

## **Syllabus**

## Descrizione corso

Titolo insegnamento	Fondamenti di Informatica e Programmazione dei Microcontrollori
Codice insegnamento	42174
Titolo aggiuntivo	
Settore Scientifico- Disciplinare	ING-INF/04
Lingua	Inglese
Corso di Studio	Corso di laurea in Ingegneria Industriale Meccanica
Altri Corsi di Studio (mutuati)	
Docenti	dr. Santos Miguel Orozco Soto, SantosMiguel.OrozcoSoto@unibz.it https://www.unibz.it/en/faculties/engineering/academic-staff/person/50657
Assistente	
Semestre	Primo semestre
Anno/i di corso	1
CFU	6
Ore didattica frontale	36
Ore di laboratorio	36
Ore di studio individuale	78
Ore di ricevimento previste	18
Sintesi contenuti	<ul> <li>Basics of programming in the C/C++ language</li> <li>Introductory analog and digital electronics</li> <li>Introductory motor control and sensor reading</li> <li>Introduction to computer architecture</li> </ul>
Argomenti	The course will introduce basic concepts in information and
dell'insegnamento	computer science (hardware and software), particularly those topics of fundamental importance to Engineering and confirm the theoretical learnings in lab projects.
Parole chiave	Electronics, programming, microcontroller

Insegnamenti propedeutici  Modalità di insegnamento  Frontal lectures and lab exercises.  Attendance at assigned laboratory sections is required; lecture attendance is very strongly recommended.  Obiettivi formativi specifici e risultati di apprendimento attesi  Through the application of the principles of Information Scien and Microcontroller Programming, students should be able:  1. To know basic software design procedures.  2. To know how to develop simple microcontroller programs and actuators.  4. To understand how to interface a microcontroller with sin sensors and actuators.  4. To understand the principles of simple electromechanical systems.  Applying knowledge and understanding  5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustair arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and interpret relevant data.	ce
Obbligo di frequenza  Attendance at assigned laboratory sections is required; lecture attendance is very strongly recommended.  Obiettivi formativi specifici e risultati di apprendimento attesi  Through the application of the principles of Information Scien and Microcontroller Programming, students should be able:  1. To know basic software design procedures.  2. To know how to develop simple microcontroller programs 3. To understand how to interface a microcontroller with sir sensors and actuators.  4. To understand the principles of simple electromechanical systems.  Applying knowledge and understanding  5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustair arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	ce
attendance is very strongly recommended.  Obiettivi formativi specifici e risultati di apprendimento attesi  Intended Learning Outcomes (ILO)  Knowledge and understanding  Through the application of the principles of Information Scien and Microcontroller Programming, students should be able:  1. To know basic software design procedures.  2. To know how to develop simple microcontroller programs.  3. To understand how to interface a microcontroller with sir sensors and actuators.  4. To understand the principles of simple electromechanical systems.  Applying knowledge and understanding  5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustair arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	ce
Obiettivi formativi specifici e risultati di apprendimento attesi  Intended Learning Outcomes (ILO)  Knowledge and understanding  Through the application of the principles of Information Scien and Microcontroller Programming, students should be able:  1. To know basic software design procedures.  2. To know how to develop simple microcontroller programs 3. To understand how to interface a microcontroller with sin sensors and actuators.  4. To understand the principles of simple electromechanical systems.  Applying knowledge and understanding  5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustair arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
risultati di apprendimento attesi  Knowledge and understanding Through the application of the principles of Information Scien and Microcontroller Programming, students should be able:  1. To know basic software design procedures. 2. To know how to develop simple microcontroller programs 3. To understand how to interface a microcontroller with sin sensors and actuators. 4. To understand the principles of simple electromechanical systems.  Applying knowledge and understanding 5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustain arguments.  Making judgments 6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
attesi  Through the application of the principles of Information Scien and Microcontroller Programming, students should be able:  1. To know basic software design procedures.  2. To know how to develop simple microcontroller programs.  3. To understand how to interface a microcontroller with sin sensors and actuators.  4. To understand the principles of simple electromechanical systems.  Applying knowledge and understanding  5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustain arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
and Microcontroller Programming, students should be able:  1. To know basic software design procedures.  2. To know how to develop simple microcontroller programs.  3. To understand how to interface a microcontroller with sir sensors and actuators.  4. To understand the principles of simple electromechanical systems.  Applying knowledge and understanding  5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustair arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
<ol> <li>To know basic software design procedures.</li> <li>To know how to develop simple microcontroller programs.</li> <li>To understand how to interface a microcontroller with sin sensors and actuators.</li> <li>To understand the principles of simple electromechanical systems.</li> <li>Applying knowledge and understanding</li> <li>To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustain arguments.</li> <li>Making judgments</li> <li>To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and</li> </ol>	
<ol> <li>To know how to develop simple microcontroller programs</li> <li>To understand how to interface a microcontroller with sin sensors and actuators.</li> <li>To understand the principles of simple electromechanical systems.</li> <li>Applying knowledge and understanding</li> <li>To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustain arguments.</li> <li>Making judgments</li> <li>To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and</li> </ol>	
<ol> <li>To understand how to interface a microcontroller with sin sensors and actuators.</li> <li>To understand the principles of simple electromechanical systems.</li> <li>Applying knowledge and understanding</li> <li>To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustain arguments.</li> <li>Making judgments</li> <li>To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and</li> </ol>	
sensors and actuators.  4. To understand the principles of simple electromechanical systems.  Applying knowledge and understanding  5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustair arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	<b>3.</b>
<ol> <li>To understand the principles of simple electromechanical systems.</li> <li>Applying knowledge and understanding</li> <li>To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustain arguments.</li> <li>Making judgments</li> <li>To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and</li> </ol>	nple
systems.  Applying knowledge and understanding  5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustair arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
Applying knowledge and understanding  5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustain arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
<ul> <li>5. To apply software design principles, programming, and hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustain arguments.</li> <li>Making judgments</li> <li>6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and</li> </ul>	
hardware interfacing in theoretical examples and hands-on laboratory exercises that complement the lectures and sustain arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
laboratory exercises that complement the lectures and sustain arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
arguments.  Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
Making judgments  6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	ì
6. To make autonomous judgments on the choice of the rig tools such as data types, programming approaches, or electric components. The labs will also require students to gather and	
tools such as data types, programming approaches, or electric	
components. The labs will also require students to gather and	
interpret relevant data.	
Communication of the	
Communication skills	
7. To correctly and properly present information, ideas,	
problems, and solutions during the labs.	
Learning skills  8. To acquire extended skills in Information Science and	
Microcontroller Programming and be able to use this new	
knowledge as a solid foundation for further study in more	
advanced courses in Engineering.	
Obiettivi formativi specifici e	
risultati di apprendimento	
attesi (ulteriori info.)	
Modalità di esame Examination of the course is conducted via a written exam an	
The written exam consists of two parts: i) a part with short	d lah

