

Design of embedded systems - Advance course

Project proposal

Adam Dalentoft, dat11ada@student.lu.se
Simon Wallström, dat11swa@student.lu.se
Viktor Sannum, dat11vsa@student.lu.se

September 4, 2015

1 Introduction

This is the project proposal for our project in the course "Design of embedded system - Advanced Course". Our proposed project goal is to design and construct a breakout style video game system. In a breakout type game, the player controls a paddle along a path which is used to bounce one or more balls onto the playfield. The goal is to bounce the balls against a group of blocks on the playfield until it is cleared of blocks, at which point the player wins. If any balls fall beyond the player, the player loses.

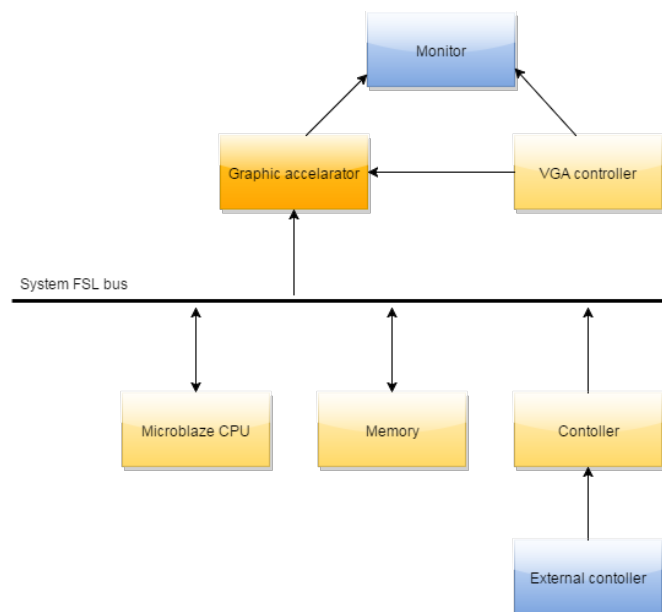


Figure 1: Diagram over the suggested design

2 Functionality

The solution should include a processing unit (the FPGA board), a screen to view the actual game on and some sort of input device to control the paddle in the game. For the VGA output, some sort of VGA controller is needed. The different parts of the system will be connected over the FSL bus.

3 Software

Software wise the game will consist of a game model / state, a main game loop, graphics routines and some sort of I/O handling routine. The game state is described by various classes representing the different entities in the game: Play field, block, ball, paddle and score/level counter. The game loop takes care of updating the state continuously as the game progresses, reading I/O and moving the paddles, moving the ball et.c. The graphics routines handle the VGA interface, copying sprites / drawing entities to the correct position on the screen et.c. Parts of this may be implemented as a hardware component. The I/O handling consists of code interfacing the I/O device(s), reading input and converting it to game actions. The game loop and graphics output will have to be synchronized to the desired refresh rate of the screen. A resolution of 640 x 480px @ 60hz would give a real time requirement of 16ms of rendering time per frame. If this is not enough, the resolution or refresh rate will have to be lowered, hardware requirements increased or game code simplified. In order to save memory it might be necessary to use a reduced color mode or some sort of palletted graphics.

4 Hardware

The project is based on a Digilent Nexys-3 Spartan 6 FPGA board, on this board a Microblaze microprocessor, a (custom if needed) VGA and PS/2 controller will be implemented.

Some of the graphics will be hardware accelerated in a custom made GPU, in the simplest form of a sprite blitter. For this rather simple application no extra memory besides the BRAM is needed.

5 I/O

5.0.1 Input

To start with input will be two buttons, possible on the Nexys board, to steer the paddle left and right. The buttons on the board might be changed to a different kind of controller.

5.1 Output

Output will be a graphic representation of the game on a monitor via the VGA controller.

6 Possible improvements

Adding sound to the game when the ball bounce and bricks break and even music could be a nice addition. This would require some sort of audio processor, possibly with it's own processor core.

Also the input method might be changed to a PS2 keyboard instead of the on board keys or some custom built controller with an analog steering wheel for that retro arcade feeling.

On the game side multiple balls and different kinds of blocks could be added along with special effects on exploding blocks and particle effects. Some hardware changes might be needed to handle the new graphic requirements, like an accelerator fore explosion physics. For even more fancy graphical effects a simple pixel shader processor could be built.

7 Time plan

Week	Adam	Simon	Viktor
1	Writing project proposal	Writing project proposal	Writing project proposal
2	Design	Design	Design
3	CPU, Memory, bus	Input controller	VGA controller
4	Game logic	Game logic	GPU accelerator
5	Game logic	Game logic	GPU accelerator
6	Testing	Testing	Testing
7	Report, demo, presenation, testing		
8	Write report	Write report	Write report

Table 1: Time plan