

CSEE 4840 Final Presentation

Group: Crazy-Arcade

Yuqi Lin, Yelin Mao, Hongkuan Yu

UNI:yl5334, ym3000, hy2819

Spring 2024

Short Introduction

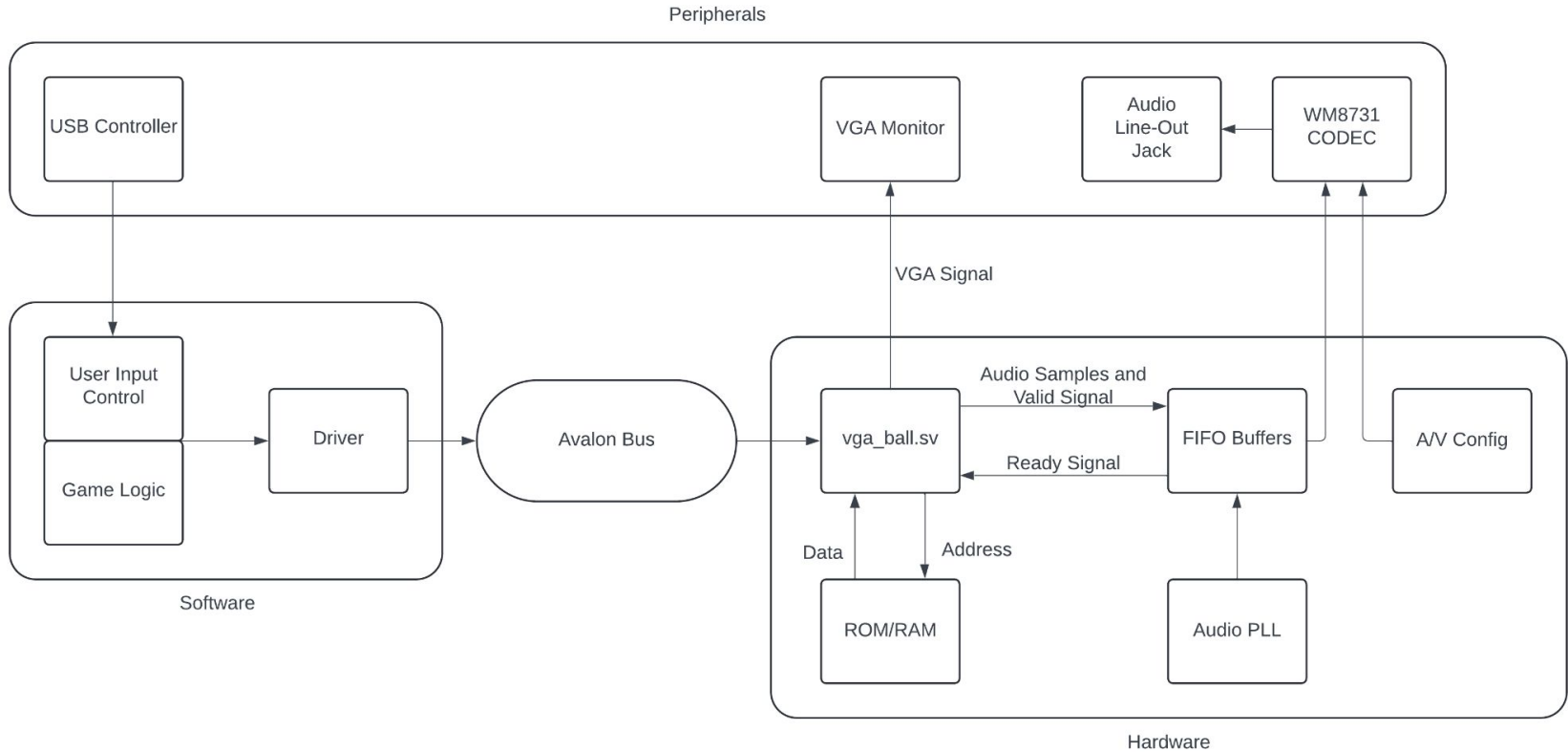
We develop a "Crazy Arcade" game (shown in Figure below) on the DE1-SoC board, leveraging SystemVerilog for hardware implementation and C for the game's algorithm. The game features two players competing in real-time, navigating through a maze, and placing bombs to destroy another player. Bombs will explode after a certain time, with the affected distance as a "+" (up, down, left, and right).
















