University of Toronto, Faculty of Applied Science and Engineering
Department of Electrical and Computer Engineering

ECE 243S – Computer Organization – 2014

Project Structure

The project is an opportunity to have fun and to express your creativity by applying what you have learned in this course. You remain working in the same group of two and with the same TA. The project portion of the course is worth 10% of the course grade.

There are 3 project lab sessions, in which you will be working under a supervision of your TA. However, there will be quite a bit of work that you will have to do outside the lab session, such as writing your project proposal and debugging your code. You will demonstrate your result during the third project lab session.

Be as creative as you would like with your project, and have fun with it! You only need to make sure you keep it relevant to the course material and unique.

Marking

The marking scheme for the project is shown in the table with more details about each component further below. Although the project is a group effort, the individual contribution of each student will be taken into account when marking.

Idea	1 mark	Coming up with a unique project idea.
Proposal	1 mark	Completing the project proposal and getting your TAs approval.
Progress Report	2 marks	You will email your code and a log of your project's progress to your TA.
Difficulty	6 marks	How challenging your project is (determined subjectively by TA).
Functionality	10 marks	Your success in completing the project core.

Idea

The first step in your project is to come up with a unique/original idea. Use this opportunity to be as creative as you can, but of course, stay relevant to the course. You should be writing assembly, interacting with peripherals, handling interrupts, interfacing C/assembly, etc. Building mechanical structures or electrical circuits does not give you any credit from an ECE243 point of view, and hence is mostly discouraged. Some of the project themes are listed at the end of this document.

Your project should be unique, that is, different than everyone else's in the class. Although we strongly encourage you to try to come up with a truly original project, similar ideas are OK, as long as there are some distinctive features/approaches that make your project stand out. For example, one can use a Lego-made car in many ways: to follow a line, to sense walls; the car can be controlled using the keyboard, push buttons, switches, etc. Project uniqueness will be determined on a first-come-first-

served basis, and we will use Piazza for this. **Please strictly adhere to the guidelines below** to make this process manageable.

There will be a Note on Piazza regarding project proposals. Starting **March 7 at 8:00am**, you are to **make a non-anonymous follow-up discussion on that post**:

- Please include your lab **day** and **station number**.
- **Content** should be a very short but yet informative-enough description of your project. “Car” or “Video game” are not good. “Keyboard-controlled car with obstacle detection and backtracking” or “VGA Tic-Tac-Toe with hex keypad” is good.

Before posting, make sure that your project idea has not been claimed by scanning all the posts. You can post only one project idea. After posting, monitor your post: TAs will check your idea and comment on it with one of the following messages:

- **“Project accepted”** – Your project idea has been accepted!
- **“Project already taken”** – Your project idea has already been claimed by someone else. In this case, you are to post another follow-up with a new project idea. The current post will be deleted to avoid duplication.
- **“Project not enough info”** - You must provide more details in order to allow differentiation of other, similar projects.

You are to post a unique idea by **March 14 at 5:00pm**. Please do this early, as you are expected to have made progress on the project by the first lab session on the week of March 14.

Proposal

After ensuring that your project idea is unique, you are to fill the **Project Proposal Form** (can be found on CoursePeer) and **give it to your TA at the beginning of the first project lab session** (week of **March 14**).

Your TA will help you to determine the scope/effort required for the project during the first project lab session. The TA will advise you on what to add/remove from your project proposal so it is sufficiently, but not overly challenging. After you implement the changes, **the TA will then approve and sign your project proposal**. You will then make **two copies** of the final filled form: one will be kept by your TA, and the other one will be for your reference. Your ability to successfully implement all that was approved in your proposal will determine your project functionality and difficulty marks.

Progress Report

Before coming to **the labs during the weeks of March 14, March 28, and April 4**, you will need to **email** two things to **your TA**:

- a snapshot of your code excluding large data files, and
- a running change log file that documents the progress made by each partner. The change log should act as a sort of journal that chronicles the development of your project and must be

consistent with the code submitted. A sample change log file snippet is shown below; you should follow this format and list which partner contributed.

Sample change log for Henry and Julie:

```
Mar 22, 2012
- Henry: Fixed bug with button 1 not triggering interrupt
- Julie: Implemented line drawing on the VGA


Mar 21, 2012
- Julie: Got timer to work in a simple test program, will integrate later
- Henry: Connected button interrupts to lego arm movement
- Julie: Wrote function to draw individual pixels on VGA
...
```

In total, you will submit 3 change logs (March 14, March 28 and April 4). The TA will use this documentation, as well as his/her observations of your time spent in the lab to grade you on the consistency of your time spent and the workload balance between partners.

Difficulty

Roughly speaking, your project should be equivalent to three new labs worth of work (picture the Lego lab + interrupt lab + processor lab, but since you have already learned those things think of 3 new things each equivalent in complexity to Lab 6 or 7). Your TA will determine your difficulty mark after the last project lab session has been completed and all TAs meet to discuss and normalize marks.

