Extended abstract for project in IS1200

Space shooter game

Kinga Anna Koltai
Samia Serbouti

VT24

Objective and requirements

This project focuses on developing a 2D game on the Uno32 embedded system. In this strategy based arcade game we took inspiration from several well-known video games, such as the classic space invader or the (original) angry birds and came up with a game optimized for our limited resources and atypical screen aspect ratio. The goal for the player is to hit as many blocks as possible in a given amount of time. We use the buttons for aiming and shooting the bullets. The main requirements for the project were as follows:

- must use the PIC32 platform as well as the display on the I/O board
- must implement some interaction and include some program logic
- must have several difficulty levels
- must have a high-score list

Solution

This project uses the ChipKIT Uno32 board and the Basic I/O shield. More specifically, the four push-buttons and the slide switches for control both during the game and in the menu. We use interrupts triggered by the built in timer to determine when the game ends. For the highscore list we are using an array in volatile memory. Updating the display is done through a buffer which is written by the functions responsible for drawing a bullet, player, etc. All the development was done using the MCB32tools and all code is written in C or MIPS assembly.

In the game, the user enters a home menu in which they can navigate using the four buttons on the chip. The user can access and view a leaderboard with previously saved scores, a rules book (which they read by scrolling using BTN4 and BTN2) and start selecting the settings. The game offers a two player mode, unfortunately, because of time constraints, only one player mode is possible. Three levels of difficulty are offered : the higher the difficulty, the more complex the target grid is.

When entering the game, the player has 15 seconds to destroy as many targets as possible by pulling a lever up or down (using BTN4 and BTN3) that moves a canon and releases a bullet (at press of BTN2). The bullet can rebound on the wall, hit a target or is lost in space. Until the player's time is over (the remaining seconds are displayed on the LEDs on the chip) new bullets are reloaded into the canon.

Verification

We made sure to carefully plan the transition between program states, and the communication through shared variables and discussed the potential problems

and difficulties that may arise in great detail to eliminate most of the mistakes. During development we continuously tested the program changes and all of the subroutines on the chip, as well as the overall functionality of the final product. We kept an eye out for the case when a single button press may get detected as two and changed the code whenever needed to avoid different behaviour.

Contributions

We started by agreeing on the concept and making our implementation choices together regarding the game. The main functions of the game (displaying the player and bullet, aiming, shooting) and most of the menu system was coded by Samia Serbouti. The leaderboard, grid, timers and interrupts and the structure of the game was written by Kinga Koltai.

Licenses

The base of the project was done on files written by Axel Isaksson for the labs of course IS1200. We also used several functions (that we commented as such) found on the course's resource page `https://github.com/is1200-example-projects/hello-display/blob/master/main.c` Copyright (c) 2015, Axel Isaksson

Reflections

After completing the project, we realized it was a good strategy to start coding directly for two players: this way, the structure and architecture was ready for two players at the start. In the last hours of development, we realized we wanted to simplify the game for just one player and it was easy to make that change. However, we should have started by developing the game and not the navigation and the menus. Indeed, we had different approaches to displaying on the OLED and we had initially thought of showing the timer and scores on the screen during game mode but it ended up not being possible or would have required a lot more time.