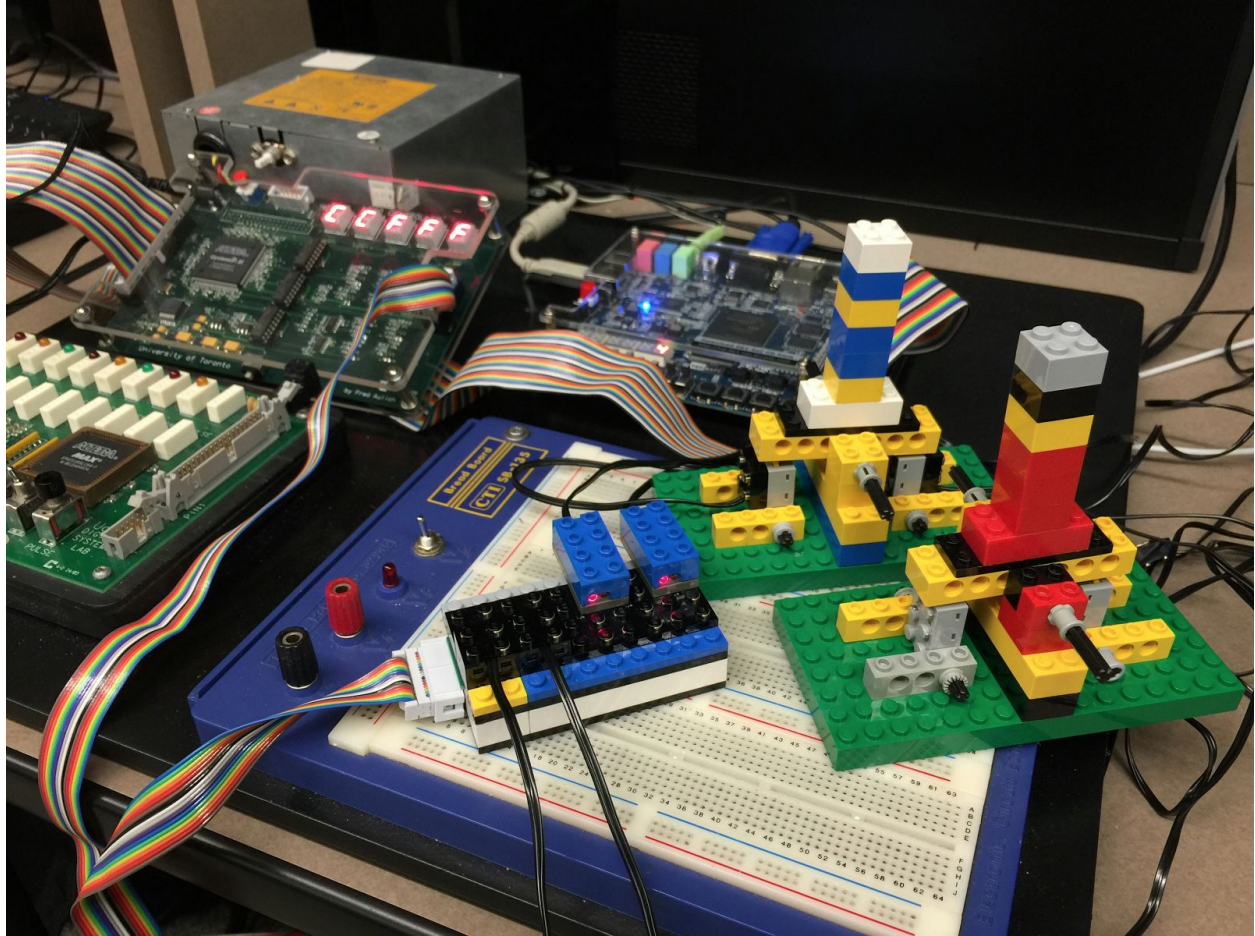


ECE385 PROJECT REPORT

Tomek Stolarczyk, Suhail Dawood
2015

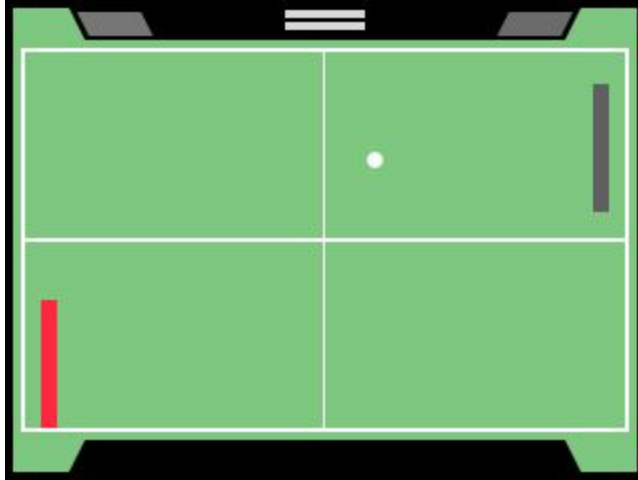
HOVER PONG

December 8,



PROJECT OVERVIEW

Hover Pong is a take on the popular game implemented in C and assembly on the DE1 board. Using Lego pressure buttons and light sensors, we created two controller types: a single axis joystick controller and a light controller (pictured above). The game is displayed on a monitor through VGA. Using the various switches on the DE1, the players can control paddle colours, game mode, audio toggle and main menu screen. The game is built on a strong foundation and allows for customization through editing the C interface code.



Above: concept design of game interface

TECHNOLOGIES USED

■ 5 interrupts

- Simultaneously occurring
- Lego pressure buttons (4 total, 2 per controller)
- Timer

■ 4 simple I/Os

- LEDs for setting/switch feedback
- 7segs for game speed indicator
- VGA for game interface
- Push buttons for reset

■ C/assembly mix

- C used for global variables, constants, game interface
- Assembly used for timer and interrupts

■ 1 complex device

- Used audio codec for interactive feedback on collisions with wall & paddle

COMPLEXITY, ALGORITHMS & ADDITIONAL NOTES

- Implemented collision detection system
- Usage of a light sensor to interpolate the values of our controller based on rigid sensor input
- Robust interface code allowing for potential customization and additions (game modes, menu screen, animated user interface)

- C was chosen for the interface drawing as it simplified the drawing process to be done pixel by pixel and by arbitrary pixels as well. This prompted quick developmental changes in the UI as we saw fit.

PROJECT LOG

November 17, 2015

