



# UNICUSANO

Università degli Studi Nicolò Cusano - Telematica Roma

<b>Insegnamento</b>	Sistemi embedded
<b>Livello e corso di studio</b>	Laurea Magistrale in Ingegneria Informatica
<b>Settore scientifico disciplinare (SSD)</b>	ING-INF/05
<b>Anno di corso</b>	4
<b>Anno Accademico</b>	2022-2023
<b>Numero totale di crediti</b>	12
<b>Propedeuticità</b>	Nessuna
<b>Docente</b>	Armando Piccardi Facoltà: Ingegneria Nickname: armando.piccardi Email: armando.piccardi@unicusano.it Orario di ricevimento: gli studenti possono contattare il docente per concordare un ricevimento
<b>Presentazione</b>	Il corso di Elettronica dei Sistemi Programmabili ha lo scopo di fornire le nozioni fondamentali sui sistemi a microprocessore e microcontrollore, nonché le basi per implementare semplici dispositivi attraverso la programmazione di schede di sviluppo. A partire dai dispositivi per l'elettronica digitale, verrà studiata l'architettura dei microprocessori, con particolare riferimento ai processori ARM, le loro componenti fondamentali e i problemi da affrontare in fase di progettazione. Successivamente verranno introdotti i sistemi a microcontrollore, studiandone l'architettura generale e dove necessario facendo riferimento a specifici modelli (NXP, Arduino). Inoltre, verranno forniti gli strumenti per la programmazione di sistemi a microcontrollore, con riferimento sia al linguaggio di programmazione (C/C++) che all'ambiente di sviluppo. Infine, anche attraverso le Etivity, verranno proposti esempi di implementazione di dispositivi attraverso l'utilizzo di microcontrollori.
<b>Obiettivi formativi</b>	Il corso di Sistemi Embedded ha i seguenti obiettivi formativi: <ol style="list-style-type: none"> <li>1. Comprendere l'architettura e il funzionamento dei sistemi a microprocessore e microcontrollore.</li> <li>2. Comprendere gli strumenti per l'analisi e/o il progetto di architetture di sistemi programmabili.</li> <li>3. Acquisire le competenze necessarie per progettare sistemi di controllo, sia per quanto riguarda l'hardware che il firmware/software.</li> <li>4. Acquisire gli strumenti per implementare semplici sistemi di controllo su piattaforme per microcontrollori.</li> </ol>
<b>Prerequisiti</b>	La frequenza al corso non richiede il superamento di alcuna propedeuticità. Tuttavia, è consigliata la <b>conoscenza</b> di base di dispositivi e sistemi elettronici. In particolare, pur fornendo il corso una parte di revisione dei concetti necessari, si consiglia di rivedere le principali nozioni riguardanti i dispositivi digitali. È inoltre consigliato avere le conoscenze di base di un linguaggio di programmazione (preferibilmente C/C++).
<b>Risultati di apprendimento attesi</b>	Le competenze fornite dal corso allo studente sono declinate secondo i descrittori di Dublino: <ul style="list-style-type: none"> <li>• Conoscenza e comprensione</li> <li>• Capacità di applicare conoscenza e comprensione</li> <li>• Autonomia di giudizio</li> <li>• Abilità comunicative</li> <li>• Capacità di apprendimento</li> </ul> <p><b>Conoscenza e capacità di comprensione</b> Al termine del Corso lo studente dimostrerà di conoscere le principali caratteristiche dei sistemi a microprocessore e microcontrollore e le problematiche da affrontare legate alla loro progettazione. Inoltre, acquisirà la conoscenza degli strumenti per la programmazione di sistemi a microcontrollore.</p> <p><b>Applicazione delle conoscenze</b> Lo studente sarà in grado di utilizzare le conoscenze acquisite per l'analisi di sistemi <i>embedded</i>. Avrà inoltre gli</p>

