Mubarik Mohamoud
6.101 Project report

The Bouncing ball analog computer

Introduction

The goal of this project was to design and build a small analog computer called "The Bouncing Ball Simulator". The Bouncing Ball Simulator was first designed by a German Company called Telefunken. The Telefunken design was aimed for existed general purpose analog computers. Since then few people tried to design them with stand-alone op-amps to my knowledge. The Bouncing Ball Simulator calculates the position of a ball falling from some initial height and bouncing due to the Newton's law of motion. What makes this an analog problem is the fact that you can simulate the horizontal and the vertical motion of the ball independently. That feature of the problem fits analog because analog computations are done in parallel. Furthermore, analog computers above all exist because of the power to integrate; hence the problems solved are almost always a set of differential equations, and this problem is in fact a set of well-known differential equations.

Design

The design of the project involves three different and mostly independent computations. First, designing the simulation of the horizontal position over time, this in reality should just be a rather slow asymptotic growth, where the ball starts accelerating relatively fast in the beginning and slows down due to the forces of friction, and restitution (elasticity of the ball), but the actual design here will be different. Second, the vertical motion of the ball governed by three coefficients of acceleration, gravity, elasticity or deformability of the ball and the coefficient of air resistance and. Finally, the projecting a ball on the screen of an oscilloscope for visual demonstration. Each sub-design went through the following process.

Simulating the horizontal position

The design of the ball’s horizontal position was changed several times, the earlier design involved restricting the ball in small an area where there are walls on both sides, and the ball bounces back and forth. Figure 1 shows what the position of the ball would look like over multiple steps.

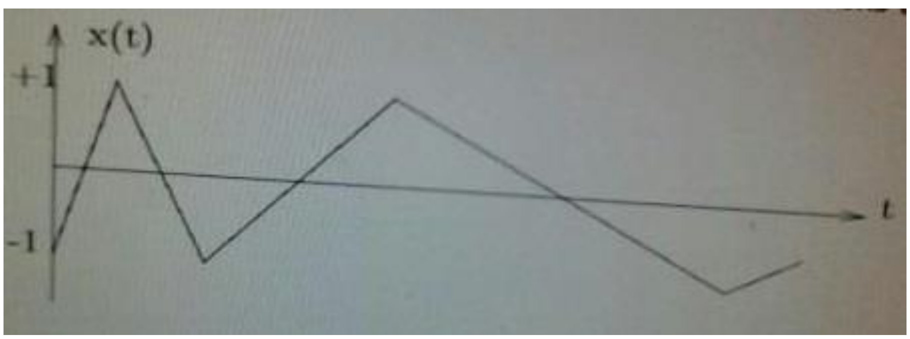


Figure 1

As the ball loses energy it takes longer and longer to bounce back.

In designing this, one would start with some initial condition and small deceleration. The initial condition in this is some velocity v_0. If the small negative acceleration is α the horizontal position can be described as follows:

$$v(t) = \int_0^{\tau} \alpha dt + v_0$$

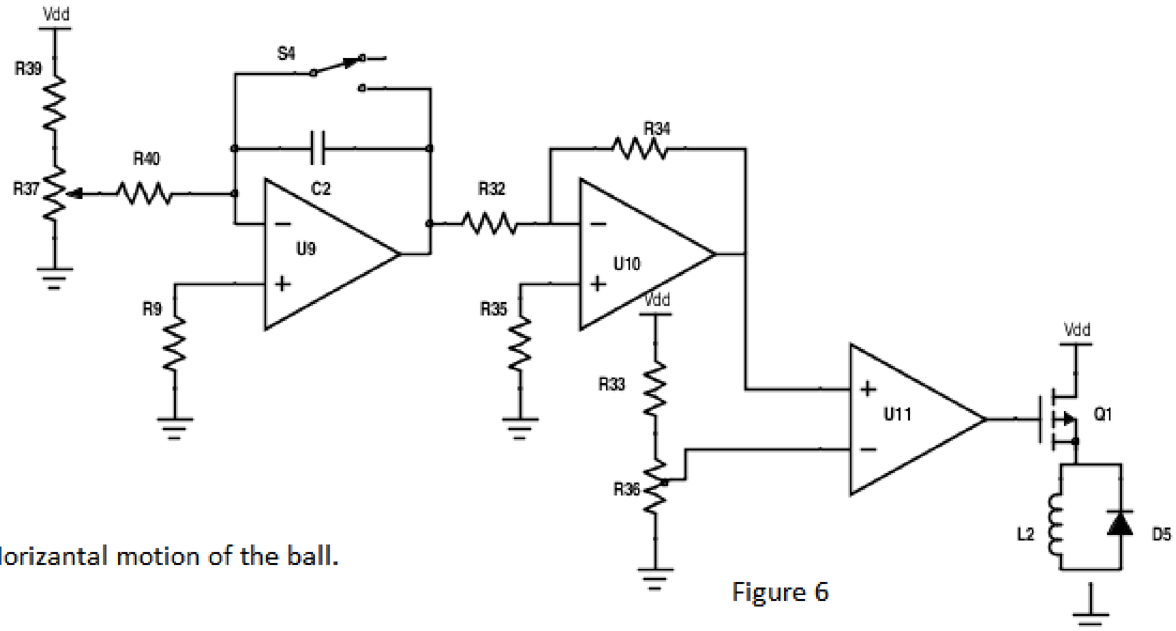
$$x(t) = \int_0^{\tau'} v(t) dt$$

The challenge here is changing the directions, and I haven't built this circuit but, the one I have built is similar with fewer restrictions. For this design, the horizontal position is just the sawtooth wave generated by the circuit shown on figure 2.

The integrator (U9) generates an increasing voltage in magnitude, and that is inverted by U10. The output of U10 is the horizontal position of the ball, which is passed into the comparator U11. U11 controls Q1 to repeatedly reset the initial condition to zero and hence sets the position to zero. S4 is the contact of a single pole double throw relay switch, and L2 is its coil. Therefore, when Q1 is conducting the common of S4 is connected to the normally open throw, which shorts out the capacitor C2 to set the initial condition.

To get a reasonable integration time constant C2*R40 needs to be big enough. Since, it is not easy to find a non-polarized capacitor to make that with a reasonably small resistor, big resistors are needed. Therefore, R40, and R9 are both 5MegΩ resistors with high input resistance U9 op-amp, namely LF356. R37, and R39 can be picked to set the sweep, R32, and R34 are both 10kΩ resistors, where R35 is a 5kΩ resistor. U10 is LM741 and U11 is LM311. The LM311 comparator needs a pull-up resistor connecting the output to pin8 or Vdd. Q1 is IRF540 and D5 is 1N4001.

□



Horizantal motion of the ball.

Figure 6

Simulating the vertical position

When calculating the vertical position of the ball, we are interested in the coefficients of acceleration. In this case, there are three coefficients of acceleration to be considered. The coefficient of gravitational acceleration, which we are simulating as a constants acceleration that is driving the ball down all the time. The two other coefficients are the damping coefficient, and the coefficient of restitution or elasticity of the ball. The Coefficient of resistance (damping) is proportional to the velocity of ball, so it increases with the speed. The elasticity or the restitution of the ball, which is how deformable the ball is sharply changes the direction of the ball. Coefficient of restitution of one means the ball doesn't deform all and keeps on bouncing.

