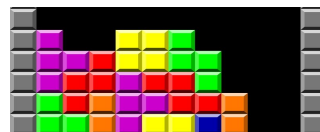


DE1-SOC Tetris Game

Zihao Wang, Shengyue Guo, Shibo Sheng

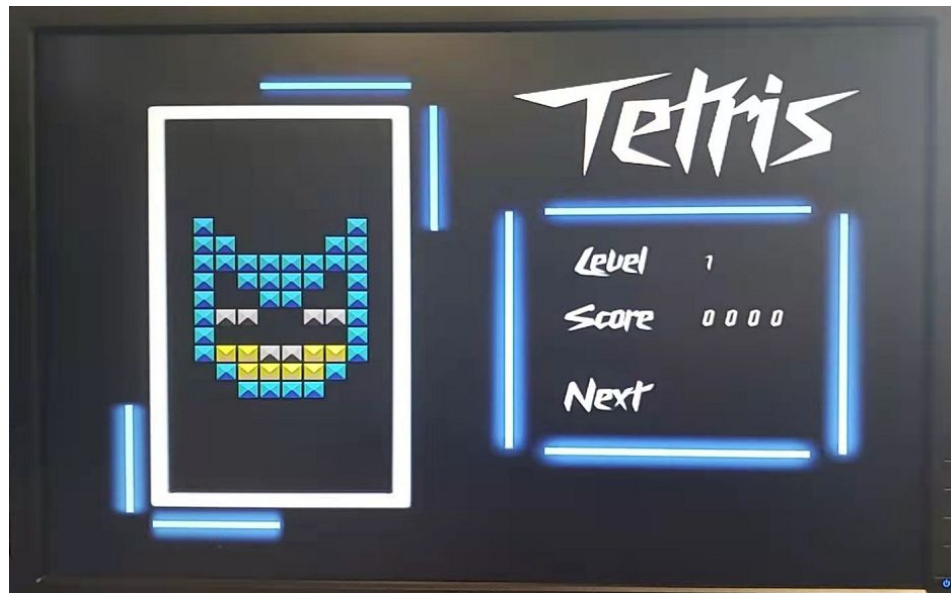
Prof. Stephen A. Edwards

Presented on May 12th

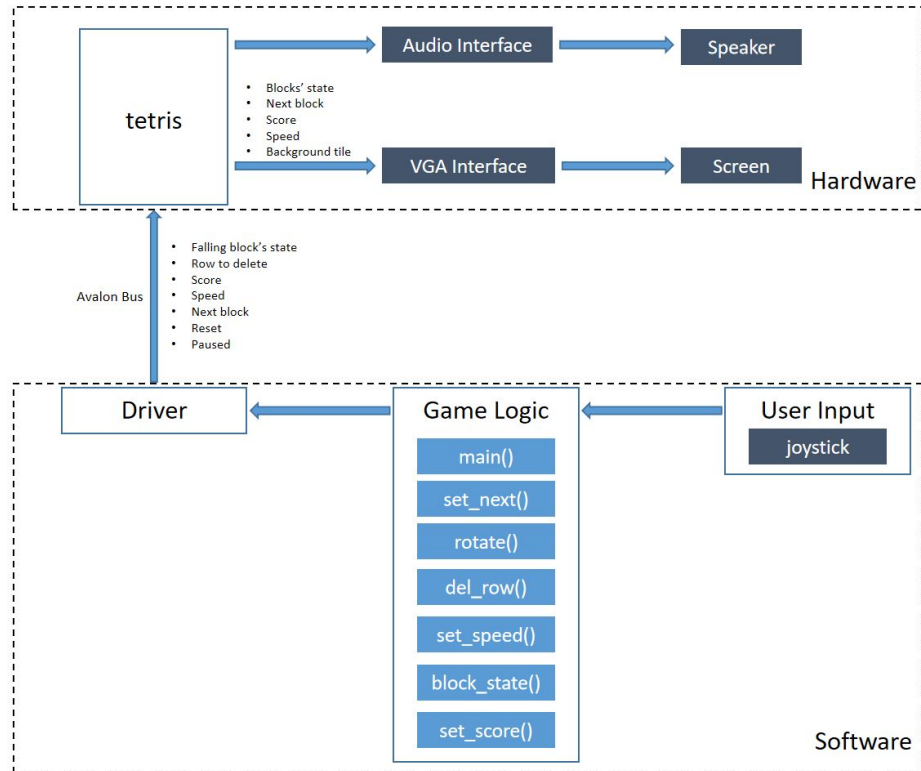


Project Overview

- Tetris Game is a puzzle video game created by Soviet software engineer Alexey Pajitnov in 1984.
- The team rebuilt the game that has 3 levels of difficulties with each level increasing the speed of the falling block along with quicker audio output.
- The team developed the game on a DE1-SoC FPGA board and players control the game via a PS2 controller for the purpose of more accurate and faster control.

