' in grado di affrontare in autonomia l'analisi, la progettazione e l'implementazione di componenti architetture per calcolatori in forma di moduli digitali e la soluzione di semplici problemi di programmazione in linguaggio macchina. Sara' in grado di valutare la qualita' delle sue soluzioni in termini di semplicita', leggibilita', strutturazione, efficienza e riutilizzabilita'. Abilita comunicative Lo studente sara' in grado di esporre, efficacemente e con proprieta' di linguaggio, analisi e soluzioni di problemi affrontabili con la progettazione funzionale di componenti architetture e la programmazione in linguaggio macchina, nonche' di problemi di rappresentazione delle informazioni. Capacita d'apprendimento Lo studente sara' in grado di affrontare in maniera autonoma problemi di progettazione di componenti architetture per calcolatori individuando e integrando soluzioni parziali gia' disponibili, sia formalizzate sia implementate. Sara' in grado di analizzare e approfondire in autonomia i paradigmi e i meccanismi di elaborazione dei calcolatori a livello architetture e microarchitettura.
ASSESSMENT METHODS	Le conoscenze e le competenze acquisite dallo studente saranno verificate attraverso una prova finale scritta. La prova scritta ha lo scopo di verificare le conoscenze dello studente riguardo agli argomenti affrontati durante il corso e la sua capacita' di applicare le conoscenze acquisite anche alla risoluzione di piccoli problemi progettuali. La valutazione della prova terra' conto dell'autonomia di giudizio dello studente nel proporre soluzioni adeguate ai problemi proposti e alla capacita' di comunicarle efficacemente. Una prova in itinere facoltativa verrà proposta a metà del corso. La prova in itinere, se superata, consentirà di non dover sostenere la prima parte della prova finale, strutturata identicamente. La formulazione della prova fornisce una valutazione in trentesimi dei risultati attesi in relazione al voto finale come segue: - da 18/30 a 20/30: sufficiente conoscenza e capacita' di comprensione degli argomenti trattati, capacita' di applicazione delle conoscenze acquisite per la risoluzione dei problemi proposti, autonomia di giudizio, abilita' comunicative e capacita' di apprendere; - da 21/30 a 23/30: discreta conoscenza e capacita' di comprensione degli argomenti trattati, capacita' di applicazione delle conoscenze acquisite per la risoluzione dei problemi proposti, autonomia di giudizio, abilita' comunicative e capacita' di apprendere; - da 24/30 a 26/30: buona conoscenza e capacita' di comprensione degli argomenti trattati, capacita' di applicazione delle conoscenze acquisite per la risoluzione dei problemi proposti, autonomia di giudizio, abilita' comunicative e capacita' di apprendere; - da 27/30 a 30/30 e lode: eccellente conoscenza e capacita' di comprensione degli argomenti trattati, capacita' di applicazione delle conoscenze acquisite per la risoluzione dei problemi proposti, autonomia di giudizio, abilita' comunicative e capacita' di apprendere.
EDUCATIONAL OBJECTIVES	The course aims at providing students with fundamentals of digital computer architecture, their functioning at the architectural and microarchitectural levels, low-level programming in machine and assembly language, based on

	fundamental concepts of digital logic, information representation and processing. Students are expected to acquire basic knowledge of digital computer architecture design and be able to code simple ARM machine language and assembly programs.
TEACHING METHODS	Lectures and computer laboratories. E-learning.
SUGGESTED BIBLIOGRAPHY	S. L. Harris, D. M. Harris, "Digital Design and Computer Architecture: ARM Edition", Morgan Kaufmann (Paperback ISBN: 9780128000564, eBook ISBN: 9780128009116) (edizione originale) o S. L. Harris, D. M. Harris, "Sistemi digitali e architettura dei calcolatori. Progettare con tecnologia ARM", Zanichelli (ISBN 978-88-08-42096-1) (edizione italiana)