Game Rules

At the beginning of each game, players will spawn on opposite corners of the screen, surrounded by randomly generated obstacles. There are fixed obstacles and temporary obstacles. Players can reach each other through different paths and place bombs, exploding after a set of times, to block another's path and destroy temporary obstacles. The bomb has an affected distance, and within this distance, the explosion will kill nearby players. So it is obvious that if obstacles and a bomb block the path of a player, then he has no chance but only waits for the explosion and gets killed! If one player is killed, the surviving player wins the game. After each game concludes, players respawn at their starting positions with a new randomly generated map.

System Block Diagram

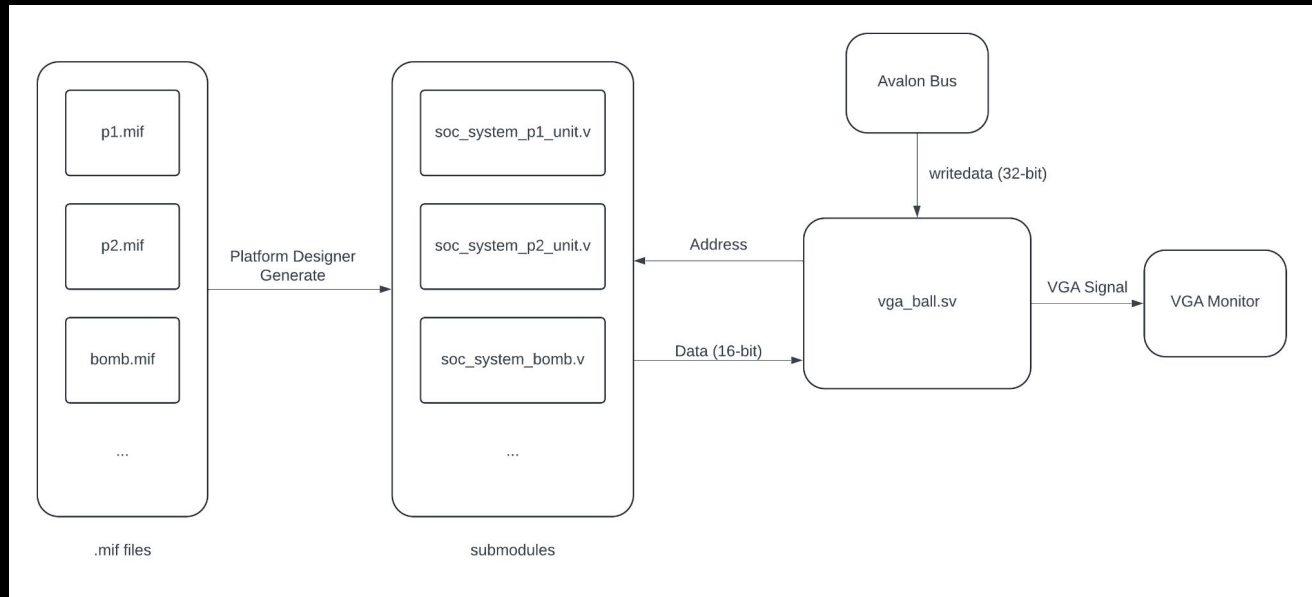


Total Memory Usage

Name	Graphics	Pixel Size	MIF File Length	MIF File Width	Total Bytes Required
Player 1		16 x 16 x 4	1024	16 Bits = 2 Bytes	2048
Player 2		16 x 16 x 4	1024	16 Bits = 2 Bytes	2048
Player 1 Die		16 x 16	256	16 Bits = 2 Bytes	512
Player 2 Die		16 x 16	256	16 Bits = 2 Bytes	512
Bomb		16 x 16	256	16 Bits = 2 Bytes	512
Fire Center		16 x 16	256	16 Bits = 2 Bytes	512
Fire Horizontal		16 x 16	256	16 Bits = 2 Bytes	512
Fire Vertical		16 x 16	256	16 Bits = 2 Bytes	512
Fixed Wall		16 x 16	256	16 Bits = 2 Bytes	512
Temporary Wall		16 x 16	256	16 Bits = 2 Bytes	512
Start Info		16 x 256	4096	16 Bits = 2 Bytes	8192
Player 1 Win		16 x 96	1536	16 Bits = 2 Bytes	3072
Player 2 Win		16 x 96	1536	16 Bits = 2 Bytes	3072
Map			1200	8 Bits = 1 Byte	1200
Explosion Sound			12109	16 Bits = 2 Bytes	24218
Place Bomb Sound			15168	16 Bits = 2 Bytes	30336
					78282

