

1. Log on to Scratch <http://scratch.mit.edu/>.
2. Select “Try it out.”
3. **Upload** your file from the previous worksheet.
4. Create a **Flipper** that responds to the **space key** and pushes the ball when it hits it.
  - 4.1. Make a new Sprite called **Flipper**. Shape it like you would expect a flipper to look. Set the origin near one end to use as a pivot point.
  - 4.2. Select the **Flipper** sprite and add a **Green flag event handler** to set the angle (**direction**) of the flipper.
  - 4.3. Add a global variable called **Flipping** and set it to zero in the **Green Flag event handler**.
  - 4.4. Create an **Event handler** for the **Space key**.
  - 4.5. Add code to rotate (**turn**) the flipper to make it go up and then down. You may want to use a couple of **Repeat Controls**.
  - 4.6. **Set Flipping to 1** while the **flipper** is moving up, and then back to zero while the **flipper** is moving down. We will use this variable to control how the ball reacts when it hits the flipper.
  - 4.7. Adjust the flipping rate until it looks natural.
5. In this section we will add code to the **ball’s Forever** loop to make it respond to flipper. When the flipper is **Flipping**, the speed will depend on how far the ball is from the pivot point. Otherwise the flipper will behave like a wall.
  - 5.1. Create a new function (**Block**) called **hitFlipper** that has a **string argument** to identify which flipper was hit.
  - 5.2. In the **hitFlipper** definition call **hitWall** to set a new direction for the ball
  - 5.3. Use the global variable “**Flipping**” to determine what action to take.
    - 5.3.1. **If Flipping is “1”** then set **Speed** to **distance to the flipper / 4**.
    - 5.3.2. **If Flipping is “0”** treat the flipper as if it were a stationary wall (no **speed** change).
  - 5.4. Add code in the **Ball’s Forever** loop to call the **hitFlipper** function.
  - 5.5. Check your flipper’s response to the ball. Adjust the equations to improve the response. See the following hints:

- 5.5.1. If the flipper is moving too fast, it may move past the ball causing the ball to look like it is going through the flipper. Try moving the ball some extra steps to get out of the way.
  - 5.5.2. If the end of the flipper is sharp, the bounce angle could be very strange.
  - 5.5.3. The HitWall function works for stationary objects but may not be the best choice if the flipper is moving.
  - 5.5.4. What happens if your flipper pushes forward rather than rotating about a point?
6. Add a second **flipper**.
    - 6.1. Duplicate the first **flipper** but change its costume so that it rotates around the opposite end as the first **flipper**. You may want to change the direction that this flipper rotates when the space bar is hit.
    - 6.2. Add code to your **Ball's forever loop** to respond to the second **flipper**.
    - 6.3. Both **flippers** will respond to the space bar. If you want them to respond to different keys you may change the event handlers in each of the flippers. If you do this, you will have to include instructions to help players know how to control them.
    - 6.4. For a normal looking pinball machine space your flippers so that the **ball** can fit between them to hit the **trap**, but this is your game so you can put the **flippers** and **trap(s)** any where you want.
    - 6.5. Test and debug your code.
  7. Save and submit your work (Pinball #6).