ELEC 1601 Foundation of Computer Systems
Notes Instructions

Format(To help you understand my notes better)

*: Not mentioned in this or before class, not necessary information.
Bold: specify a concept
Italic: emphasise
Text in Red: Important notes
Text in Yellow: Examples
Text in Purple: Java Codes and Assembly Language

Author's reminder about this course

- Outline:
  - The project is based on controlling a robot (boe-bot) with an Arduino board and sending information remotely using a communication device.
  - Lecture: Week 2-5 Computer basic knowledge, encoding information, computer memory, boolean algebra, Sequential circuit.
  - Lecture: Week 7-12 AVR Architecture, Assembly Language
  - Tutorial: Materials and questions about lecture topic
  - Lab: Week 2-6
    1. Work on several pre-defined activities programming the Arduino board with certain sensors and actuators.
    2. Design a novel application using those sensors and actuators and create a prototype.
  - Lab: Week 7-13 work on final project, report, demo, presentation.
- Study strategies by author:
  - New to computer industry? Probably this course will make you more confused. One of the biggest challenges is that you have to read nearly 10000 words materials every week! This course is about reading, theory, and you have to spend a great amount of time in labs. But do not PANIC! If you panic, nothing will succeed.
  - My notes are extremely useful at this point, although there are so many texts in some parts, but I really tried hard to summarise. Take my notes to lectures and write your thoughts on it would be a good strategy.
  - Practice, Practice and Practice. In this course you will face a lot of problems every week. Make sure you understand them. Do not put off anything. Or you will just find that information just begin to pile up and you can't handle it.
- If you purchased my notes:
  - Thank you very much. You have just made a wise decision. I do put a lot of effort in writing them, and they are simple, orderliness, understandable.

Computer Industry is changing rapidly. I'll try to keep my notes up to date.

Once you fall behind, you'll find it even harder to catch up.

ELEC 1601 Course Contents(2016 S2)

Week	Topic	Navigation
Week1	Introduction & Study Technique	1. Course Senario 2. Objectives and Milestones 3. Learning Strategies 4. Study Techniques 5. Optimal Working Session 6. Level of Computer Systems
Week2	Encoding Information	1. Binary Logic 2. Representing numbers in different bases(2, 8, 10, 16) 3. Encoding Integers: Sign & Magnitude, 2's complement 4. Floating Point 5. Encode set of symbols
Week3	Computer Memory	1. Computer memory 2. Memory Operations 3. Connection between memory and processor 4. Data Storage 5. Arrays(tables) in memory 6. Indirection
Week4	Boolean Logic and Combinational Circuits	1. Boolean Algebra 2. Logic Gates 3. Design Criteria for Digital Circuits 4. Example of combinational circuits
Week5	Sequential Digital Circuits	1. The clock 2. Flip-Flop 3. FSM
Week6	Mid-Term Examination	
Week7	AVR Architecture and Execution Environment	1. The Execution Environment 2. The Execution Cycle 3. The Stack
Week8	AVR Instruction Set Architecture	1. Types of Instruction Sets, CISC/RISC 2. Instruction format of AVR Architecture 3. The assembly language
Week9	Assembly Programs in AVR	1. Assembly Program Basic 2. Create an Execution Program from Assembly code 3. Data Definition 4. Stack and Register Management
Week10	Addressing Modes of AVR Architecture	1. Notations to describe Addressing Modes 2. Addressing Modes in the AVR Architecture
Week11	Translation of High Level Programming Constructs	1. If-else 2. Switch 3. While 4. For
Week12	Subroutine Execution	1. CALL and RET Instructions 2. Parameters and Result 3. Activation Block

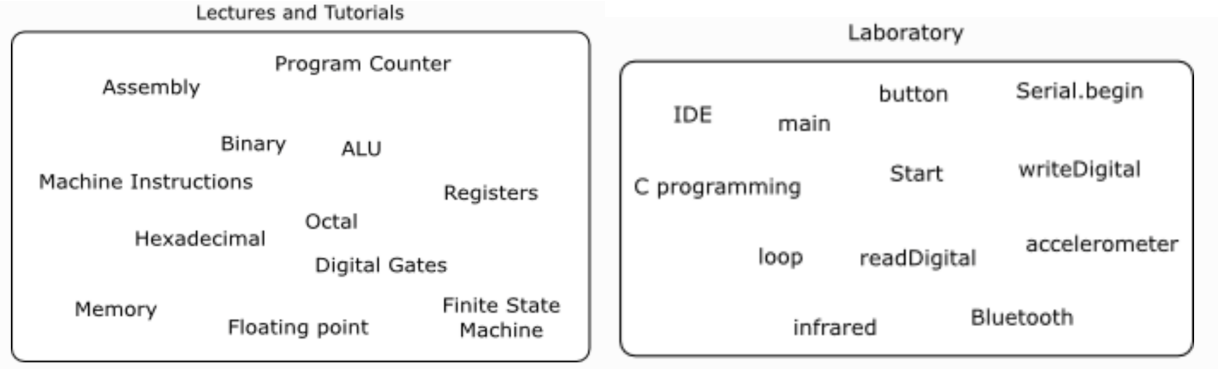
ELEC 1601 Week 1
Introduction & Study Technique