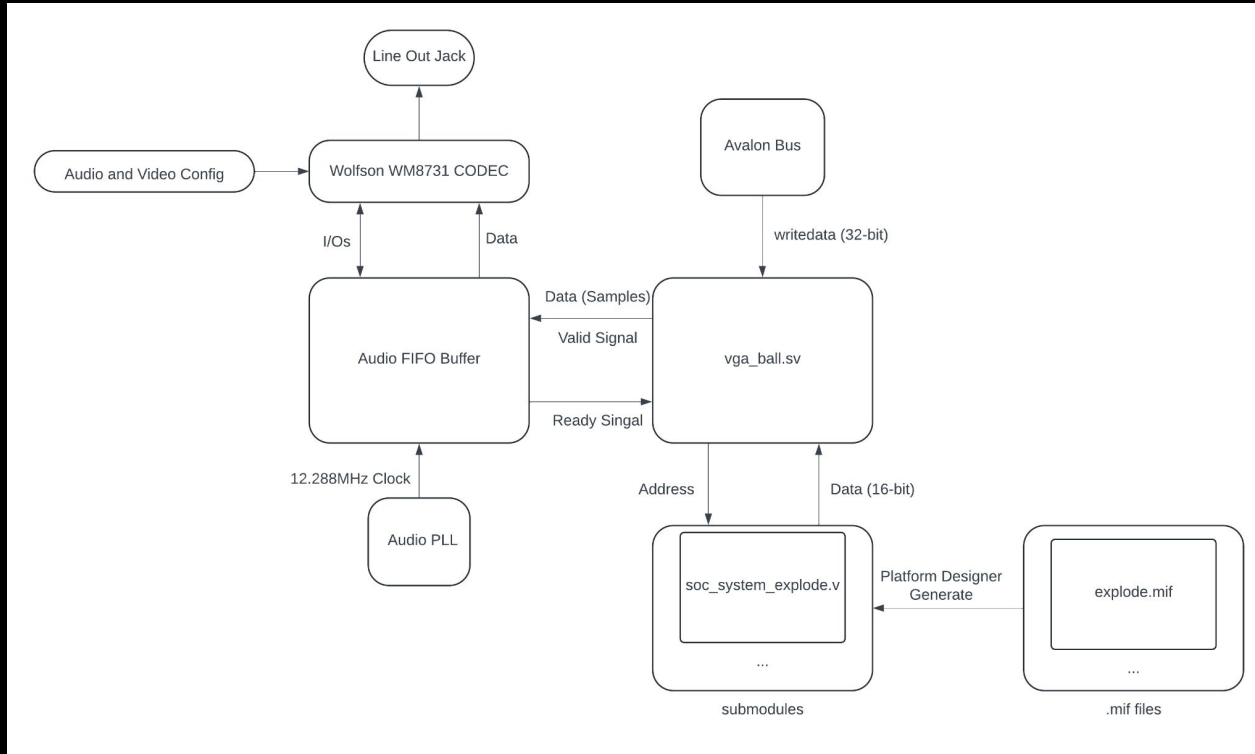
Graphics

The main SystemVerilog of hardware is written in "vga_ball.sv". It contains submodules (generated by Platform Designer) for calculating screen position and assigning RGB pixel coordinates to the VGA output. Data is transferred from software to "vga_ball.sv" in 32-bit through the Avalon bus interface through the device driver, specifying when and where the corresponding sprite should be displayed. Using these inputs and the hcount and vcount coordinates, "vga_ball.sv" sends addresses to the corresponding On-Chip Memory ROM of sprite and gets the 16-bit returned data. Then, "vga_ball.sv" slices the returned data and determines the pixel color.



Audio

We obtained the .mif file of our bomb placing and explosion sound as .wav format and converted them to .mif file for the FPGA. They are 8 kHz samples. The .mif file loads sound data into the On-Chip Memory ROM. We then adjusted the qsys connection to ensure that the audio sample was correctly routed to the Wolfson WM8731 CODEC.