	<p>strumenti per poter programmare sistemi a microcontrollore per l'implementazione dispositivi elettronici di controllo.</p> <p><b>Capacità di trarre conclusioni</b> Lo studente sarà in grado di identificare le caratteristiche in termini di prestazioni di sistemi a microprocessore e microcontrollore. Dimostrerà inoltre di saper analizzare problemi legati alla progettazione di sistemi <i>embedded</i> e individuare le scelte hardware e software più appropriate all'applicazione.</p> <p><b>Abilità comunicative</b> Lo studente acquisirà un adeguato linguaggio tecnico scientifico che gli permetterà di descrivere e sostenere conversazioni su architettura e funzionamento di sistemi programmabili di controllo.</p> <p><b>Capacità di apprendere</b> Al termine del Corso, attraverso le conoscenze e competenze necessarie per l'analisi di sistemi a logica programmabile, lo studente avrà acquisito maggiore consapevolezza nell'affrontare problemi di analisi e progettazione dei sistemi <i>embedded</i>. Questo contribuirà alla sua crescita formativa nell'ambito ingegneristico per poter affrontare problemi posti sia in ambiente accademico che lavorativo.</p>
<p><b>Organizzazione dell'insegnamento</b></p>	<p>Il corso è sviluppato attraverso le <b>lezioni preregistrate audio-video</b> che compongono i materiali di studio disponibili in piattaforma insieme alle <b>slide/dispense</b> fornite dal docente.</p> <p>Sono proposti, per ogni modulo, dei <b>test di autovalutazione</b> di tipo asincrono, aventi la funzione di permettere agli studenti di accertarsi del grado di conoscenza acquisita dei contenuti del corso.</p> <p>La <b>didattica interattiva</b> comprende 3 <b>Etivity</b> che hanno l'obiettivo di applicare le conoscenze acquisite nelle lezioni più un'ulteriore <b>Etivity</b> riguardante il progetto di un semplice dispositivo da implementare con un sistema a microcontrollore.</p> <p>Il Corso di Sistemi Embedded prevede 12 Crediti formativi. Il carico totale di studio per questo modulo di insegnamento è di circa 320 ore, così suddivise:  <b>circa 140</b> ore per la visualizzazione e lo studio del materiale videoregistrato (16.5 ore di teoria e 3.5 ore di esercitazioni videoregistrate).  <b>Circa 13 ore di Didattica Interattiva</b> per l'elaborazione e la consegna di 2 Etivity  <b>Circa 7 ore di Didattica Interattiva</b> per l'esecuzione dei test di autovalutazione.  Si consiglia di distribuire lo studio della materia uniformemente in un periodo di 12 settimane dedicando tra le 15 alle 20 ore di studio a settimana.</p>
<p><b>Contenuti del corso</b></p>	<p>Il corso è diviso in 10 moduli. I primi due saranno dedicati a una revisione dei dispositivi dell'elettronica digitale e utili per la comprensione dei sistemi a microprocessore e microcontrollore. Nel terzo e quarto modulo si studierà l'architettura dei microprocessori. Nel quinto modulo si affronterà lo studio delle basi del linguaggio di programmazione Assembly. Il sesto modulo sarà dedicato all'architettura dei sistemi a microprocessore. Nel settimo modulo si tratteranno i principi fondamentali sul funzionamento di sensori e gli attuatori, dal punto di vista del loro utilizzo nei sistemi <i>embedded</i>. Nell'ottavo modulo si richiameranno le basi del linguaggio di programmazione C. Il nono modulo sarà dedicato ai principi per la programmazione dei sistemi a microcontrollore. L'ultimo modulo è dedicato al progetto di un sistema automatico di controllo, da implementarsi attraverso la programmazione di un microcontrollore.</p> <p><b>Modulo 1 – Elettronica digitale combinatoria</b> (2 ore di lezioni di teoria videoregistrate corredate da dispense per approfondimenti per un impegno di 21 ore - Settimana 1)  Richiami di elettronica digitale: dai transistor alle porte logiche. Principi di Algebra di Boole. Circuiti a logica combinatoria: porte logiche, implementazioni di generiche funzioni logiche. Dispositivi per l'elettronica digitale: multiplexer, demultiplexer, codificatori, decodificatori; dispositivi programmabili, ROM. Principali metodi di sintesi di circuiti digitali combinatori.