Navigation

- Course Senario
- Objectives and Milestones
- Learning Strategies
- Study Techniques
- Optimal Working Session
- Level of Computer Systems

Course Senario

You are part of a company department considering the use of digital systems such as Arduino as their next strategic step. But before moving into that space they need to know in detail the technology, its potential, how feasible is to program, and possible applications.
The task for your team is to explore these possibilities. You need to understand how digital systems work, the architecture of Arduino, its machine language, how to program it, how to connect other devices and sensors, how to read and write data from those sensors, become proficient with the tools to carry these tasks, and tests all of them in a practical project.
At the end of the semester you have to explain to the company the possibilities of these systems and propose future applications.



In the lectures and tutorials we will study the system from the point of view of a circuit and the architecture(AVR Architecture) of the microprocessor. How the circuit works and its components.
In the lab, we will approach the Arduino from the C-like programming tasks, dealing with deals with the the specific syntax two write C programs for Arduino.

Objectives and Milestones

The main objective of the course is to demonstrate how to program and communicate with a digital system.

A. Demonstrate that you understand how computers work, from the digital logic level to how they execute basic programs.
B. Design, build, configure, program and test an electronic system for a specific engineering problem observing common professional practice.
C. Write reports about the design process and its results
D. Engage in team-based design and creative tasks to solve an engineering problem.

Milestones

1. How computers encode all information they manipulate in binary logic.

2. How digital circuits are created to manipulate information encoded in binary format.
3. Which instructions are executed by computers and the steps required for that execution.
4. How to write programs in machine language (assembly).
5. How to design and configure a computer system (Arduino) using a high level programming language.

Learning strategies

- System 1(intuition and instantaneous) and System 2 (require rational thinking)
  - learning requires active rational activity, not in System 1 but in System 2.
- learning occurs when there is some degree of difficulty that is overcome
  - **Maximum Learning with Reasonable Effort (MLRE) strategies** are much more effective for academic achievement.
- DEVOTE TIME!
- NO CONGNITIVE DEBTS, NO POSTPONE!
- Take the opportunity in this course to refine your electronic device strategy (EDS).

Study Techniques

- Avoid System 1
- Avoid person bias: make sure you have learned the content, procedure or concept in a reliable way, not in system 1. Put some time to review; Put some difficulties in the path;
- Avoid massed practice!
- Avoid improvised study sessions
- Distributed Practice Practice Practice
- Interleave topics in a session.
- Keep Asking! Myself and others
- Self-explanation
- Highlighting and Underlining, not FREQUENTLY
- Summarising
- Keyword mnemonic, from complex to easy
- Imagery for text
- Time management
  - Try to pre-allocate working slots. NO INTERRUPTIONS, NO EXCUSES
  - Follow these slots with relaxing slots
  - Check how many of these slots are you able to include in a regular working day and see if you can increase them. What is your maximum? How successful are you keeping them interruption-free? How many did you fail to protect? Would you be able to quantify your cognitive debt in working slots?

Optimal Working Session

1. **Allocate a period of time** of solid work in advance. Make sure nothing interfering.
2. **Sketch a plan** of activities you will try to tackle during that period.
3. **Specify the order of the tasks** and you have all the material. DO NOT SEEK MATERIAL AT THAT TIME.
4. **Start on time**. Disable as many distractions as possible.
5. **Focus on**. If not, draft a precise description of why you cannot finish the task, and the steps you will take once the session is done, to **unblock** this situation.
6. Keep working until time is up. See your notes and try to **assess how much of your cognitive debt has disappeared**, and how much extra time you need to reduce it further.
7. Repeat these steps until the level of cognitive debt is adequate.

Level of Computer Systems

The abstraction levels of computer systems(left), The model of a computer system(right).