1. Items can be drawn on VGA screen such as letters and pixels coloured
2. Both Light Sensors working with displaying state on LEDs. A bit noisy, but signal can be cleaned or binded directly to the Pong paddle once done.

November 24, 2015

1. VGA drawing and updating with colour. Score being displayed plus paddles can move with a nice looking playing field
2. Controller built out of Lego and push buttons
3. Controller interrupts working, triggering when a push button is pressed. ie the joystick moves in a direction either up or down

November 28, 2015

1. Created documentation for user interface C code to be interfaced with the assembly code and other game components

November 30, 2015

1. Adding a bit of code to determine collisions with walls of the game and paddles
2. Imported light sensor code into the main file
3. Created API for various elements to be updated (paddle movements, ball movements)
4. Implement memory / VGA swap for smooth frame updates
5. Added functionality for menu screen
6. Added functionality for pausing (game speed set to 0)

December 1, 2015

1. Connected game controllers to main game
2. Finalized ball physics
3. Fix bugs in collision code, ball bounces around, bounces off paddles and walls
4. Update ball code to remove previous drawing of ball and update to new ball position

5. Game timer interrupts and screen updates every 1/60 seconds

December 2, 2015

1. Updates to user interface
2. Implement JTAG UART output for debugging and logging

December 8, 2015

1. Light sensors interpolation finalized + sounds module loaded and tested
2. Scoring updated for both users and updated mapping of more buttons to provide pausing functionality
3. Game speed variable with display on HEX
4. Demo showing game working with both joystick and light sensor controllers