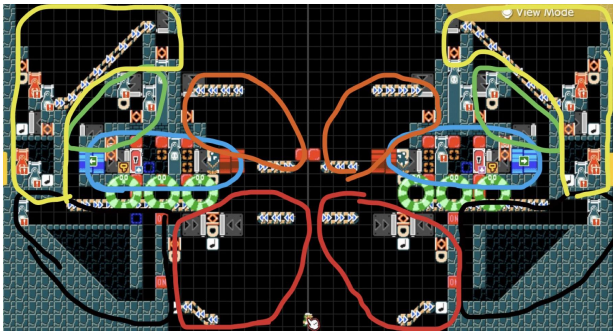
Shotgun Game Explanation

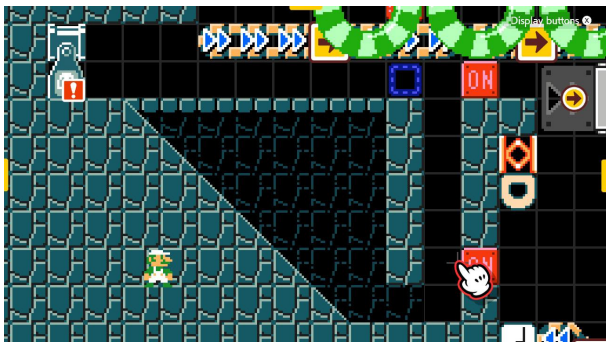
So this game is an attempt to mimic the schoolyard game of reloading, firing, and blocking on the same simultaneous turn basis that rock-paper-scissors uses. That is to say, on each turn, each player chooses between the three options without knowing what the other player will choose, and then show each other their choice at the same time.

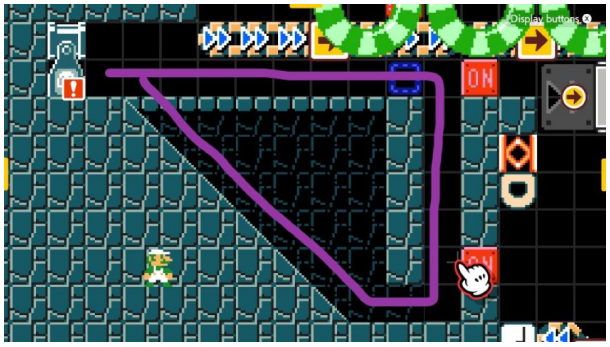
In real life, this is usually done by two people hitting their knees twice and then immediately indicating what their play is: thumbs back indicates a reload, a hand shotgun indicates a shot, and hands crossing the chest indicates a block.

The rules of the game are that you must reload before you can shoot once (rules vary on whether ammo stack, but in my level design's case, they do stack, I couldn't figure out how to restrict it to one without risking that the player have no ammo). If a player shoots while the other player shoots or blocks, the game continues. But if a player shoots while the other player is reloading, they win.



So here's the arena I've designed. I've marked off the main zones to explain each one in more detail:
Black - timer
Blue - player cage
Yellow - reloading area
Orange - firing area
Green - blocking area
Red - killzone





Timer

This one's pretty easy to follow. The cannon shoots a single buzzy beetle shell which will activate the first on/off block and then fall down and activate the second on/off block. This leaves the blocks on blue for about half a second. Then the shell takes a few more seconds to climb up the hill before resetting the cycle.

So the cycle is:

- red for about 3.5 seconds
- blue for about 0.5 seconds

I have the same timer built into both sides of the arena so that both of the players may be able to see the timer. However, this will not work if the players enter the arena and activate the cannon at different times, desyncing their cycles. The only way this design works is if both players enter the arena at the same exact time, one of the flaws with this design.