Fina Connection

Platform Designer - soc_system.qsys (/homes/user/stud/fall23/ym3000/EE4840/project-hw/soc_system.qsys)

File Edit System Generate View Tools Help

System Contents Address Map Interconnect Requirements

System: soc_system Path: p1_unit

Use Connections

Use	Name	Description	Export	Clock	Base	End	RQ	Tags	Opcode Name
<input checked="" type="checkbox"/>	E clk_0	Clock Source	clk	exported					
<input checked="" type="checkbox"/>	clk_in_reset	Clock Input	clk_reset	Double-click to					
<input checked="" type="checkbox"/>	clk	Reset Input	clk	Double-click to					
<input checked="" type="checkbox"/>	clk_reset	Reset Output	clk_reset	Double-click to					
<input checked="" type="checkbox"/>	E hps_0	Arria V/Cyclone V Hard Proc...							
<input checked="" type="checkbox"/>	h2f_user1_clock	Clock Output	hps_0	Double-click to					
<input checked="" type="checkbox"/>	memory	Conduit	hps_0	Double-click to					
<input checked="" type="checkbox"/>	h2f_reset	Reset Output	hps_0	Double-click to					
<input checked="" type="checkbox"/>	h2f_wai_clock	Clock Input	h2f_wai_master	Double-click to					
<input checked="" type="checkbox"/>	h2f_wai_master	AXI Master	h2f_wai_slave	Double-click to					
<input checked="" type="checkbox"/>	h2f_wai_slave	AXI Slave	h2f_wai_master	Double-click to					
<input checked="" type="checkbox"/>	h2f_wai_clock	Clock Input	h2f_wai_slave	Double-click to					
<input checked="" type="checkbox"/>	h2f_wai_master	AXI Master	h2f_wai_slave	Double-click to					
<input checked="" type="checkbox"/>	h2f_irq0	Interrupt Receiver	h2f_irq0	Double-click to					
<input checked="" type="checkbox"/>	h2f_irq1	Interrupt Receiver	h2f_irq1	Double-click to					
<input checked="" type="checkbox"/>	E vga_ball_0	VGA Ball	clk	Double-click to					
<input checked="" type="checkbox"/>	clock	Clock Input	clk_0	Double-click to					
<input checked="" type="checkbox"/>	reset	Reset Input	clk_0	Double-click to					
<input checked="" type="checkbox"/>	avalon_slave_0	Avalon Memory Mapped Slave	vga	Double-click to	0x0000_0000	0x0000_007f			
<input checked="" type="checkbox"/>	vga	Conduit	vga	Double-click to					
<input checked="" type="checkbox"/>	avalon_streamin...	Avalon Streaming Source	clk_0	Double-click to					
<input checked="" type="checkbox"/>	avalon_streamin...	Avalon Streaming Source	clk_0	Double-click to					
<input checked="" type="checkbox"/>	E audio_pll_0	Audio Clock for DC-series Boa...	ref_clk	Double-click to					
<input checked="" type="checkbox"/>	ref_clk	Clock Input	audio_pll_0	Double-click to					
<input checked="" type="checkbox"/>	ref_reset	Reset Input	audio_pll_0	Double-click to					
<input checked="" type="checkbox"/>	audio_clk	Clock Output	audio_pll_0	Double-click to					
<input checked="" type="checkbox"/>	reset_source	Reset Output	audio_pll_0	Double-click to					
<input checked="" type="checkbox"/>	E audio_and_vide...	Audio and Video Config	clk	Double-click to					
<input checked="" type="checkbox"/>	clk	Clock Input	audio_and_video...	Double-click to					
<input checked="" type="checkbox"/>	reset	Reset Input	audio_and_video...	Double-click to					
<input checked="" type="checkbox"/>	avalon_av_config...	Avalon Memory Mapped Slave	audio_and_video...	Double-click to					
<input checked="" type="checkbox"/>	external_interface	Conduit	audio_and_video...	Double-click to					
<input checked="" type="checkbox"/>	E audio_0	Audio	clk	Double-click to					
<input checked="" type="checkbox"/>	clk	Clock Input	audio_0	Double-click to					
<input checked="" type="checkbox"/>	reset	Reset Input	audio_0	Double-click to					
<input checked="" type="checkbox"/>	avalon_left_chan...	Avalon Streaming Source	audio_0	Double-click to					
<input checked="" type="checkbox"/>	avalon_right_chan...	Avalon Streaming Sink	audio_0	Double-click to					
<input checked="" type="checkbox"/>	avalon_left_chan...	Avalon Streaming Sink	audio_0	Double-click to					
<input checked="" type="checkbox"/>	avalon_right_chan...	Avalon Streaming Sink	audio_0	Double-click to					
<input checked="" type="checkbox"/>	external_interface	Conduit	audio_0	Double-click to					
<input checked="" type="checkbox"/>	E p1_unit	On-Chip Memory (RAM or ROM...	clk1	Double-click to					
<input checked="" type="checkbox"/>	clk1	Clock Input	p1_unit	Double-click to					
<input checked="" type="checkbox"/>	s1	Avalon Memory Mapped Slave	p1_unit	Double-click to	0x0000_1000	0x0000_17ff			
<input checked="" type="checkbox"/>	reset1	Reset Input	p1_unit	Double-click to					
<input checked="" type="checkbox"/>	E p1_die	On-Chip Memory (RAM or ROM...	clk1	Double-click to					
<input checked="" type="checkbox"/>	clk1	Clock Input	p1_die	Double-click to					
<input checked="" type="checkbox"/>	s1	Avalon Memory Mapped Slave	p1_die	Double-click to	0x0000_2000	0x0000_2fff			
<input checked="" type="checkbox"/>	reset1	Reset Input	p1_die	Double-click to					
<input checked="" type="checkbox"/>	E p2_unit	On-Chip Memory (RAM or ROM...	clk1	Double-click to					
<input checked="" type="checkbox"/>	clk1	Clock Input	p2_unit	Double-click to					
<input checked="" type="checkbox"/>	s1	Avalon Memory Mapped Slave	p2_unit	Double-click to	0x0000_3000	0x0000_3fff			
<input checked="" type="checkbox"/>	reset1	Reset Input	p2_unit	Double-click to					

