**Obstacle Runner (Draft)**

**By Daniel Cho (971106-2373) and Arian Javdan (970922-4910)**

**This project aims to reach the advanced level.**

# Objective and Requirements

Our game uses the concept of side-scrolling games in combination with an obstacle course. The main objective is to avoid these obstacles by either jumping or ducking depending on where the obstacles are placed on the screen (these obstacles are either on ground-level or suspended in mid-air).

The pivotal requirements are:

* Listing score in a post-game screen.
* The runner *must* be able to jump depending on when BUTTON4 is pressed on the ChipKIT board.
* The obstacles move from segment 127 to segment 0.
* The game keeps track of the score by using a counter.
* The game *must* increase the speed of the background/obstacles depending on how many obstacles you’ve passed (e.g. 10 obstacles).

# Solutions

The ChipKIT Uno32 board and the basic I/O shield will be used for the purposes of this project. The display of the I/O shield will be used for displaying the game and the buttons will provide the required inputs for the game (jumping and restarting the game).

All of the functions that are used in this game will be implemented with C programming as well as MIPS Assemply code for using interrupts. We will also be using MCB32 tools to communicate with our ChipKIT board.

# Verification

We will be testing the various functions step-by-step. First, we will focus on the runner isolated from all other functions that are in the game. Second, we will implement the background and the objects without the runner. When these two implementations work correctly, we will combine these two functions. When this process is completed, we will begin to test additional functions such as the scoreboard, the speed, randomizing obstacles in this environment.

# Contributions

*All* logic that was implemented on this project was a mutual effort, on the other hand, we took turns to write the actual code. Throughout the whole project, we have worked together simultaneously on all tasks, no exceptions.

# Reflections

We would like to admit that we started off with the project a bit later than we would have wanted. It felt like we wasted too much time on NOT understanding the SSD1306 Display Controller manual. In other words, we spent too much time doing research about the display rather than writing the actual code for the game (implementing the game logic).

Besides the previous statement, we think that the project is going relatively well. There have been a lot of problems that we encountered but we managed to solve most of the problems.