Player Cage

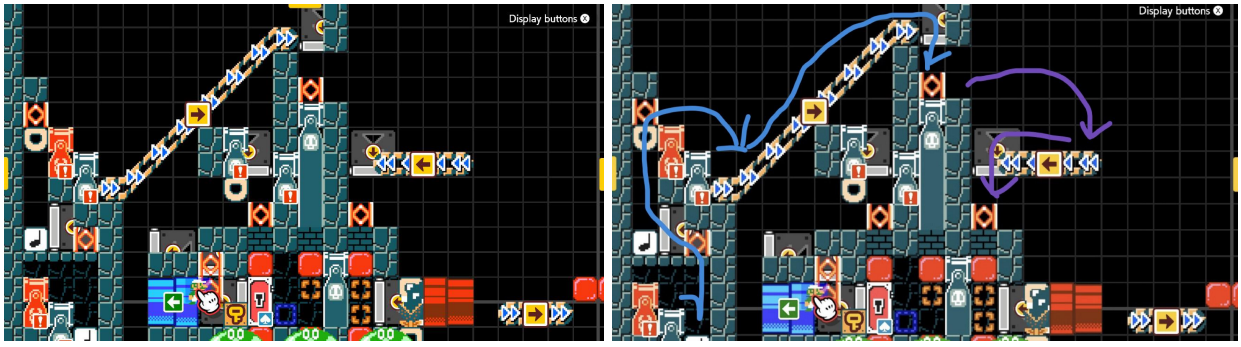
This place where the player is unfortunately is the most complicated part of the level. The player is first shoved into the cage with those sideways springs on the left. Then, the player must hit 1 of three brick blocks, labelled 1, 2, and 3 on the top. The player should stand on the lower spots labelled 1, 2, and 3, under the brick blocks so that when the on/off blocks flicker to blue, they will bounce up from the bumper to the respective brick block. I did a lot of trouble-shooting on the timing and the design of this setup to make sure that players should

almost always be able to hit one block even if they know what they are doing, while also making it near impossible to hit a block twice or hit two different blocks.

Brick block 1 is the block option. Brick block 2 is the reload option. Brick block 3 is the firing option. The cannon blocking brick block 3 will be removed after the first cycle (by the blue block at its base, similar to the left of it). The P-blocks are simply indicators.

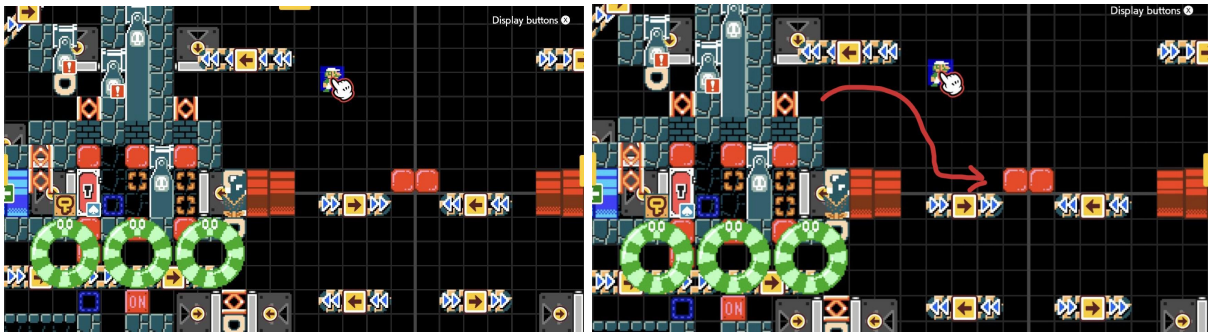
Note - I would actually extend the blue blocks upward (putting 2 more on top of the existing blue blocks there), I don't know why I designed it with just 1 block at the bottom since I don't think separating them into distinct columns while blue blocks are turned on would possibly hurt the game design.

Now on the right you see a pipe blocked by a muncher. If that muncher is removed (unlocking the spike ball), then the player in the cage is killed and the key is unlocked and goes to the nearest player, which should naturally by the competing player, as long as the arenas are spaced properly. The cause of this win condition is discussed in the killzone.



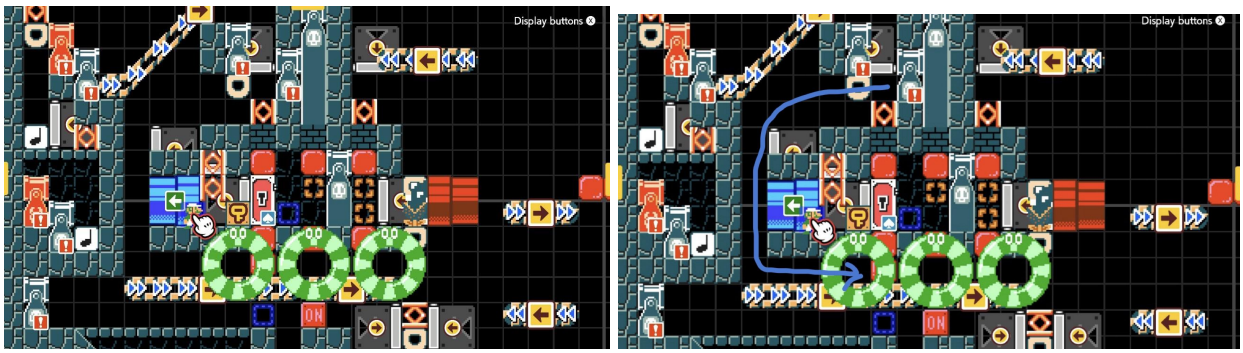
Reloading Area

This part is pretty simple, but was a pain to design. The four cannons there are all stocked with munchers, which act as the game's ammo. They travel by the note blocks, springs, and conveyor belts to the spot above the sideways spring (follows blue arrows). When a player hits the reload brick block, that spring is pushed upward and shoves the muncher out onto the conveyor on the right, and over the spring on the right (follows purple arrows), giving the player the ability to fire it on the next cycle.