	questions to assess the knowledge and understanding of the theoretical fundamentals of the course topics. ii) a part with exercises on Datatypes and Operations, exercises on the dimensioning of electrical circuits, as well as exercises on code writing.  The single experiments of the lab will be examined by i) the correct functionality of the student's own implementation as described in the relative task description ii) the ability of the single students in the lab groups to explain the selected approaches iii) the level of observation of physical processes iv) the overall implementation, documentation, and appearance of the electrical circuit and software code of the selected approach.
	Formative assessment: Labs: 40%; ILOs assessed: 1-7; Summative assessment: Written exam: 60% (4 hours); ILOS assessed: 1-4,6,8;
Criteri di valutazione	Written Final Exam: Completeness and correctness of answers. Labs: Completeness and correctness of the student's own implementation, the ability of students to explain the selected approaches, the level of observation of physical processes, and the overall implementation, documentation, as well as appearance of the electrical circuit and software code of the selected approach. Students are required to receive an overall grade of greater than 60/100 points in order to pass the course.
Bibliografia obbligatoria	Smith, A. G. Introduction to Arduino: A piece of cake, CreateSpace Independent Publishing Platform, 2011. ISBN: 978-1463698348
	Hard copies available in library reserves, or can be downloaded here – <a href="http://www.introtoarduino.com/downloads/IntroArduinoBook.pdf">http://www.introtoarduino.com/downloads/IntroArduinoBook.pdf</a>
	Subject Librarian: David Gebhardi, <u>David.Gebhardi@unibz.it</u> and Ilaria Miceli, <u>Ilaria.Miceli@unibz.it</u>
Bibliografia facoltativa	Blum, J. Exploring Arduino: Tools and Techniques for Engineering Wizardry, John Wiley & Sons, 2013. ISBN: 978-1-118-54936-0

Altre informazioni	Software used: Arduino IDE freely available at: https://www.arduino.cc/en/software
	It needs to be installed on the student's personal laptop.  The Ubuntu operating system is recommended, but MacOS or Windows are also acceptable.
Obiettivi di Sviluppo Sostenibile (SDGs)	Istruzione di qualità