

Penrose Stairs

Guided Project Instructions

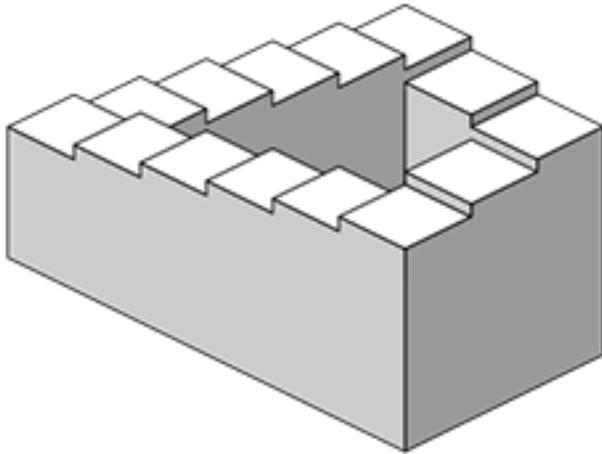


Figure 1 Penrose Stairs

Related Core Concepts:

Solenoid

React!

“The idea of continuous travel in a type of spiraling staircase can be realized.”

Learn It!

Objects that appear feasible in 2 dimensional drawings but are impossible in real life are called paradoxical illusions. Paradox illusions trick the viewer through manipulation of the brains understanding of how 3 dimensional objects should behave, of the most notable, the idea that adjacent edges must join. Lionel and Rodger Penrose created many of these illusions, one of which is the Penrose Stairs. The idea behind the Penrose stairs is someone would climb continuously but never get anywhere.

Many attempts have been made to try to replicate this illusion in the reality but the structure is truly impossible to create. The idea of continuous travel in a type of spiraling staircase, however, can be realized. In this module you will experiment with building a realization of the Penrose Stairs that will continuously move a ball around a track using solenoids and switches. You will have the opportunity to apply fundamental logic in order to understand the position of the ball and use actuation to keep it moving.

Build It!

In order to move a ball smoothly through a series of rails, we will first need to create a rail system. In this project four rails should be created in order to simulate the four stair cases that are connected. As a first step getting a single rail working with a solenoid and switch will be beneficial when creating the entire system. The infrastructure can be made out of LEGOs (the solution is done using LEGOs but feel free to use construction materials you feel comfortable with).

Task 1: Refer to Figure 1 to construct a single rail to allow a ball to travel based off of a switch input.

Task 2: Create LabVIEW code that monitors a switch input for a switch close and outputs a digital high on your National Instruments hardware for a static amount of time before returning to a low. Include decision making to allow the code to look for a second switch input or output another True on the same digital line. If the second switch close is received then output a digital high on another digital line.

Task 3: Test the rail system you created on different sizes and weights of the ball you are using. Find the optimum size and weight of the ball you will use for the final system.



Figure 2 Single Rail Example

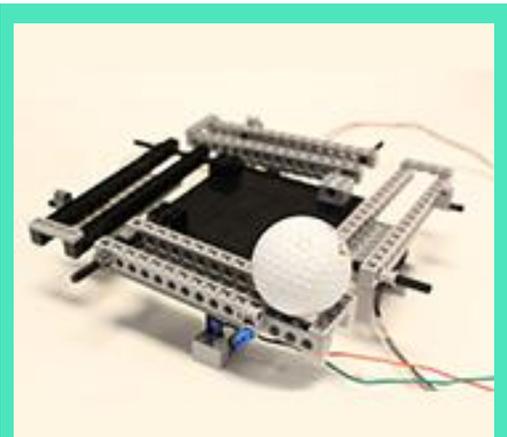


Figure 3 Completed Example Penrose Stairs

Task 6: Duplicate the two rail system in order to realize the full Penrose Stairs. You should be able to place a ball on any of the four rails, run the code and that rail should actuate to begin the process of moving the ball around the stairs (the four rails). Test your setup to ensure consistency. Include a self check to actuate a rail again if the ball gets stuck.

Task 4: Update your single rail system to include a contact switch which will be tripped by the ball rolling down the rail as well as a solenoid that can lift the rail to allow the ball to roll. Begin testing this manually by triggering the solenoid by hand and monitoring how the ball interacts with the switch. Optimize the design to allow for consistent switch triggering.

Task 5: Adapt your LabVIEW code to the single rail system then expand to have two rails perpendicular to one another. Fine tune the logic to cause the solenoid of the second rail to actuate based off of your switch scheme. The first rail should actuate causing the ball to roll and fall onto the second rail. Based off of the switches, your code should be able to tell when the ball has rolled onto the second rail and is ready to be moved down the second rail.

Guiding Questions:

- What factors do you have to overcome in moving the individual rails with solenoids?
- What are some other solenoid based actuators? What are the pros and cons of using a solenoid versus other actuators?
- Can multiple solenoids be on at once? Explain what considerations must be taken in order to have multiple solenoids on and why.

Expand it!

- Add control aspects to your design to allow a user to enter the number of complete rotations the ball should make before stopping. The user should be able to place the ball on any rail as before, then enter the number of revolutions and run the code. The ball should travel through that many revolutions around the rails and end back on the rail it was initially placed on.

Research It!

Actuators-Solenoids

http://www.societyofrobots.com/actuators_solenoids.shtml

Limit Switches

<http://www.marchantdice.com/forumimages/limit-switches.pdf>