Firing Area

Assuming a muncher is ready to be fired, when the player hits the brick block under the muncher it will be pushed down onto the conveyor belt and stopped by the red block.



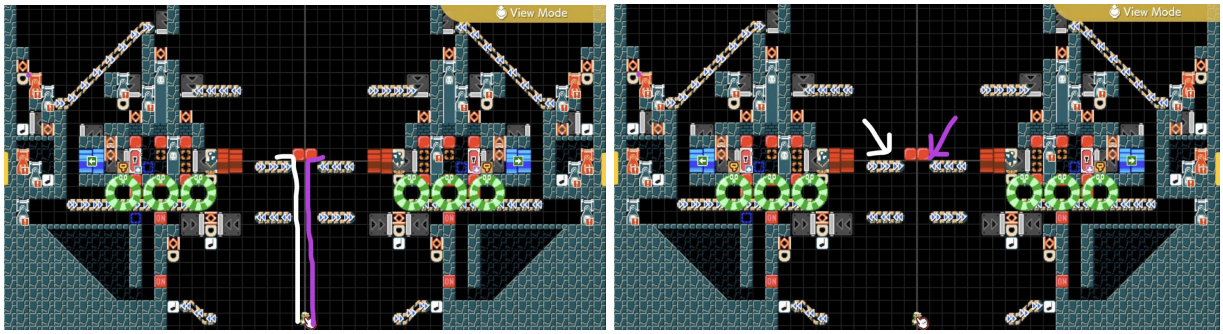
Blocking Area

_____The two cannons located above the blocking brick block are also stocked with munchers, and will hopefully refill the blocking station when empty. When a player hits the blocking brick block, the muncher will be pushed out to the conveyor below and stopped by the red block.

Interactions

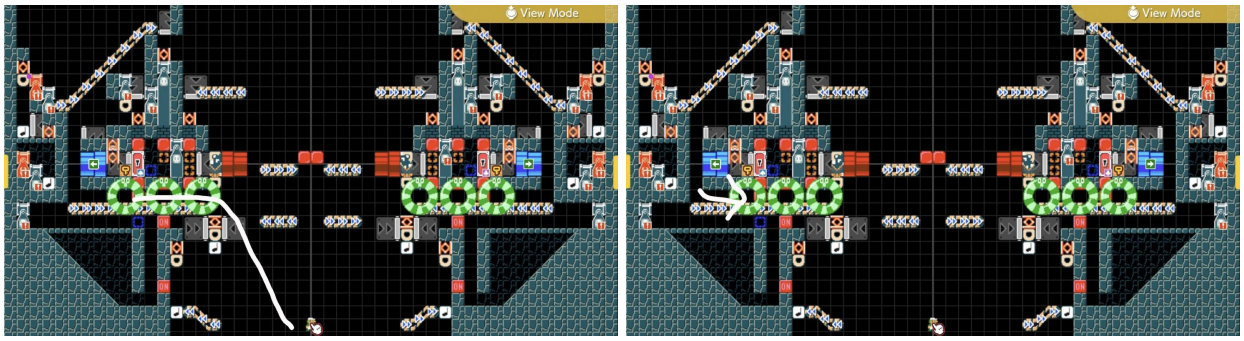
Obviously, these actions have to affect the other player in some way. I'll show the main interactions. Interpret these theoreticals as the first action is done by the player on the left and the second action is done by the player on the right.