Current filter:

Type	Path	Message
Warning	3 Warnings	
Warning	soc_system.audio_0.avalon_left_channel_source	audio_0.avalon_left_channel_source must be connected to an Avalon-ST sink
Warning	soc_system.audio_0.avalon_right_channel_source	audio_0.avalon_right_channel_source must be connected to an Avalon-ST sink
Warning	soc_system.audio_and_video_config_0	audio_and_video_config_0.avalon_av_config_slave must be connected to an Avalon-MM master
Info	2 Info Messages	
Info	soc_system.hps_0	HPS Main PLL counter settings: n = 0 m = 73
Info	soc_system.hps_0	HPS peripheral PLL counter settings: n = 0 m = 39

0 Errors, 3 Warnings

Generate HDL... Finish

User Input

- NES USB controller
- Directional keys for movement
- Press 'A' to plant bombs
- Press 'B' to start the game and plant breakable walls

Constant	Constant	Constant	Left/Right	Up/Down	A, B	Select, Start
----------	----------	----------	------------	---------	------	---------------



Field Value	Button Pressed
0xFF (255)	Left & Down
0x00 (0)	Right & Up
0x2F (47)	A
0x3F (63)	
0xAF (175)	
0xBF (191)	
0x4F (79)	B
0x5F (95)	
0xCF (207)	
0xDF (223)	

