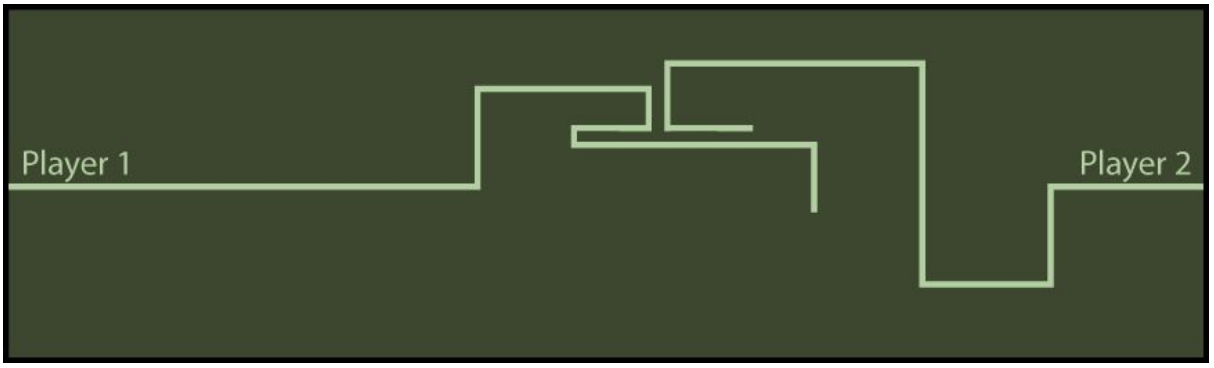
Developing a TRON game for the ChipKit Uno32

Max Dern 19940319-1357
Alexander Lundqvist 19920209-7797

Objective and Requirements

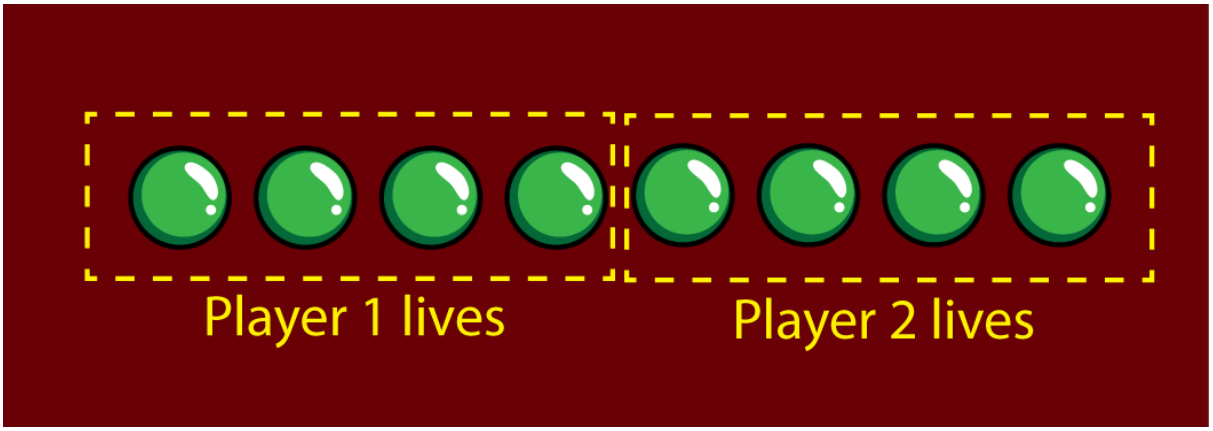
Our project goal is to develop the game Light Cycles from the movie TRON in C and then have it run on the ChipKit Uno32 with the basic I/O-shield for user interaction. The game pits two players against each other and the goal is to make the opponent's bike crash into a wall. The user controls the bike with two buttons and can change the speed of the game with the switches. Each user starts out with 4 lives indicated by the green LED:s, and loses one life when they run into a wall.



Requirements:
- Display game on built-in OLED graphical display.
- 2-player game
- Game speed can be set via the switches
- Bikes move across the screen in X and Y directions.
- Bikes can collide with themselves and walls.
- Detect when a bike collides, and turn off a light from the green LED:s

Solution

The project is written in C and developed for the ChipKIt Uno32 board using the MCB32tools. The game utilizes the small screen on the Basic I/O shield to display the interactions and the buttons to steer the bike left and right. The switches are used to add speed to the game. We use the LED:s to indicate the number of lives left for a player. A light is turned off starting either from the eight or fourth LED depending on the player.



The first player to turn off 4 lights loses the match. After a match the game will restart by itself.

Verification
The primary testing was done by continuously running the code on the chipkit and verifying that it reads and writes correctly, and that input/output behaves in the expected way. We also came up with a few test cases to properly account for examples such as ties and multiple inputs at the same time.

Contributions
Alexander handled most of what is happening on the screen, such as drawing the pixels, the steering and checking for collisions. Max worked on the user input for the switches and buttons, as well as the output on the LED:s

Reflections
The project runs well on the Chipkit but the input buttons can be a bit sensitive depending on what period time we set through the switches. This could most likely be solved by moving the button code from the segment dependent on timeoutcount.

We experimented a bit with the thought about implementing an AI. The idea was that we could've used switch 1 to disable player 2 controls, and then instead run a simple AI that would've made decisions based on 2 things. Randomized decision and self preserving decisions. An initial atempt proved that the code could've worked if we had more time, but it wasn't working good enough for the presentation.

Some general thoughts about the project. It was interesting to work with the chipkit to create the game. Altough it didn't feel like we interacted with the assembly language much at all since we only worked with C code. Had we chosen to work with external components it might have been a more interesting project but due to time constraint and us not starting early enough, it was not possible.

And since alot of code was supplied from the labs, the project felt a bit more like a programming project rather than an “embedded systems/ computer technology” project.