It is important to note that difficulty is rewarded only when it is

a) achieved not just proposed, and
b) relevant to the course material.

Building a fancy electronic circuit and having the Nios II turn it on is not a difficult project in the eyes of your TA. Your project should be mostly assembly and should communicate with many devices, using interrupts as appropriate, and have those all work together in some meaningful way.

Below is a guideline of what a project would receive depending on what devices it used. **Please note that this is only a guideline – the difficulty of your project will be determined in comparison with other projects in the class**. The grade in the table would be assigned assuming all of the components used are fully working and not used in a trivial manner. This is meant only to be an example, and factors such as software complexity will be considered (subjectively by your TA) on a case-by-case basis.

Grade	Components Successfully Used in Project
6	2 or more **simultaneously occurring** and **meaningful** interrupts 3 or more simple I/Os: LEDs, 7segs, VGA, buttons Some C/assembly mix (with significant amount of assembly code) 1 new complex device: PS/2, Audio Codec, Custom Instruction, Custom bus component Very complex code/algorithms
5	2 or more **simultaneously occurring** and **meaningful** interrupts 3 or more simple I/Os: LEDs, 7segs, VGA, buttons Some C/assembly mix (with significant amount of assembly code) 2 new devices: Hex keypad (rows and columns), RS-232, PS/2 Pretty complex code/algorithms
4	2 or more **simultaneously occurring** and **meaningful** interrupts 2 or more simple I/Os: LEDs, 7segs, VGA, buttons Some C/assembly mix (with significant amount of assembly code) Either: • 1 of: Hex keypad (rows and columns), RS-232, PS/2 • 3 of: Lego, JTAG UART, Timer, LCD Somewhat complex code/algorithms
3	1 **meaningful** interrupt 2 or more simple I/Os: LEDs, 7segs, VGA, buttons 2 of: Lego, JTAG UART, Timers, LCD, Hex keypad, Somewhat complex code/algorithms

Table 1: Project difficulty guideline. This is only a guideline: ***the difficulty of your project also depends on how the devices are used and will be determined in comparison with other projects in the class.***

The table below is not meant to be a quantitative measurement of difficulty; rather, it helps you understand roughly how some devices/concepts stack up against others. How you use the device also affects its difficulty, for example, using a PS/2 keyboard to observe when any key is pressed is simpler than identifying the specific key, which is simpler than also distinguishing between key-down, repeat, and key-up events.

Difficulty	Device/Concept
Less	Slider switches LEDs 7seg Displays Push buttons
	Using meaningful interrupts JTAG UART Timer Lego Controller Linking C and Assembly
	RS-232 UART VGA Adapter 16x2 LCD Digital Protoboard (GPIO) Hex keypad (rows/columns only)
	Hex keypad (rows and columns)

Difficulty	Device/Concept
	Audio Codec PS/2 Video input
	Custom instruction IrDA UART
More	Custom bus component SD Card Ethernet

Table 2. Per-device difficulty. Again, this is only a guideline.

Functionality

Your functionality grade will be determined by how successful you were at implementing the **core of your project**. The project core is a part of your project that you are committing to deliver and **it should be of at least grade 3 (Table 1)**. Your TA will help you determining if your project core is of a right difficulty. How you are penalized for your project core not being fully functional depends on:

- the severity of the bug;
- the contribution of that component to the overall function of your project; and
- the contribution of that component to your project's difficulty.

You may not necessarily receive this mark in the lab. TAs may need to consult your proposal, change log, and/or their own notes before assigning this mark.

Example Project Themes

Microsoft Paint w/ Mouse
Dance Dance Revolution type game
BlackJack
Space Invaders
Tetris
Light tracking Lego car
Lego scanner/detector
Audio visualization
Oscilloscope on VGA
Duckhunt w/ Lego light sensor
Musical bar scanner and/or player
Asteroids
Wireless DE2-DE2 communication w/ laser & light sensor
Voice recognition

One good way to come up with your own idea is to look at the DESL->Nios II->Devices webpage, and try to think of something neat you can do with some combination of those peripherals. Some previous-year projects are posted on the DESL website at http://www-ug.eecg.toronto.edu/desl/lego.html

Golden Project Rules

1. Design incrementally. Get the basics working as early as possible and then keep adding new features and new peripherals to it. For example, if the project intends to control a figure on the screen with a Playstation controller, build the game so that you can control the figure with the DE2 switches while your partner figures out how to use the PS controller, and then merge the two.
2. Have a backup plan. If the project's success is heavily dependent on a single but rather complicated task, be sure you are prepared for the possibility of that task failing. In the above example, make sure you are prepared to fall back on the switches if you don't get the Playstation controller working.
3. Demonstrate something. The worst thing you can do is have nothing to demonstrate. No matter what it is, get *some piece* of your project working on demo day.

Notice how 2 and 3 come naturally if you follow Rule 1, so please **design incrementally,** and keep your code organized and maintainable!