Bicycle Jump

Brian Braeckel

Tyler Brown
Introduction

The Bicycle Jump project was created for the Eisenbergs’ Things That Think course at the University of Colorado at Boulder during the spring of 2007. The goal of the project was to build a moving mechanical toy which incorporated elements of computation. Its construction and implementation will be described below, along with its intended purpose and ideas for future enhancement.

Construction

Our project utilizes the Pico Cricket® construction kit to control the operation of a bicycle that jumps over obstacles which are attached to a conveyer belt. A small wooden bicycle lifts off the conveyer belt just before the obstacle hits it. By raising its front wheel before its rear, it mimics the way a mountain biker would jump over a rock on a trail. After the obstacle has passed under the bicycle, both wheels return to the ground and continue on their path. Obstacles can be placed anywhere on the conveyer belt which, in turn, can be rotated at variable speeds depending on how fast the user turns the hand crank. A series of two light sensors tell the Cricket® when to raise and lower the bicycle. Every piece of the project (except for the conveyer belt track) is made from wood, cut by a laser cutter, and glue to hold it together.

The conveyer belt is made out of a combination of Velcro® and textured sand paper tape. The Velcro® is on the outside of the track to allow for the attachment of obstacles, and the tacky surface is on the inside in order to create friction between the belt and the wheels that rotate it. Two wheels are used to turn the conveyer belt. One is attached to hand crank, and the other has an adjustable position to allow for fine tuning the tautness of the Velcro® belt.

Resting on top of the conveyer belt is a small wooden bicycle that turns its wheels when the conveyer belt rotates, creating the illusion of a moving bicycle. Two thin rods extend out perpendicular from the bicycle, attached at the axles of the wheels. These rods pass through tracks and into the box that houses the mechanical and computational elements of the project. The front track is a thin vertical line, while the rear track is significantly wider. The extra width of the rear track allows the front wheel to rise off the ground before the rear wheel does. As the front axle is raised by the motor, the rear axle slides forward until it hits the edge of the track before it rises.

Inside the box that contains the mechanics of the project is a series of pulleys and tracks, as well as a motor that raises the bicycle off the ground and pulls it back down again. The front axle rod is attached to a loop that, in turn, passes through two pulleys (one above and one below) and loops around the motor. The loop design allows the motor to not only raise the bicycle, but also pull it back down simply by reversing its direction. The front axle rod also passes through two guides that allow the motor to lift the weight of the bicycle from a location on the rod away from the bicycle’s center of mass. To further offset the weight of the bicycle, a counter weight is used.
Implementation

The main design was fairly simple. Make a conveyor belt that would allow items to be attached at random places. Place light sensors at the front and back of conveyor belt to sense an object passing by. Construct a box to hold the mechanisms for raising and lowering the bike. Raise the bike when the front sensor was triggered and lower the bike when the rear sensor was triggered.

The first major design problem that we came across was how to raise the bike. We wanted the bike to imitate a real life jumping style rather than simply raise both wheels at the same time. One of the solutions to this problem was to raise the wheels with cams. Although this would have probably worked, the cams would have to be approximately six inches in diameter on one side which would force the box holding the cams to be at least a foot in width. We didn’t want the project to look clunky, so we decided against using cams to raise the wheels of the bike.

Another design was to use a rack and pinion system to raise each wheel separately. Although this design would take up less room it would be very complicated. One of the racks or one of the wheels hubs would have to be floating in the x axis. This is because once the front wheel would start to raise the distance along the x axis between the two wheels would decrease. This would quickly stop the front wheel from raising more than half an inch.

Our final design for raising the wheels at different times was to create tracks for the axels of the bike. Although both tracks were simple vertical slits, the rear track is wider. When the front axel is lifted the rear wheel will stay on the ground and slide forward. Once the front axel is raised high enough the rear wheel will start to rise. When lowering the bike the rear wheel will land first and begin to slide backward until the front wheel lands. This is very similar to the proper technique of jumping a bike in real life.

Although this proved to be the simplest design we still ran into several problems along the way.