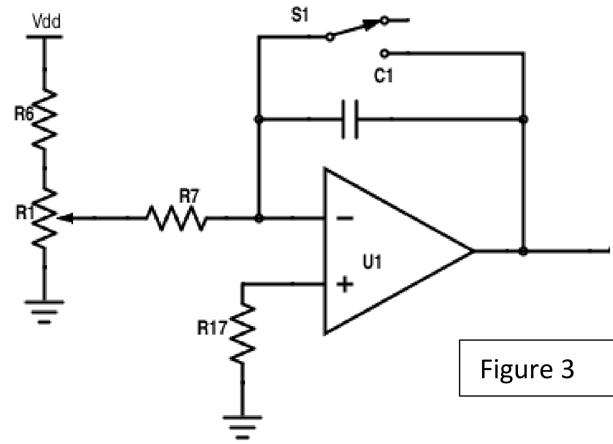


Figure 3

As figure 3 shows, we start with the gravitational acceleration, $\dot{y} = \int g dt + v_0$, to generate the vertical velocity of the ball. S1 is one contact of a 2PDT relay. R1 and R6 can be chosen to set the gravitational acceleration constant. Their relative values matter but their values don't matter as long as they are reasonably big, 10k, and 3.9k were used for R1 and R6 respectively. R7, and R17 are large resistors $5Meg\ ohm$ resistors were used to generate a relatively large integration time constant. $.1\mu F$ Capacitor was used for C1, and LF356 was used as U1, any other JFET input general purpose op-amp should work fine.

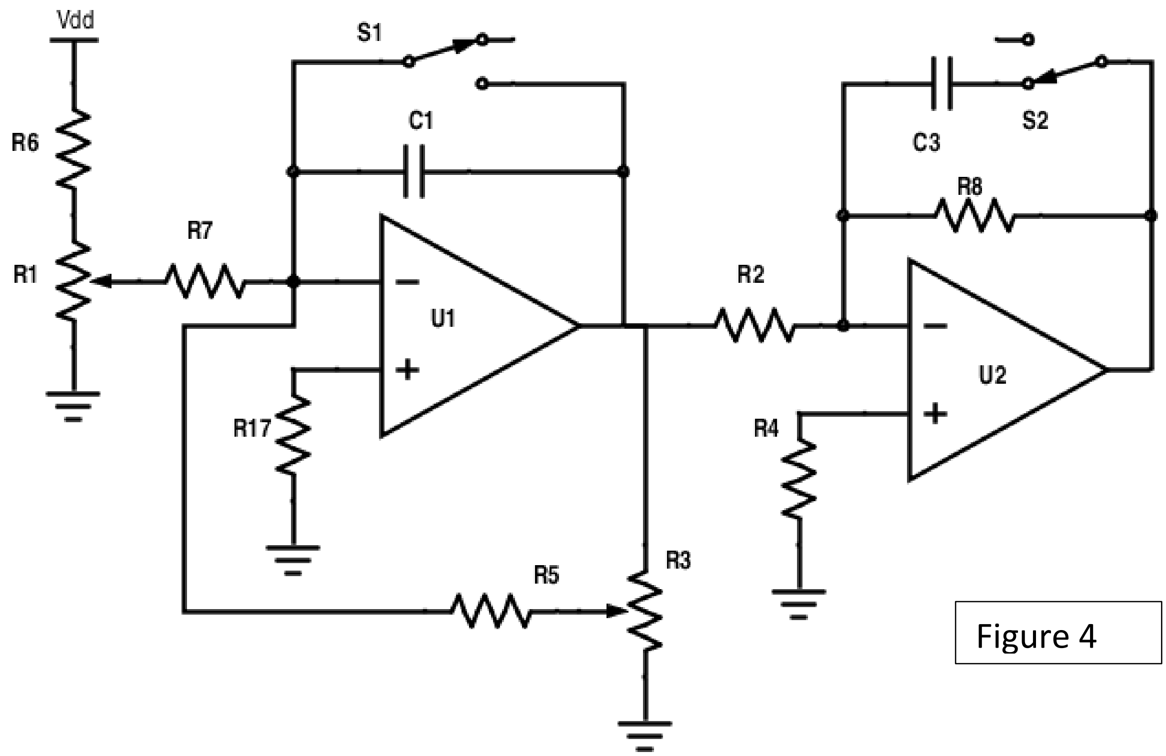


Figure 4

Few more components are added to figure 3 on Figure 4 to add damping to the system and to integrate the velocity. The feedback loop is designed to add damping to system where R3 is used to control the coefficient of air resistance. R5 is also a 5Meg resistor, where R3 is a 10k potentiometer. With this feedback the acceleration becomes $\ddot{y} = d\dot{y} - g$, where d is the damping coefficient. The integrator (U2) integrates the velocity to calculate the free fall position. R4 and R2 are 5Meg resistor where R8 is a 100Meg resistor for maintaining close loop when setting the initial condition. U2 is also a JFET input general purpose operational amplifier, LF356 is used here as well. C3 is .1µF capacitor, where s2 is the second contact of 12V 2PDT relay.

