



# UNIVERSITÀ DEGLI STUDI DI PALERMO

<b>DEPARTMENT</b>	Ingegneria
<b>ACADEMIC YEAR</b>	2018/2019
<b>MASTER'S DEGREE (MSC)</b>	COMPUTER ENGINEERING
<b>SUBJECT</b>	EMBEDDED SYSTEMS
<b>TYPE OF EDUCATIONAL ACTIVITY</b>	B
<b>AMBIT</b>	50369-Ingegneria informatica
<b>CODE</b>	17873
<b>SCIENTIFIC SECTOR(S)</b>	ING-INF/05
<b>HEAD PROFESSOR(S)</b>	PERI DANIELE                      Ricercatore                      Univ. di PALERMO
<b>OTHER PROFESSOR(S)</b>	
<b>CREDITS</b>	6
<b>INDIVIDUAL STUDY (Hrs)</b>	96
<b>COURSE ACTIVITY (Hrs)</b>	54
<b>PROPAEDEUTICAL SUBJECTS</b>	
<b>MUTUALIZATION</b>	
<b>YEAR</b>	1
<b>TERM (SEMESTER)</b>	1° semester
<b>ATTENDANCE</b>	Not mandatory
<b>EVALUATION</b>	Out of 30
<b>TEACHER OFFICE HOURS</b>	<b>PERI DANIELE</b> Wednesday 15:00 - 16:00    Ricevimento in modalita a distanza sulla piattaforma MS Teams

<p><b>PREREQUISITES</b></p>	<p>Fundamentals of programming, logic circuits, and computer architectures. Comprehensive knowledge of the C language.</p>
<p><b>LEARNING OUTCOMES</b></p>	<p>Conoscenza e capacita' di comprensione Lo studente conoscerà l'evoluzione storica e lo stato dell'arte dei sistemi di elaborazione embedded dal punto di vista teorico e strutturale. Conoscerà quindi i principi di funzionamento delle principali architetture di tali sistemi e dei loro componenti. Avrà conoscenza approfondita della programmazione a basso livello e dei relativi strumenti di sviluppo.</p> <p>Capacita' di applicare conoscenza e comprensione Lo studente sarà in grado di applicare le conoscenze acquisite prospettando possibili soluzioni a problemi di programmazione embedded, e saprà affrontarne l'implementazione.</p> <p>Autonomia di giudizio Lo studente sarà in grado di affrontare in autonomia l'analisi, la progettazione e l'implementazione di soluzioni per problemi di programmazione embedded e sarà in grado di valutarne la qualità in termini di semplicità, versatilità ed efficienza.</p> <p>Abilità comunicative Lo studente sarà in grado di esporre, efficacemente e con proprietà di linguaggio, i principi strutturali e di funzionamento dei calcolatori embedded. Saprà inoltre presentare analisi e soluzioni di problemi affrontabili con la programmazione embedded.</p> <p>Capacità d'apprendimento Lo studente sarà in grado di affrontare in maniera autonoma lo studio dei calcolatori embedded, con particolare riferimento a quanto disponibile in commercio e nella letteratura scientifica. Saprà affrontare problemi di programmazione embedded, individuando e integrando soluzioni parziali già disponibili, sia formalizzate sia implementate. Sarà in grado di approfondire in autonomia la conoscenza di strumenti di progettazione e di sviluppo. Sarà inoltre in grado di integrare le conoscenze acquisite negli altri insegnamenti con quelle del corso.</p>
<p><b>ASSESSMENT METHODS</b></p>	<p>La valutazione dell'apprendimento si basa sulla valutazione dei risultati attesi in accordo con i descrittori di Dublino. Il voto finale viene espresso in trentesimi e varia da 18/30 a 30/30 con lode.</p> <ul style="list-style-type: none"> <li>- Valutazione della conoscenza e capacità di comprensione Per la verifica di questo obiettivo l'esame comprende una discussione orale sugli argomenti del programma. L'obiettivo contribuisce per il 25% al voto finale.</li> <li>- Valutazione della capacità di applicare conoscenza e comprensione Per la verifica di questo obiettivo l'esame comprende lo sviluppo completo di una applicazione embedded, corredata da documentazione testuale che ne descriva scopo, struttura, funzionamento e modalità di utilizzo, a partire da specifiche testuali fornite dal docente. L'applicazione viene sviluppata dallo studente, singolarmente o in gruppo, al termine del corso e deve essere presentata funzionante al momento della discussione orale. L'obiettivo contribuisce per il 25% al voto finale.</li> <li>- Valutazione dell'autonomia di giudizio Nello sviluppo dell'applicazione embedded allo studente si richiede di effettuare delle scelte progettuali in autonomia. L'obiettivo viene quindi valutato discutendo nel dettaglio, durante l'esame orale, le scelte progettuali e implementative compiute nello sviluppo dell'applicazione assegnata. L'obiettivo contribuisce per il 20% al voto finale.</li> <li>- Valutazione delle abilità comunicative Per la verifica di questo obiettivo l'esame comprende la valutazione della documentazione dell'applicazione e la discussione dell'applicazione durante la prova orale. L'obiettivo contribuisce per il 10% al voto finale.</li> <li>- Valutazione della capacità di apprendere Per la verifica di questo obiettivo l'esame comprende la discussione di alcuni argomenti avanzati introdotti a lezione e il cui approfondimento è lasciato agli studenti per la realizzazione dell'applicazione embedded. In particolare, l'obiettivo viene valutato discutendo le tecniche apprese autonomamente dallo studente e impiegate nella realizzazione dell'applicazione. L'obiettivo contribuisce per il 20% al voto finale.</li> </ul> <p>La formulazione delle prove fornisce una valutazione dei risultati attesi in relazione al voto finale come segue:</p> <ul style="list-style-type: none"> <li>- da 18/30 a 20/30: sufficiente conoscenza e capacità di comprensione degli argomenti trattati, capacità di applicazione delle conoscenze acquisite per la risoluzione dei problemi proposti, autonomia di giudizio, abilità comunicative e capacità di apprendere.</li> </ul>