The first problem we thought of was that the bike’s center of mass would make it tilt severely in one direction. Although counter-weighting can change the center of mass it would be incredibly difficult to weight it precisely. We decided to place large washers on each side of the front axel track to prevent it from tilting. One problem with these washers is the amount of added friction. Another problem is that if the washers somehow tilt far enough they could pinch against the track. We attempted to create something that can best be explained as wheeled washers. These might have worked if we hadn’t used wood to create them. We decided to solve the friction problem with simple sanding and graphite. Our solution for the pinching problem was to simply create a larger gap between the two washers so that the uncontrollable angle change was less.

Once all of this was working properly we quickly realized a problem with the Lego® motor’s power. It was enough to raise the bike without friction yet with the added friction it only raised it correctly about half the time. The friction also caused problems with the bike lowering back to the ground. To solve the raising of the bike we added a
simple counter-weight. To solve the lowering of the bike we tried a couple different approaches.

One approach was to use two Lego® motors. One motor was going to raise the bike and one was going to lower it. This might work with some motors yet the Lego® motors are only controlled by the operation time. Out of the two motors the one that wasn’t doing the lifting was spinning faster. This caused slack in the string which in turn got caught on several objects and created knots. Our other solution was to use a loop of string around a single motor. After several attempts at coating strings in different substances and using different forms of grip tape we finally used a toothed belt for both the string and the motor axel.

We had the bike working correctly yet still needed a conveyor belt that would spin around and allow objects to be attached at any point. We made the belt with grip tape on one side and Velcro® on the other. Our objects simply had Velcro® attached to the bottom of them. We originally planned to use a Lego® motor to power the conveyor belt yet, after seeing its power with the bike, we decided to make the conveyor belt hand powered. This decision also showed the adaptability of the bike to varying speeds of the conveyor belt.

Once all of the mechanics were working correctly we programmed the Pico Cricket® using Pico Blocks®, a simple API created by Pico®. The programming was fairly straightforward, turn the motor on when the first light sensor was triggered and reverse it when the second light sensor was triggered. The only problem we had to solve for the programming was where to set the threshold for the difference in light. We simply calculated the ambient light when the Pico Cricket® was turned on and set the threshold to an extremely low value.

Overall, the largest problems throughout the project were concerned with friction, the power of the Lego® motor, and matching the correct dynamics of jumping a bike.

**Context**

The Bicycle Jump project was motivated in part by a paper construction kit of a motorcyclist that we assembled earlier in the semester and part by our interest in playing with things that have two wheels. We are both mountain bikers that wanted to create something familiarly appealing.

In addition to our personal interests in the subject, Arthur Gansen’s work influenced our project to some degree. His elegant and artistic contraptions motivated us to construct a toy that could be mounted somewhere and gracefully brought to life by the turning of a hand crank.

Educationally speaking, our project does little to influence its users other than to falsely suggest that a bicycle can jump an obstacle on its own, without a rider. The project’s goal was purely aesthetics and entertainment.
Improvements

There are few functional improvements that we would consider, but many aesthetic enhancements that could be made. Our original goal was to make the whole project out of translucent Plexiglas® material, and to spend more time on the design of the bicycle and the exterior of the mechanical box in order to add to the artistic appeal of the toy. As is, the project is not as attractive or as robust as it could have been.

One thing we could have done to improve the functionality would have been to update the source code to allow for multiple objects. We anticipate problems with the current process of detecting objects with light sensors if additional obstacles are added to the conveyer belt, or if the obstacles are long enough to set off both light sensors at the same time.
Photographs
• ENGINE #1 DRIVES CONVEYOR BELT
• LIGHT SENSOR #1 TRIGGERS ENGINE #2
• ENGINE #2 USES PULLEY TO RAISE BICYCLE
• LIGHT SENSOR #2 TRIGGERS ENGINE #2 IN OPPOSITE DIRECTION AS BEFORE
• ENGINE #2 LOWERS BICYCLE BACK TO THE CONVEYOR BELT

• WIDE REAR WHEEL TRACK CAUSES REAR WHEEL TO ROLL FORWARD BEFORE LIFTING AND RAISE AFTER THE FRONT WHEEL.