Shot versus Shot (game continues):

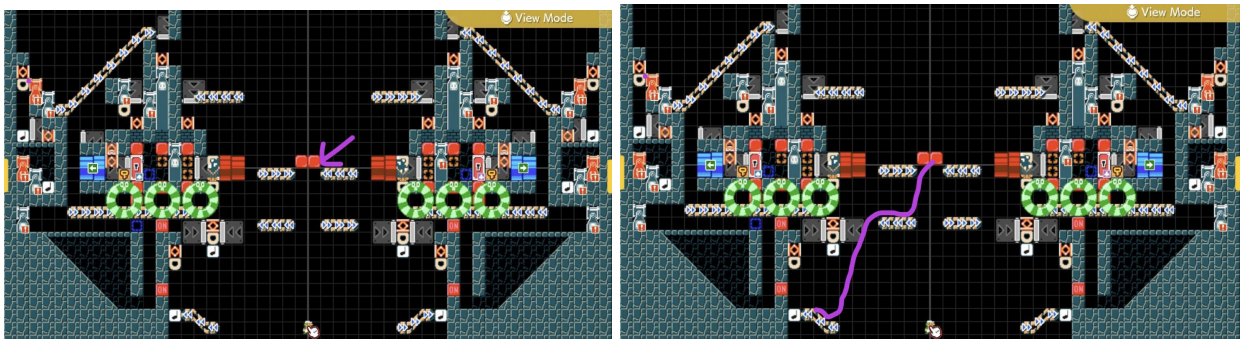


Block versus Shot (game continues):

Block versus Reload (game continues):



Reload versus Shot (right player wins):



Essentially every interaction is calculated on the cycle after the players chose their moves. This can lead to problems like the players killing each other, by “winning” on separate but adjacent cycles and the player who survived longer did not get to the key door in time before being killed by the spike ball. It is also very confusing for the player since the game does not feel very responsive to their actions, as everything is on a delay.

Killzone

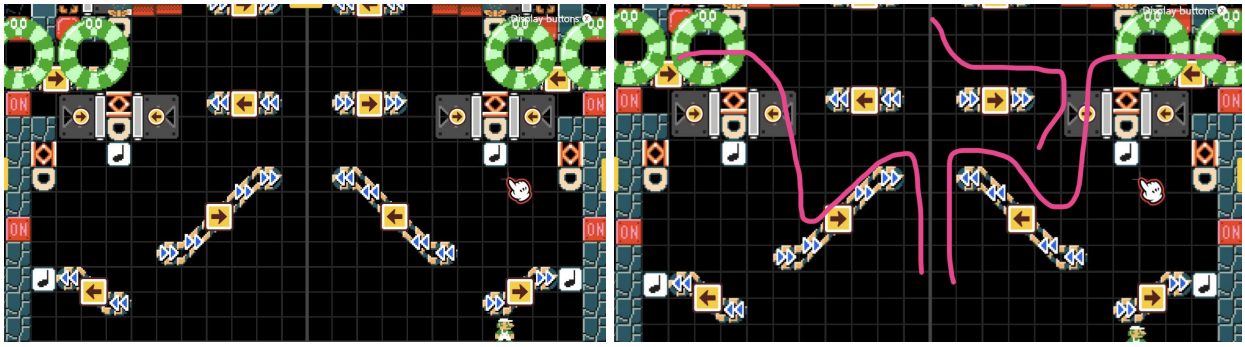
In the killzone, when an attacking player’s muncher is unblocked it will fall onto the conveyor belt at the bottom and rise up through the note blocks and springs and push the vertical spring onto the donut block at the top without reaching the donut block itself. The spring will bounce the muncher blocking the pipe, unlocking the pipe to release a spike ball, thus killing the player and breaking the box with the key. The winning player should then receive the key and enter their door.

Conclusion

This game is extremely hard to communicate to the common player. This being because they need to first understand the game concept, and then ultimately figure out how the level design relates to the game concept.

On top of this, there are many things that could go wrong. The cycle would be desynced any player enters a pipe at a different time than the rest of the players, and I don’t know if there is a way to force somebody into a pipe. And it is possible that both players kill each other and soft lock the game.

Another thing is that when you play this game, you should be able to see what the other player is doing. You can always see when they fire, but cannot differentiate between blocking and reloading. One way to partly counteract this would be to put these conveyor belts at the bottom at the expense of another two object:



The lines on the left side show a block vs reload interaction, and the lines on the right side show a shot vs block interaction, both of which will result in both munchers being thrown into the center of the screen, where both players can see the result and interpret what the other player played. However, they will see this a round too late, because it will be right before they have to act for the cycle two rounds later, not one. They will not be able to immediately interpret whether the other player either reloaded or blocked.

I would really like to see somebody make a single-elimination tournament with this idea. Since I have the objects limited to 50 per arena (without the changes made in the above images), there can be two arenas in the overworld or subworld. You could put two of them in the subworld as a semifinal, and then put the final in the overworld, and use the rest of the overworld objects as a demonstration to players of the game mechanics.

However, I am very bad with the specifics of how this game works (object limits, what exactly is different in multiplayer, etc.). I want to see somebody else make this work.