	<p>- 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.</p> <p>- 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.</p> <p>- 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.</p>
<b>EDUCATIONAL OBJECTIVES</b>	The course aims at providing students with advanced knowledge of embedded computing systems. Students are expected to acquire advanced skills in low-level and system programming.
<b>TEACHING METHODS</b>	Lectures and computer laboratories.
<b>SUGGESTED BIBLIOGRAPHY</b>	<ul style="list-style-type: none"> <li>- E. A. Lee and S. A. Seshia, Introduction to Embedded Systems - A Cyber-Physical Systems Approach, Second Edition, MIT Press, ISBN 978-0-262-53381-2, 2017, disponibile on-line (LeeSeshia.org)</li> <li>- L. Brodie, "Starting Forth", disponibile on-line.</li> <li>- J. Bartlett, Programming from the ground up, disponibile on-line.</li> <li>- A. S. Tanenbaum, T. Austin, "Structured computer organization. 6th ed.", Pearson</li> <li>- D. A. Patterson, J. L. Hennessy, "Computer Organization and Design", Morgan Kaufmann</li> <li>- L. Brodie, "Thinking Forth", disponibile on-line.</li> </ul>

### SYLLABUS

Hrs	Frontal teaching
6	Architectures of Embedded Systems. Sensors and actuators. Embedded processors. Memory Architectures. Input and Output. Multitasking. Scheduling. bus.
4	Input and Output. General-purpose Input/Output. Parallel buses. Serial Interconnections. UART and USART. Serial Peripheral Interconnect (SPI). I2C.
6	The STM32-F446 architecture. Memory organization. Clocks. GPIO and alternate functions. I2C controller. Hardware timers. General-purpose timers. Interrupts and Interrupt controller. Vector table and Interrupt Service Routines (ISR).
4	Forth. Interactive environment and the interpreter. The stack. Definition and compilation of Words. Stack notation. The Dictionary. Interpretation and Execution. Arithmetic Operations with the Stack. Stack manipulations. Input and Output.
4	Forth. Selection constructs. Loops. Numeric types. Variable, constants and arrays.
4	Forth. Interpreter and Dictionary Implementation. Execution Token. Vectorial Execution. External and Internal Interpreters. System Variables. PAD. Text Input Buffer. User Dictionary.
4	Forth. Return Stack. User Variables. Vocabularies.
4	Hardware Abstraction Layers and driver programming.
4	Assembly. Integer and floating point arithmetic. Vector instructions. Multimedia extensions.
Hrs	Practice
6	Forth Programming.
4	Hardware Abstraction Layers and driver programming.
4	Assembly. Integer and floating point arithmetic. Vector instructions. Multimedia extensions.