</p> <p><b>Modulo 2 – Elettronica digitale sequenziale</b> (2.5 ore di lezioni di teoria videoregistrate corredate da dispense per approfondimenti per un impegno di 24.5 ore - Settimane 2)  Dispositivi digitali sequenziali: latch, flip-flop; registri, contatori; memoria RAM, architettura e indirizzamento. Metodi di sintesi di circuiti digitali sequenziali: la macchina a stati finiti, esempi.</p> <p><b>Modulo 3 - Architettura di Microprocessori</b> (3.5 ore di lezioni di teoria videoregistrate per un impegno di 24.5 ore - Settimana 3)  Microprocessori, cenni storici e caratteristiche generali. Classificazione: architettura di Von Newman e Harvard, architetture RISC e CISC. Componenti, blocchi funzionali, architettura. Microprocessore minimo Mu0: set di e formato delle istruzioni; datapath e unità di controllo; ALU; possibili estensioni. Introduzione all'architettura di microprocessori: formato delle istruzioni, metodi di indirizzamento, pipeline. Approfondimenti sull'architettura RISC.</p> <p><b>Modulo 4 – Il microprocessore ARM</b> (3 ore di lezioni di teoria videoregistrate per un impegno di 21 ore - Settimane 4-5)  Introduzione e generalità sui processori ARM. Architettura, blocchi funzionali. Set di istruzioni, tipi e formato delle istruzioni e loro esecuzione. Banco registri, gestione della memoria: gerarchia di memoria, modalità di indirizzamento. Gestione degli interrupt. Pipeline, datapath. Architettura e set di istruzioni Thumb. Generalità</p>

	<p>sulle architetture per linguaggi ad alto livello. Hardware: ALU, unità di controllo, Barrel shifter.</p> <p><b>Modulo 5 – Linguaggio Assembly</b> (3 ore di lezioni di teoria videoregistrate per un impegno di 21 ore - Settimana 6) Linguaggi a basso e alto livello. Interprete, compilatore, assemblatore. Caratteristiche generali e importanza del linguaggio Assembly. Istruzioni ARM Assembly e legame con l'architettura hardware dei processori ARM. Istruzioni di elaborazione, di trasferimento dai e di salto. Indirizzamento; gestione delle costanti numeriche. Direttive. Esempi.</p> <p><b>Modulo 6 – Architettura di Microcontrollori</b> (3 lezione di teoria videoregistrata e 1 ora di esercitazione per un impegno di 26 ore - Settimana 7-8) Architettura generale di un microcontrollore. Conversione A/D e D/A: esempi di convertitori. Memoria di sistema: memorie ROM, PROM, EPROM, EEPROM. Periferiche e comunicazione: UART, I2C, SPI. Temporizzazione: Timer, PWM, Watchdog.</p> <p><b>Modulo 7 – Sensori e attuatori</b> (3.5 ore di lezioni di teoria videoregistrate per un impegno di 24.5 ore - Settimana 9) Definizione, generalità e classificazione di trasduttori, sensori e attuatori. Principali parametri per la valutazione delle caratteristiche e delle prestazioni di trasduttori. Sensori resistivi e capacitivi: principali caratteristiche e interfacciamento con dispositivi a microcontrollore. Sensori e attuatori di luce: generalità, interfacciamento e controllo attraverso sistemi programmabili. Caso di studio: dispositivo a touchscreen.</p> <p><b>Modulo 8 – Linguaggio di programmazione C</b> (Studio individuale indicazioni del docente, per un totale di 15.5 ore – Settimana 10) Il linguaggio di programmazione C, generalità. Programmazione in C per sistemi embedded. Struttura di un programma. Identificatori: dichiarazione e tipi di variabili. Operatori. Istruzioni condizionali. Funzioni. Strutture dati.</p> <p><b>Modulo 9 – Programmazione di Microcontrollori</b> (3 ore di lezioni di teoria videoregistrate e 1 ora di esercitazione corredate da dispense del docente per un impegno di 26 ore - settimane 11) Ambiente di sviluppo, IDE, software di sviluppo. Strumenti: librerie, debug. Cenni sulla programmazione: progetto di un sistema con macchina a stati finiti, struttura del codice corrispondente. Progetto di un sistema a microcontrollore: interfacciamento, breadboarding, collegamento dispositivi, gestione dei segnali di ingresso e uscita. Il simulatore Mbed, analisi di semplici sistemi di controllo.