

Programma degli argomenti svolti

anno scolastico: **2019/2020**

classe: **3F (ITT – articolazione Telecomunicazioni)**

materia: **Tecnologia e Progettazione di Sistemi Informatici e di Telecomunicazioni**

docenti: **Prof. Lorenzo Donati, Prof.ssa Mara Aloisi**

Nota:** il presente programma non riflette pedissequamente la distribuzione temporale degli argomenti durante l'anno scolastico, bensì è organizzato secondo blocchi tematici omogenei. **In grassetto sono indicati gli argomenti indispensabili per il raggiungimento degli obiettivi didattici minimi; su tali argomenti si baserà la prova di esame per gli alunni con giudizio sospeso.

***Nota:** Dato che il docente di teoria (Lorenzo Donati) insegna nella medesima classe anche la materia "Sistemi e Reti", per agevolare l'apprendimento degli studenti si è deciso di razionalizzare la programmazione dei due insegnamenti. Infatti i contenuti di tali materie presentano numerosissimi punti in comune. Pertanto i medesimi moduli didattici sono stati inseriti nei piani di lavoro di entrambe le materie e sono stati svolti in contemporanea.*

Algebra di Boole

- **Stati logici elementari, bit, parole binarie, variabili logiche.**
- **Operazioni elementari AND, OR e NOT. Espressioni logiche. Espressioni logiche equivalenti ed identità logiche. Determinazione della tabella di verità di un'espressione logica.**
- **Proprietà dell'algebra di Boole: associativa, commutativa, distributiva, doppia negazione, dei complementi, di idempotenza, di neutralità, di annullamento, di DeMorgan, di assorbimento, di assorbimento del complemento.**
- **Operazione logica di OR esclusivo (XOR), sua importanza applicativa e sue proprietà.**
- **Operazioni bit a bit tra parole binarie. Operazioni di shift su parole binarie.**

Reti Logiche

- **Porte logiche: AND, OR, NOT, buffer, NAND, NOR, XOR, XNOR. Circuiti logici. Circuiti logici equivalenti. Porte a più ingressi.**
- **Universalità delle porte NAND e NOR.**
- **Determinazione della legge di commutazione di una rete combinatoria a partire dal suo schema logico e viceversa.**
- **Somma binaria. Sommatore binario (Full Adder).**
- **Concetto di ingressi ed uscite attivi alti o bassi. Concetto di ingresso di abilitazione.**
- **Diagrammi di temporizzazione. Ritardo di propagazione nelle reti logiche.**

- **Azzardi nelle reti combinatorie (cenni) e circuiti rilevatori di fronti del segnale.**
- **Concetto di riconoscimento di uno stato logico. Mintermini. Passaggio dalla tabella di verità alla legge di commutazione: forma canonica SP.**
- **Multiplexer (MUX), demultiplexer (DEMUX) e decoder binari. Applicazioni.**
- **Reti sequenziali sincrone ed asincrone: latch SR, latch D, flip-flop D.**
- **Pilotaggio delle reti sequenziali: tempo di setup e tempo di hold.**
- **Registri di memoria, registri a scorrimento, contatori, divisori di frequenza, prescaler, contatori. Applicazioni.**

Elettronica Digitale

- **Rappresentazione dei livelli logici con livelli di tensione elettrica.**
- **Tecnologie digitali attuali: CMOS ed ECL, differenze principali e loro ambiti applicativi. Cenni alla tecnologia obsoleta TTL e legami con le tecnologie attuali. Famiglie logiche CMOS e compatibilità con le TTL.**
- **Cenni ai transistori MOSFET: funzionamento ON/OFF semplificato. Cenni al transistor FG MOS (Floating Gate MOSFET) e sua applicazione ai circuiti di memoria non-volatile.**
- **Concetto di "stato di Alta Impedenza", reti con uscite 3-state e loro applicazione (bus bidirezionali e multidrop).**
- **Porte di trasmissione CMOS e loro applicazioni (in particolare: realizzazione di MUX/DEMUX analogici e porte 3-state).**

