COMMUNICATION ENGINEERING AND ELECTRONIC TECHNOLOGIES

(Lecce - Università degli Studi)

Teaching LABORATORY OF ELECTRONIC DESIGN AND PROTOTYPING

GenCod A004575

Owner professor Paolo VISCONTI

Teaching in italian LABORATORY OF ELECTRONIC DESIGN AND

Teaching LABORATORY OF ELECTRONIC DESIGN AND PROTOTYPING

SSD code ING-INF/01

Reference course COMMUNICATION ENGINEERING AND ELECTRONIC

Course type Laurea Magistrale

Credits 6.0

Teaching hours Front activity hours: 54.0

For enrolled in 2017/2018

Taught in 2018/2019

Course year 2

Language ENGLISH

Curriculum PERCORSO COMUNE

Location Lecce

Semester Second Semester

Exam type Oral

Assessment Final grade

Course timetable https://easyroom.unisalento.it/Orario

BRIEF COURSE DESCRIPTION

Overview

The course aims to provide an overview on the hardware and software platform related to Arduino prototyping board. By initially addressing the basic concepts related to Arduino board and IDE programming, the course introduces the Arduino hardware, its interfacing with sensors, components and accessories in order to give an overview on rapid prototyping solutions for Arduino-based electronic design. Furthermore, the principles of ATMEGA micro-controllers programming are addressed with the aim to read correctly signals provided by the interfaced sensors and to drive load such as motors / actuators. Finally, the realization and testing of realized prototypes during course are performed. In addition a comparison of features and performance between Arduino prototyping board and the Raspberry PI 3 platform, is carried out, in order to be able to choose the most suitable architecture for a specific application, as function of the development board / microcontroller characteristics.

Course Contents

Introduction: Arduino platform - circuital scheme, embedded microcontroller, board pinout - Analog and Digital pins.
Arduino Integrated Development Environment and firmware structure.
Arduino board interfacing with sensors, transducers, actuators, processing devices and smart units with related firmware implementation.
Prototypes realization and testing on proto-boards.

REQUIREMENTS

Knowledge in analog and digital electronic, C/C++ programming.

COURSE AIMS	**Learning Outcomes;** after the course the student should be able to: * Understand the programming principles of Arduino prototyping platform (ATMEGA microcontroller). * Define technical specifications and to select sensors and transducers for a given application. * Interface sensors and actuators with the Arduino prototyping board with the related reading and driving of the different interfaced devices. * Realize and test Arduino-based circuital protpotypes in order to verify the correct operation of the implemented electronic solutions. * Understand and to resolve the eventual problems occurring in the prototype realization and testing phase. * Choose the suitable development board, between Arduino and STM32 Nucleo, for a specific application.
TEACHING METHODOLOGY	The course consists of lectures by using the slides provided to the students and laboratory activities related to the design and testing of smart arduino-based electronic solutions. Also the teacher makes available on the website in addition to the slides of the lessons, further handouts to facilitate the understanding of the topics and the designing of the electronics systems. The final exam consists of an oral question on the theoretical topics of the course and in the presentation of a project realized by the student based on the Arduino / Raspberry platform
ASSESSMENT TYPE	**Examination:** oral discussion related to course topics, Arduino boards architecture and related to Arduino-based project realized by the student. The exam consists of an oral examination related the theoretical and practical contents of the course. In addition, the student has to present a circuital project realized with Arduino platform and discuss its contents showing operation modes of designed electronic board managed by Arduino microcontroller and related simulation results (maximum overall duration: two hours).
OTHER USEFUL INFORMATION	**Office Hours:** By appointment; contact the instructor by email or at the end of class meetings.

FULL SYLLABUS

Overview

The course aims to provide an overview on the hardware and software platform related to Arduino prototyping board. By initially addressing the basic concepts related to Arduino board and IDE programming, the course introduces the Arduino hardware, its interfacing with sensors, components and accessories in order to give an overview on rapid prototyping solutions for Arduino-based electronic design. Furthermore, the principles of ATMEGA micro-controllers programming are addressed with the aim to read correctly signals provided by the interfaced sensors and to drive load such as motors / actuators. Finally, the realization and testing of realized prototypes during course are performed. In addition a comparison of features and performance between Arduino prototyping board and the Raspberry PI 3 platform, is carried out, in order to be able to choose the most suitable architecture for a specific application, as function of the development board / microcontroller characteristics.

Course Contents

Introduction: Arduino platform - circuital scheme, embedded microcontroller, board pinout - Analog and Digital pins.
Arduino Integrated Development Environment and firmware structure.
Arduino board interfacing with sensors, transducers, actuators, processing devices and smart units with related firmware implementation.
Prototypes realization and testing on proto-boards.