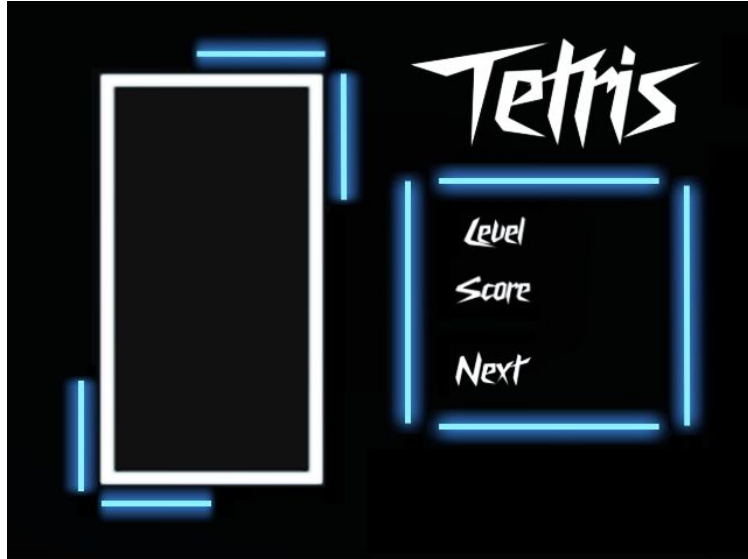


System Architecture



Hardware Design

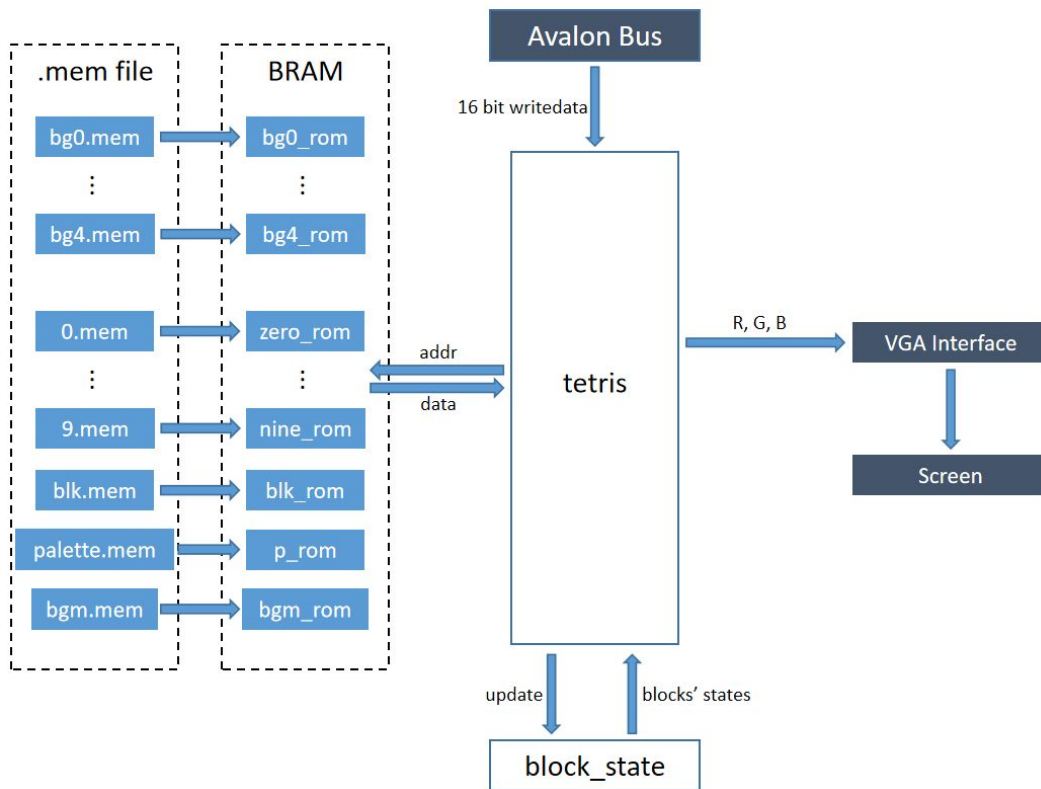
1. Graphic Design



0 1 2 3 4 5 6 7 8 9

Hardware Design

2. Block Diagram



Audio

Music Conversion:

- Downloaded an 8-bit bgm music from the internet and converted it to 16 bits mono PCM.wav.
- Used Audacity to convert the music into an 8kHz one-channel wav file.
- Wrote a python script to convert the .wav file into .mem file, filter out the useless text in the .mem file and shorten the audio to eliminate the noise at the end of the audio file.
- Manually divide the frequency by 6250 in order to get the proper audio output (The clock's frequency is 50MHz and our music's sample frequency is 8kHz)
- Wrote the audio output code in the main hardware code file.