The deformability of the ball needs to take the mass of the ball into account, hence we can use the spring equation to determine the path of the ball as it reaches what is designated to be the floor. Therefore the acceleration of the ball is, $\ddot{y} = \begin{cases} dy - g, when\ free - fall \\ dy - g + \frac{c}{m} * y, when\ going\ up \end{cases}$. Figure 5 shows the complete circuit with the initial condition setting components.

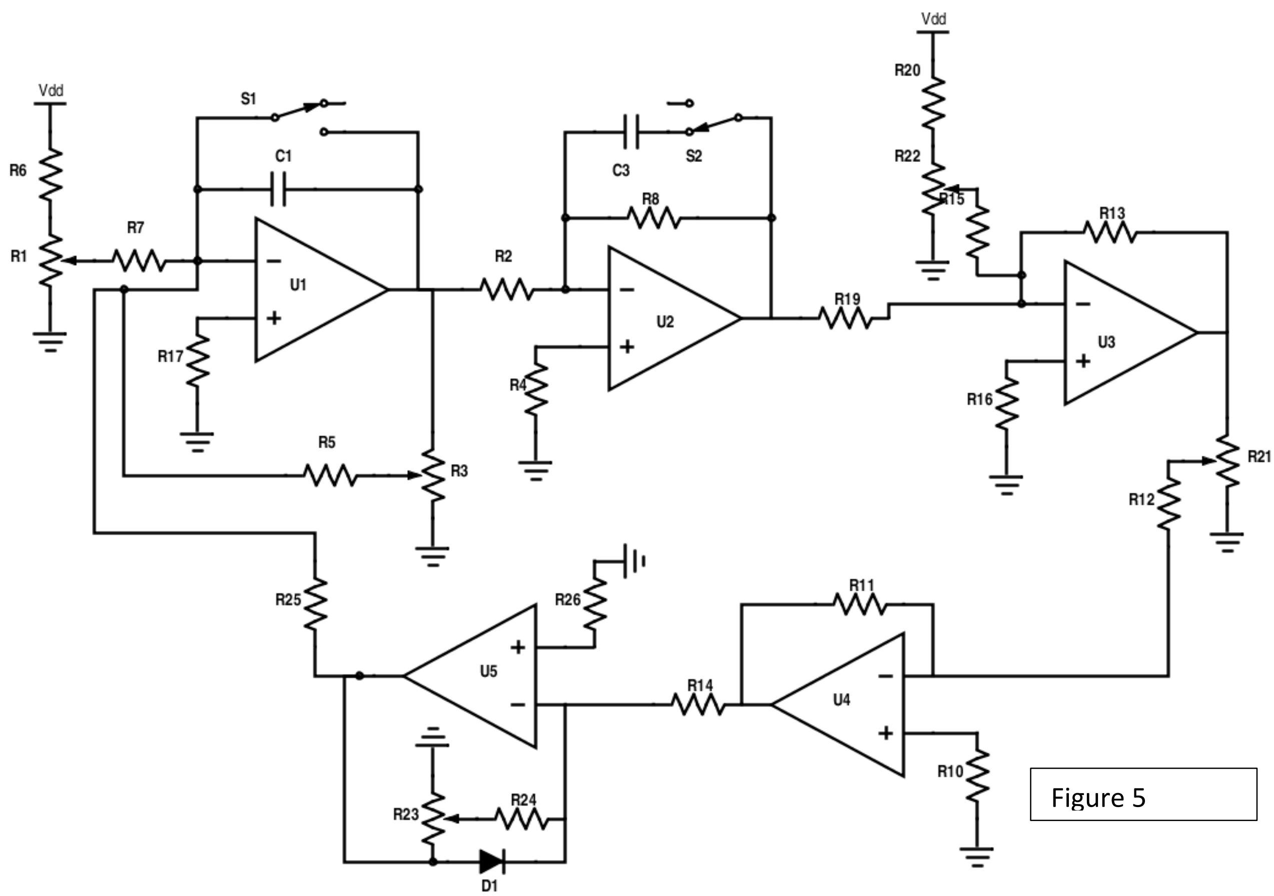


Figure 5

U2 which is a general purpose operational amplifier (LM741 was used) sets the floor by adding an offset to the signal. R20, and R22 set the offset and they are 5kΩ, and 10kΩ respectively. 10kΩ was used for both R19 and R13, and R16 is a 5kΩ resistor. R21 setts the coefficient of restitution. U4, LM741, is used to invert the signal and both R11 and R12 are 10kΩ, where R10 is 5kΩ. U5, LM741, is used to regulate the application of the restitution and used to divide the mass. D1 which is 1N4148 was chosen so it doesn't conduct until the signal level reaches the floor. R23 which is a 2kΩ potentiometer is used to divide the divide by the mass. R24 is 10kΩ, R14 is 5kΩ, and R26 is 3.9kΩ.