</p> <p><b>Modulo 10 – Progetto di un sistema embedded</b> (Studio individuale e realizzazione di un progetto, 28 ore - Settimana 12) Il docente assegnerà il progetto di un sistema di controllo. Sviluppo del progetto: scelta delle componenti hardware, reperimento e studio della documentazione tecnica sulla piattaforma assegnata (consigliata dal docente o concordata con lo studente) sia delle caratteristiche hardware che delle librerie per la programmazione, implementazione (realizzazione e programmazione) del sistema <i>embedded</i>, stesura della documentazione da allegare al sistema.</p> <p><b>Etivity 1</b> – Implementazione di dispositivi attraverso metodi di sintesi di circuiti digitali (3 ore di carico di studio – Settimana 2 / Settimana 3).</p> <p><b>Etivity 2</b> – Progetto di un sistema proprio dei sistemi a microprocessore, da affrontare attraverso i metodi appresi nei moduli precedenti (6 ore di carico di studio - Settimana 4).</p> <p><b>Etivity 3</b> – Implementazione di un algoritmo attraverso il linguaggio ARM Assembly (8 ore di carico di studio – Settimana 6).</p> <p><b>Etivity 4</b> – Progetto di un sistema programmabile da implementare su piattaforma a microcontrollore (25 ore di carico di studio – Settimana 12).</p>
<p><b>Materiali di studio</b></p>	<p>· MATERIALI DIDATTICI A CURA DEL DOCENTE</p> <p>Il materiale didattico presente in piattaforma è suddiviso in 6 moduli. Essi ricoprono interamente il programma e ciascuno di essi contiene dispense, slide e videolezioni in cui il docente commenta le slide. Tale materiale contiene tutti gli elementi necessari per affrontare lo studio della materia.</p> <p>Testi consigliati: S. Furber , "ARM, System on Chip Architecture", Addison Wesley.</p>
<p><b>Modalità di verifica dell'apprendimento</b></p>	<p>L'esame consiste nello svolgimento di una <b>prova scritta</b> tendente ad accertare le capacità di analisi e rielaborazione dei concetti acquisiti e di una serie di attività (<b>E-tivity</b>) svolte durante il periodo di studio.</p> <p>La valutazione sarà effettuata attraverso:</p> <ul style="list-style-type: none"> <li>- la prova scritta, che consiste in un certo numero di domande a risposta aperta o chiusa, per cui è assegnato un punteggio totale pari a 30. Il numero delle domande e il punteggio assegnato a ogni domanda dipendono dalla difficoltà delle domande stesse (il punteggio assegnato a ogni domanda verrà</li> </ul>

	<p>indicato sulla prova). Le domande potranno essere teoriche o riguardare l'applicazione dei concetti studiati per risolvere problemi.</p> <ul style="list-style-type: none"> <li>- la correzione delle E-tivity, la cui consegna è da effettuarsi entro il giorno della prova scritta. Il punteggio totale assegnato alle E-tivity sarà specificato all'assegnazione delle stesse e potrà pesare sulla votazione finale, fermo restando il superamento della prova scritta.</li> </ul> <p>N.B. Nella prova scritta, le domande possono coprire l'intero programma, lo svolgimento delle E-tivity non garantisce l'esonero da parte del programma.</p> <p>I risultati di apprendimento attesi circa le conoscenze della materia e la capacità di applicarle sono valutate sia dalla prova scritta che dalla risoluzione delle E-tivity, mentre le abilità comunicative, la capacità di trarre conclusioni e la capacità di autoapprendimento sono valutate in itinere attraverso le E-tivity.</p>
<p><b>Criteri per l'assegnazione dell'elaborato finale</b></p>	<p>L'assegnazione dell'<b>elaborato finale</b> avverrà sulla base di un colloquio con il docente in cui lo studente manifesterà i propri specifici <b>interessi</b> in relazione a qualche argomento che intende approfondire; non esistono <b>preclusioni</b> alla richiesta di assegnazione della tesi e non è prevista una <b>media particolare</b> per poterla richiedere.</p>