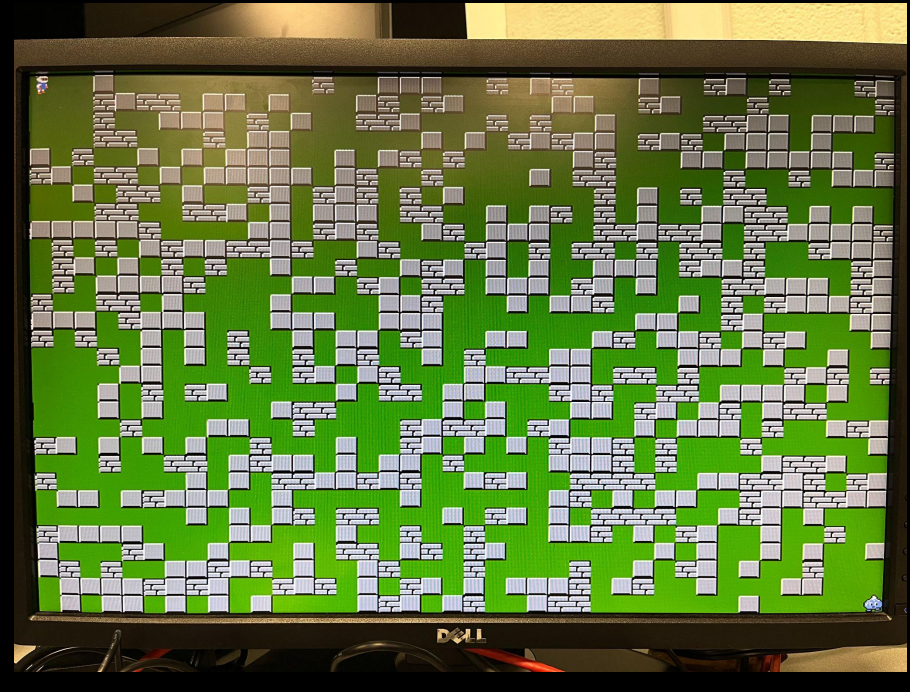
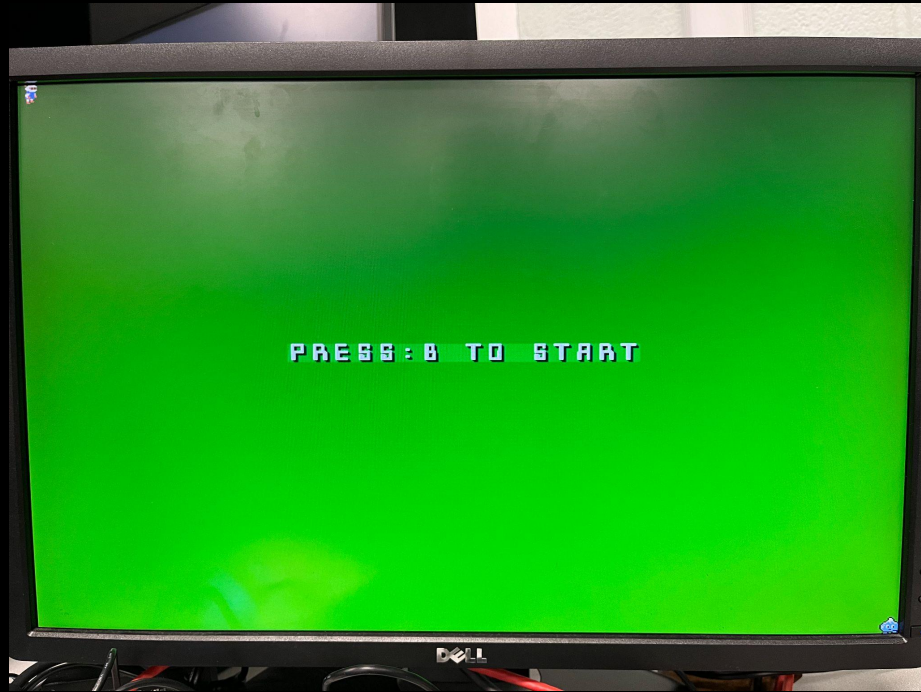
Avalon Bus Interface

- 17*32 bits registers and each has offset of 4 bytes
- Detailed data transferring is shown below

Address	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Description	
0x04																																		Player 1 coordinates
0x08	Start on																																	Player 1 states
0x0C																																		Bomb's x coordinate
0x10																																		Bomb's y coordinate
0x14																																		Fire center's x coordinate
0x18																																		Fire center's y coordinate
0x1C																																		Fire up's x coordinate
0x20																																		Fire up's y coordinate
0x24																																		Fire down's x coordinate
0x28	Start on																																	Fire down's y coordinate
0x2C																																		Fire left's x coordinate
0x30																																		Fire left's y coordinate
0x34																																		Fire right's x coordinate
0x38																																		Fire right's y coordinate
0x3C																																		Player 1 fire center
0x40																																		Player 1 fire up
0x44	Map chipselect	Map write																																Map information
																																		Map address
																																		Map input

Initialization

- Require both players to press B to start the game
- Initial random generated map and players' initial spawn location



Conclusion

- We complete all features listed in the design document
- Breakable wall (extra)
- Random generated map (extra)

Thank you!