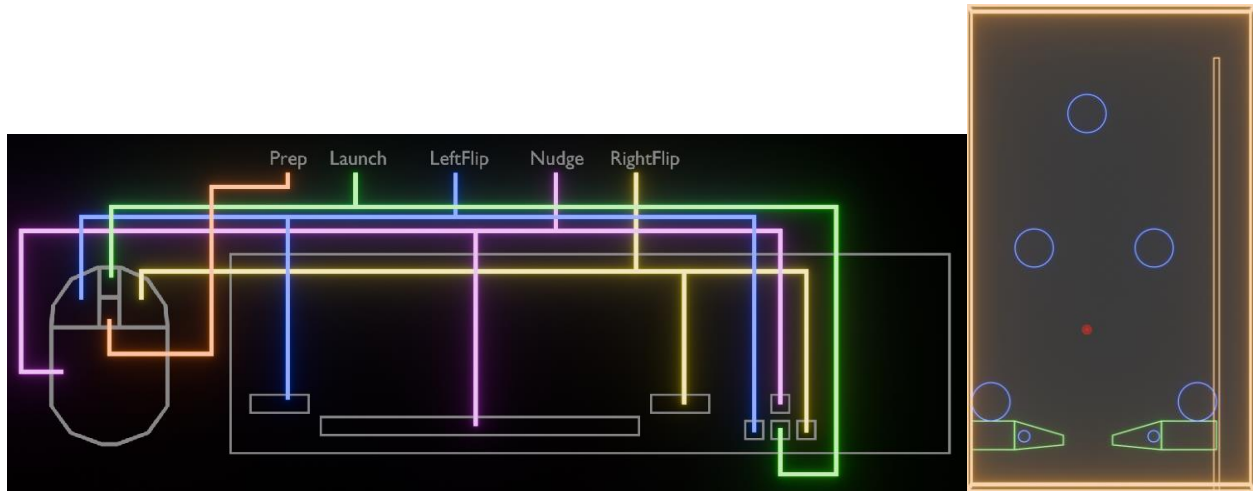


# Spring Interactive Design Page

This is an initial spec for a rudimentary pinball video game.



Above left: proposed input layout for main player actions. Above right: proposed initial playfield layout. Images shown are for illustrative purposes and may not be an exact representation of the final product.

Sf::RenderWindow will be set to 600U in x and 1016U in Y, displaying the entire pinball table in portrait.

The pinball, table and (if implemented) features such as plunger, flippers, drains, lanes, bumpers, kickers, targets, holes, saucers, spinners, gates and stoppers will be constantly visible on-screen.

For the purposes of this design, I will refer to the player inputs 'Prep', 'Launch', 'LeftFlip', 'RightFlip' and 'Nudge', assigning these inputs to the mouse buttons and keys in the attached diagram.

User Input will primarily be through the mouse, though I would also like to provide more traditional pinball keyboard input using L/R shift, spacebar and down arrow (and dedicated cursor key controls).

With a ball in the plunger lane, the player starts by calling 'Prep' multiple times and then calling 'Launch' (on mouse), or by holding 'Launch' for a few seconds and releasing, sending the ball into the playfield.

In the playfield, the ball will collide with the walls whilst continually drawn by gravity towards the trench at the bottom of the table. The ball may also collide / react to other playfield furniture (eg. Bumpers).

Movement and collision of the ball with the playfield environment will be achieved w/basic physics, employing trigonometry (eg. Pythagorean theorem), radial collision and SFML rect ('Intersects').

Players attempt to keep the ball out of the trench, using LeftFlip and RightFlip to trigger the flippers positioned left and right of the trench when they are in contact with the ball, sending it up the playfield.

The player starts with three balls. If the remaining ball on the playfield enters the trench, then the round is over. When all three balls have entered the trench, the game is over and the score is displayed.

Players can attempt to prevent the ball from entering unfavorable zones (eg. The trench) by calling 'Nudge' (usually by moving the mouse quickly), which will push the ball in the direction of that move.

If the game works with one ball and an empty playfield, then I'll attempt to add targets, bumpers and standard pinball elements, including multi-ball (multiple interactive balls moving around the screen).

Score will be kept in an overlay at the top-centre of the screen. Players score points by causing the ball to collide with playfield furniture (eg. Bumpers). Pinball collisions with environment play sound effects.

#### ASSETS REQUIRED

Art: Pinball sprite, table background, bumper sprites, flipper sprites.

Audio: Ball collision sound. Bell. Voice samples (3). Music stings (2).

Font: 1 for hud score, tbd.

SFML & Code: Collision detection by combining radial collision (for pinball and rounded elements of table), SFML Rect 'Intersects', detecting if object enters triangle by calculating area of sub triangles etc.

Movement by public Move() in pinball with direction and force arguments. Force passed on collision detection, redirection by reversing relevant vector values of pinball. Potentially finding normal of collided surface, though it's likely to be redirect based on ball's relative position to object origin. Flipper collision and force by distance from flipper 'hinge' while within Flipper rects. Momentum by lerp to target speed, friction 0.99 scalar. I can't indicate the correct gravity value without testing the drop of the ball during development.

---

#### PETE NOTES:

the question I have is about Entry to the system being different (and random), Perhaps start between the two flippers with some sort of swing meter for power and direction to determine the initial launch.

Robert I had thought that if you consider the top of table as being inside a circle then when the ball stops overlapping the semi circle it should hit the curved wall.

Robert remember 10 hours, walls 4, flippers 2, bumper 1 will be enough. When you get that simple table working add more stuff.