Setting the initial conditions is something that was rarely covered as far as my research goes, and I had had a lot of problems making it work. However, the circuit on figure 6 worked fine. U6 is connected to the normally open port of S2 (the second contact of the 2PDT relay) for setting the initial height. U2 needs to be able to operate rail to rail, so LT1632 was used. L1 is the coil of the 2PDT relay, and S3 is toggle switch used to trigger the initial condition. The initial condition can be triggered periodically using a MOSFET in place of S3 where 555 Timer in mono-stable configuration is used to turn on and off the MOSFET. The value of R18 determines the charging time and bigger values are better when manually regulating the switch but otherwise it depends on the time constant of your regulating circuit.

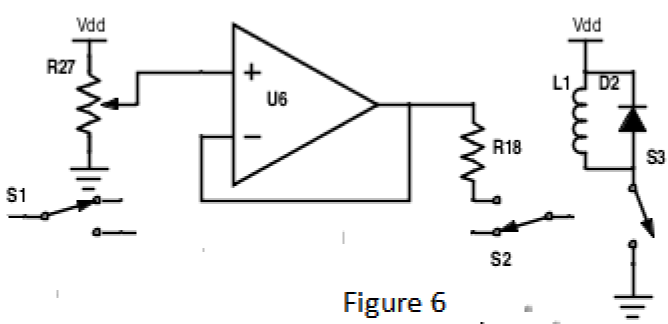


Figure 6

Putting all together, the circuit on figure 7 was used to simulate the vertical position of the ball. The performance of circuit was very sensitive to the value of R23 or the mass setting potentiometer. One may want set all the potentiometer around half way and use signals from the function generator for debugging. Connecting R20 to waveforms from the Function generator are comparable with a bouncy or unstable floor, hence with low frequency pulse wave you should be able to detect a response that is similar to the trajectory of a bouncing ball repeating. Figure7 shows what a photo of the performance of the circuit.