PROGRAMMA DETTAGLIATO DEL CORSO DI LABORATORY OF ELECTRONIC DESIGN AND PROTOTYPING

- **LEZIONE 1**: cenni storici su Arduino: evoluzione temporale. Introduzione all'architettura hardware della scheda Arduino UNO R3. Funzionalità ed ambiti applicativi.
- **LEZIONE 2**: Architettura hardware dettagliata e relativi registri di settaggio dei pin e del modulo di conversione AD della scheda Arduino UNO.
- **LEZIONE 3**: Introduzione all'ambiente di sviluppo integrato (IDE) di Arduino e relativi settaggi. Introduzione al linguaggio di programmazione ed esempi applicativi. Procedure per il debugging del codice realizzato in funzione della specifica applicazione.
- **LEZIONE 4**: Introduzione agli apparati e dispositivi impiegati per l'esecuzione delle esercitazioni di laboratorio: dispositivi e schede per il sensing e l'acquisizione dei parametri.
- **LEZIONE 5**: Fondamenti del linguaggio di programmazione di Arduino: installazione delle librerie, analisi della sintassi delle differenti funzioni, esempi di codice.
- **ESERCITAZIONE 1**: Esempi applicativi relativi all'utilizzo dei GPIO di Arduino settati come uscita per il pilotaggio di LED o altri dispositivi.
- **ESERCITAZIONE 2**: Esempi applicativi relativi all'utilizzo dei GPIO di Arduino settati come Input per l'acquisizione del livello logico fornito da dispositivi esterni (pulsanti o sensori digitali ON/OFF).
- **LEZIONE 6**: Descrizione e analisi dei protocolli di comunicazione *1–wire* e I2C (interintegrated circuit). Casi di studio: sensori commerciali con uscita digitale DHT11 (sensore di temperatura/umidità one-wire), DS18B20 (sensore di temperatura one-wire) e LM75a (sensore di temperatura I2C).
- **ESERCITAZIONE 3**: Esempi applicativi relativi all'utilizzo dei pin Analogici di Arduino per l'acquisizione di segnali analogici. Esempi di utilizzo dei pin PWM di Arduino per la generazione di segnali PWM.
- **LEZIONE 7**: Descrizione e analisi dei protocolli di comunicazione SPI *(Serial Preripheral Interface)* e UART *(Universal Asinchronous RiceTrasmission).*
- **ESERCITAZIONE 4**: Realizzazione dell'hardware e implementazione firmware per l'acquisizione della temperatura per mezzo di un termistore, (implementazione equaz. di SteinhartHart nel firmware); verifica del funzionamento mediante specifica strumentazione (oscilloscopio).

- LEZIONE 8: Introduzione all'architettura interna dell'integrato ESP8266; esempi di moduli che integrano il modulo ESP8266 (NodeMCU, WeMOS,….). Architettura della scheda elettronica di acquisizione/processing nodeMCU ver 1.0.
- **ESERCITAZIONE 5**: Esempio di utilizzo dell'interfaccia I2C per lo scambio dei dati tra due schede Arduino; realizzazione dell'hardware e implementazione firmware per l'acquisizione del segnale da un microfono ad elettrete.
- **LEZIONE 9**: Esempio di applicazione tramite scheda nodeMCU per la realizzazione di una WSN per applicazioni in ambito agricolo. Realizzazione nodo sensore e modalità di funzionamento della WSN.
- **LEZIONE 10**: Introduzione alla scheda di prototipazione Raspberry: cenni storici e architettura della versione Pi3 B della scheda.
- **ESERCITAZIONE 6**: Modalità di comunicazione del sensore di temperatura DS18B20; realizzazione dell'hardware e implementazione firmware per l'acquisizione della temperatura per mezzo di un Interfacciamento con il sensore di temperatura Dallas DS18B20. Verifica del funzionamento mediante specifica strumentazione (oscilloscopio).
- **LEZIONE 11**: Fondamenti di programmazione Python su Raspberry Pi, comunicazione tra nodi sensori mediante protocollo MQTT, introduzione alla piattaforma NodeRed per realizzazione di interfaccia grafica per il monitoraggio e controllo remoto di sensori e attuatori.
- **ESERCITAZIONE 7**: Realizzazione dell'hardware e implementazione firmware di un sistema contapezzi con emettitore laser Arduino-compatibile a 650nm e foto-rilevatore (foto-resistenza).
- **ESERCITAZIONE 8**: Realizzazione dell'hardware e implementazione firmware per un sistema di controllo di livello e temperatura (sensore DS18B20) applicato ad un serbatoio; realizzazione dell'hardware e implementazione firmware per un sistema di monitoraggio della temperatura per mezzo del sensore LM75A; verifica della corretta comunicazione mediante specifica strumentazione (oscilloscopio).
- **LEZIONE 12**: Comunicazione tra scheda Raspberry e nodeMCU per la realizzazione di una WSN e visualizzazione dei dati acquisiti su pagina web realizzata per mezzo di una piattaforma NodeRed. Interfacciamento tra la scheda Raspberry Pi e una scheda Arduino UNO per l'acquisizione dati e visualizzazione degli stessi da remoto.
- **ESERCITAZIONE 9**: Realizzazione dell'hardware e implementazione firmware di un sistema di monitoraggio dell' umidità e della temperatura per mezzo del sensore digitale DHT11; Realizzazione dell'hardware e implementazione firmware di un sistema di monitoraggio degli ingressi in una stanza con sensore di presenza IR e dei parametri ambientali (temperatura ed umidità) tramite sensore digitale DHT11.
- **ESERCITAZIONE 10**: Realizzazione dell'hardware e implementazione firmware di un sistema di monitoraggio della chiusura porta con encoder incrementale e knock sensor (sensore di urto).
- **ESERCITAZIONE 11**: Realizzazione dell'hardware dello specifico nodo sensore (tramite scheda NodeMCU) con connessione al sensore DHT11 e LED/relè ed implementazione del firmware necessario all'interfacciamento tramite WiFi con il nodo centrale realizzato mediante scheda Raspberry.
- **ESERCITAZIONE 12**: Realizzazione interfaccia grafica nodo centrale della WSN realizzata nella precedente esercitazione, per la visualizzazione da remoto dei dati ricevuti dai diversi nodi sensori (e trasmessi on cloud) e controllo remoto degli utilizzatori connessi ai nodi.
- **ESERCITAZIONE 13**: esercitazioni propedeutiche alla realizzazione del progetto da presentare all'esame; attività di supporto ad ogni singolo studente.
- **ESERCITAZIONE 14**: esercitazioni propedeutiche alla realizzazione del progetto da presentare all'esame; attività di supporto ad ogni singolo studente.

REFERENCE TEXT BOOKS	Teaching materials: teacher handouts.