SYLLABUS

Hrs	Frontal teaching
1	Computers and information processing. Layered organization of computing systems: application software, operating system, architecture, microarchitecture, logic circuits, gates, analog circuits, electronic devices. The digital discipline. Hierarchy, Modularity and Regularity. The Digital Abstraction: Physical continuous values and discrete subsets of values. Signal levels, and binary and logic values. Brief history of digital computers.
3	Number Systems. Decimal numbers. Binary numbers. Information representation with finite sets of bits. Number systems. Binary system. Binary to decimal conversion. Hexadecimal numeral system. Bit, byte, and their multiples. Binary addition. Overflow. Codes.
5	Signed integer representation. Base complement representation. Real number representation: fixed point and floating point. Alphanumeric representation. ASCII code. Image representation.
2	Two-valued logic gates (AND, OR, NOT, XOR. NAND, NOR). Inverting and non-inverting buffers. Truth tables. Signal levels and noise margins.
1	Structure of logic circuits: inputs, outputs, functional and temporal specifications. Circuit modules: nodes and circuit elements. Types of Logic Circuits. Rules of Combinational Composition. Functional specification with boolean equation. Boolean functions. Definitions: variable, complement, literal, minterm, maxterm. Sum-of-Products (SOP) Form. Product-of-Sums (POS) Form.
3	Boolean Algebra. Axioms and theorems. Duality. Boolean axioms. Boolean Theorems of One Variable. Identity Theorem. Null Element Theorem. Idempotency Theorem. Involution Theorem. Complement Theorem. Recap: Basic Boolean Theorems. Boolean Theorems of Several Vars. Associativity. Distributivity. Covering. Combining. Consensus. Boolean Theorems of Several Vars: dual forms. Simplifying an Equation. Simplification methods. DeMorgan's Theorem. Implicants, prime implicants and essential prime implicants of a boolean function.
1	Combinational logic design. Schematics. Guidelines. Wires. Connections. Crossings. Buses. Two-level circuits. Illegal value. Floating value. Tristate buses.
2	Combinational circuits. Multiplexers. Logic synthesis using multiplexers. Decoders. Logic synthesis using decoders. Timing. Propagation and contamination delay. Critical and shortest path. Glitches.
3	Sequential circuits. Latches. Flip-flops. Finite State Machines. Mealy machines and Moore machines.
4	Synthesis of synchronous sequential circuits. Factored FSM. Timing. Delay. Input Timing Constraints. Output Timing Constraints. Timing Analysis.
5	Hardware description languages. SystemVerilog. Modules. Simulation and synthesis. Behavioral models of combinational logic. Bitwise Operators. Assignment. Reduction Operators. Conditional Assignment. Operator precedence. Numbers and constants. Z and X values. Delays. Structural modeling.
3	Sequential logic modeling with SystemVerilog. Idioms. Behavioral instructions. Blocking and non-blocking assignments. Assignment rules. HDL descriptions of finite state machines.
2	Modeling and verification with SystemVerilog. Data types. Parameterized modules. Testbenches.
1	Programmable digital computers. Structure of a digital computer. Von Neuman model: CPU, memory, input/output.

SYLLABUS

Hrs	Frontal teaching
4	Digital building blocks. Arithmetic circuits. 1-bit adders. 1-bit adders. Half adder. Full adder. Multibit adders. Carry propagate adders (CPA). Ripple-carry adder. Ripple-carry adder delay. Carry-lookahead adder. 32-bit CLA with 4-bit Blocks. Carry-Lookahead Addition. Prefix Adder. Adder Delay Comparisons. Subtractor. Equality comparator. Less than comparator. ALU: Arithmetic Logic Unit. Shifters: arithmetic shifter, logical shifter, rotator.
1	Digital building blocks. Sequential building blocks. Counters. Shift registers.
2	Memory arrays. Dynamic Random Access Memory. Static Random Access Memory. Register files. Read Only Memory. Logic arrays.
10	Machine language and assembly language (ARM). Instructions. Operands. Data-processing instructions. Condition flags. Branching. Conditional statements. Loops. Load and store instructions. Function calls. Stack management.
2	Compiling, assembling, loading, and running programs.
4	Microarchitecture. Single-cycle processors. Single-cycle datapath.
4	Microarchitecture. Multicycle processors. Multicycle data path. Pipelined processors.
1	Memory systems. Cache memory. Virtual memory.
1	Input/Output (I/O) systems. Memory-mapped I/O. PC I/O systems. USB. PCI and PCI Express. Memory bus.
Hrs	Practice
2	Information representation. Boolean algebra.
6	Design and test of logic circuits with SystemVerilog.
2	Digital building blocks and microarchitectures.
6	Machine language and assembly language.