- Struttura e funzionamento di un inverter CMOS. Implementazione delle uscite 3-state mediante struttura CMOS.
- Ingressi con trigger di Schmitt.
- **Le tecnologie delle memorie digitali. Memorie volatili: RAM statiche e dinamiche. Memorie non volatili: mask-ROM, PROM, EPROM, EEPROM, FLASH, NVRAM.**
- **Pilotaggio di un LED mediante un'uscita digitale. Dimensionamento della resistenza di limitazione.**
- **Contatti elettrici normalmente chiusi e normalmente aperti e loro uso come sorgenti di livelli logici. Resistori di pull-up e di pull-down.**

Architettura dei Sistemi di Elaborazione Digitale

- Cenni ai vari dispositivi programmabili usabili come cuore dei sistemi di elaborazione (CPU, MCU, DSP, FPGA), loro peculiarità ed ambiti applicativi.
- **Struttura semplificata di una CPU: parte operativa (ALU, registri generali) e parte controllo. Instruction register (IR) e program counter (PC). Memoria di lavoro e bus dati. Bus indirizzi e di spazi di indirizzamento. Architettura di Von Neumann e architettura di Harvard.**
- CPU ed il linguaggio macchina. Cenni alla struttura del set di istruzioni macchina: codici operativi ed operandi. Ciclo FDE (Fetch-Decode-Execute). Necessità pratica di un linguaggio a più alto livello di astrazione: il linguaggio assembly ed il programma assembler.
- Lo stack come struttura dati. Stack hardware di un microprocessore e registro Stack Pointer (SP). Importanza dello stack hardware. Subroutine.
- CPU con architettura AVR e set di istruzioni della CPU interna della MCU ATmega328P e sua programmazione in assembly.
- **Esame dettagliato del processo di traduzione del codice sorgente C/C++ a codice eseguibile: preprocessore C, compilatore vero e proprio, linker, loader. Concetto di macchina/architettura HOST e macchina/architettura TARGET. Cross-compilazione.**
- **Il meccanismo delle interruzioni. Routine di gestione delle interruzioni (ISR – Interrupt Service Routine) e vettore delle interruzioni (Istruzione assembly IRET). Abilitazione/disabilitazione globale delle interruzioni (istruzioni assembly CLI e SEI). Mascheramento delle interruzioni.**

Microcontrollori

- Programmazione di sistemi embedded: problematiche e particolarità. Sistemi in tempo reale soft e hard. Sistemi embedded basati su microcontrollore (MCU).
- Caratteristiche principali della MCU ATmega328P. Schema a blocchi e funzionalità di base dei singoli blocchi.
- **Struttura e funzionamento delle porte GPIO (General Purpose Input/Output). Configurazione e utilizzo delle porte GPIO come ingressi e come uscite. Controllo delle porte GPIO tramite programmazione dei registri in linguaggio C/C++. Lettura dello stato di contatti elettromeccanici mediante ingressi GPIO. Resistenze di pull-up e pull-down.**
- Sistema di gestione dell'alimentazione e del reset.
- **Sistema Watchdog, sue caratteristiche e suo utilizzo.**
- Comparatore analogico e riferimento di tensione "bandgap".
- **Interruzioni nel microcontrollore ATmega328P.**
- Programmazione del firmware nei sistemi a microcontrollore. Bootloader ed uso dell'interfaccia UART per la programmazione "in-circuit".
- **Uso dell'ambiente Arduino per la programmazione della scheda Arduino UNO.**
- **Preprocessore C: macrooggetti (object-like macros) e macrofunzioni (function-like macros); direttive del preprocessore. Uso delle macro nella programmazione embedded.**
- **Linguaggio C: operazioni bit a bit e operazioni di shift.**
- Suddivisione di un progetto software in unità logicamente coerenti: la programmazione modulare (concetti, vantaggi). Concetto astratto di modulo: interfaccia verso il cliente/utente e implementazione; il contratto tra cliente e interfaccia del modulo. Organizzazione di un programma C/C++ in moduli: modello realizzativo mediante singolo file header (interfaccia) per singolo file sorgente

Sistemi Operativi

- **Il sistema operativo come gestore unico delle risorse della macchina. Esecuzione del software in modalità sistema e modalità utente e relativo supporto hardware (modalità privilegiata e non privilegiata della CPU; istruzioni privilegiate).**