Audio

code:

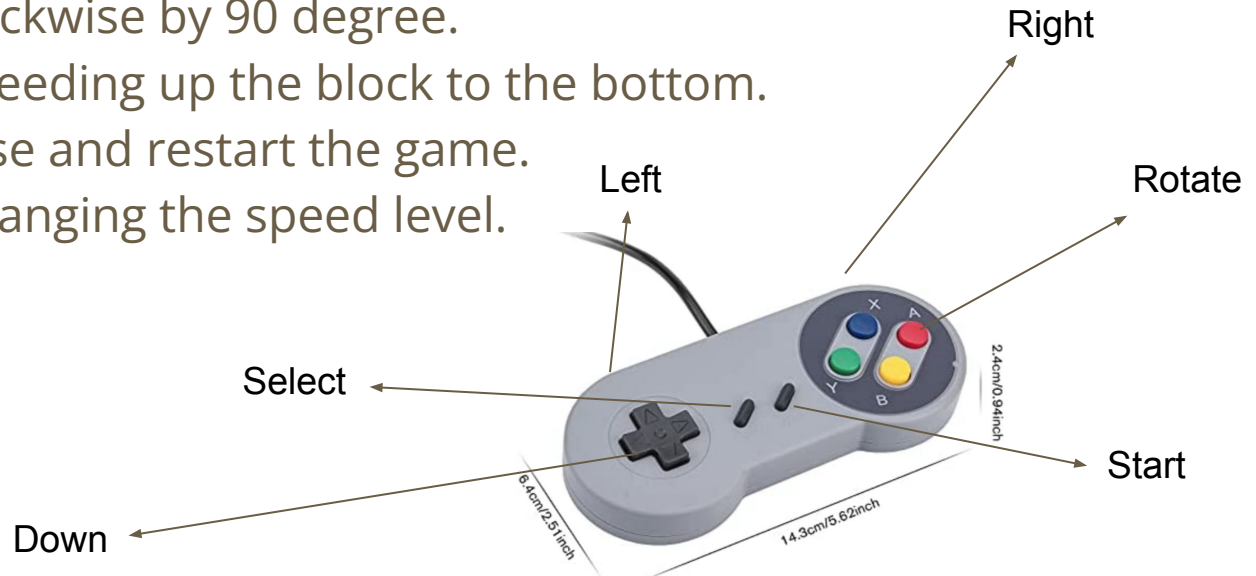
```
// music
if (divider < interval[speed - 1]) begin
    divider <= divider + paused;
    avalon_streaming_source_l_valid <= 0;
    avalon_streaming_source_r_valid <= 0;
end
else begin
    divider <= 0;
    bgm_rom_addr <= bgm_rom_addr + paused;
    if (bgm_rom_addr >= 92897)
        bgm_rom_addr <= 0;
    avalon_streaming_source_l_valid <= paused;
    avalon_streaming_source_r_valid <= paused;
    avalon_left <= bgm_rom_data;
    avalon_right <= bgm_rom_data;
end
end
```



Software Design

1. User Input

- Left, right move of the block, one press represent one unit of move.
- Rotate the block clockwise by 90 degree.
- Down button for speeding up the block to the bottom.
- Start button to pause and restart the game.
- Select button for changing the speed level.



Software Design

2. Game Logic

The software side is designed to be a single thread looping program. The application will do the following:

- Take the joystick input and process the keys.
- The game status is saved as a 20×10 matrix, and after the keys are pressed, the matrix will be updated.
- Send the corresponding instructions to the hardware modules.
- Each loop is considered a frame with timeout, and is used for controlling the automatic falling down.

Software Design

3. Avalon Bus Data

Addr	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Comment
0	row					col				value					e		Block's state
1												row_to_del				Row to delete	
2	score[3]				score[2]				score[1]				score[0]				Player's score
3														value_next			Next block
4															speed		Game speed
5															reset		Reset from sw
6															o	p	Paused or not Over or not

Problem Encountered

1. Hardware:
 - Storage space
 - Latency
2. Audio:
 - Qsys
 - Music conversion
3. Software:
 - Joystick input