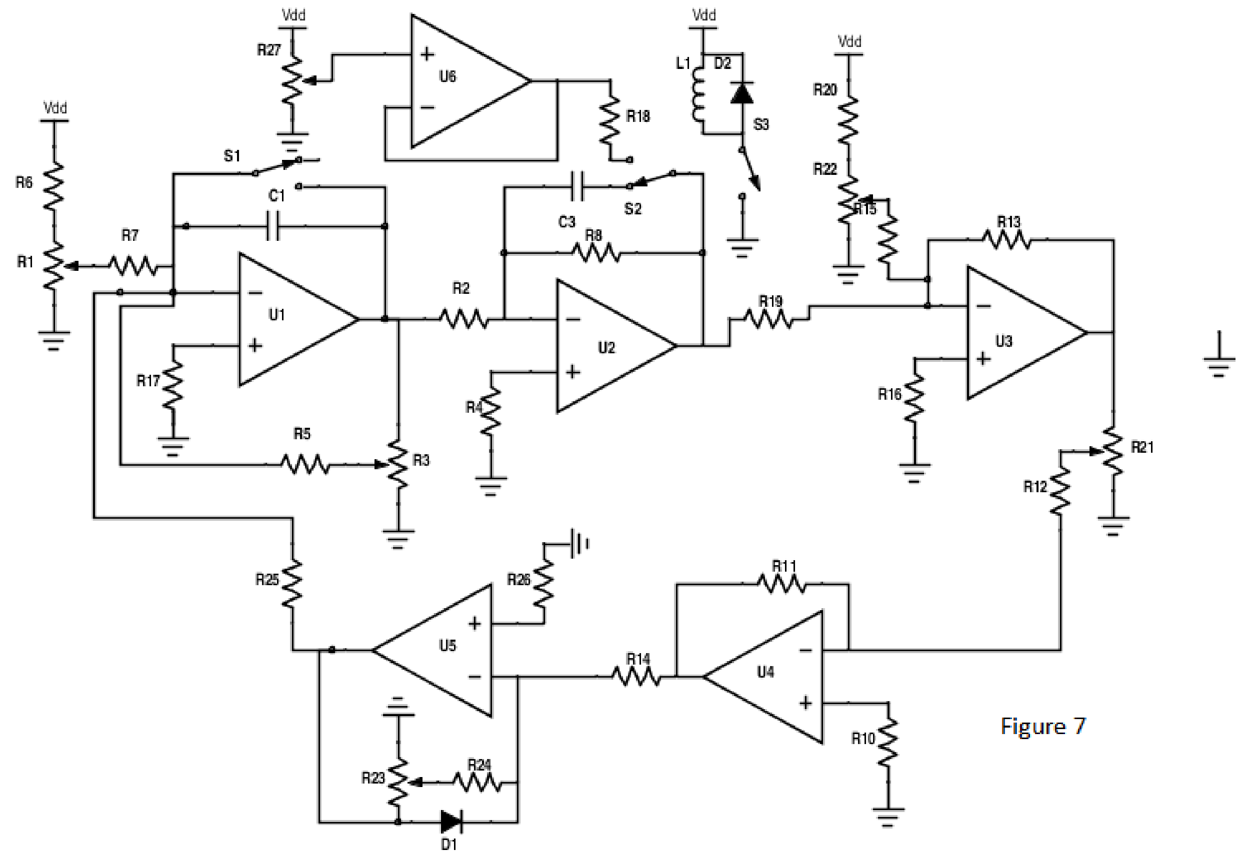


Figure 7

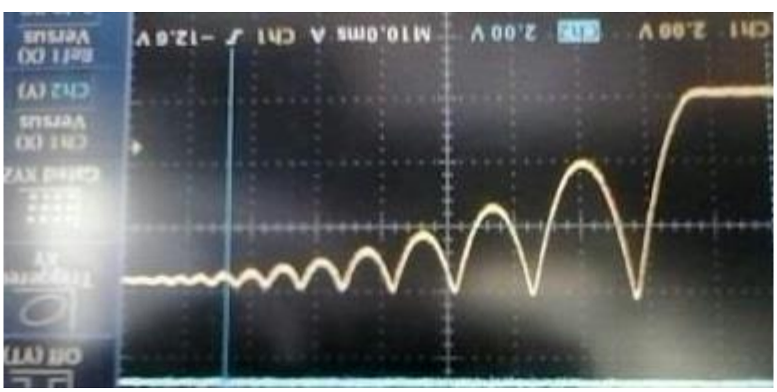


Figure 8

Projecting a ball on an Oscilloscope

To create a ball like figure on an oscilloscope screen, cosine/sine pair with relatively high frequency is necessary. Since the power of analog computers is integration, we need to solve the following linear second-order differential equation, $\ddot{y} + \omega y = 0$. The circuit on figure 9 exactly does that. This circuit is called Quadrature oscillator generates cosine/sine pair as needed. The two Zener, 1N747A, diodes are added to limit the amplitudes, as this circuit with high frequency will quickly drive the op-amps into saturation. All the Resistors are 10K, except R28 which is 9k and all the capacitors are $.01\mu F$. Any general purpose op-amp should work fine, but LF356 was used for this experiment.

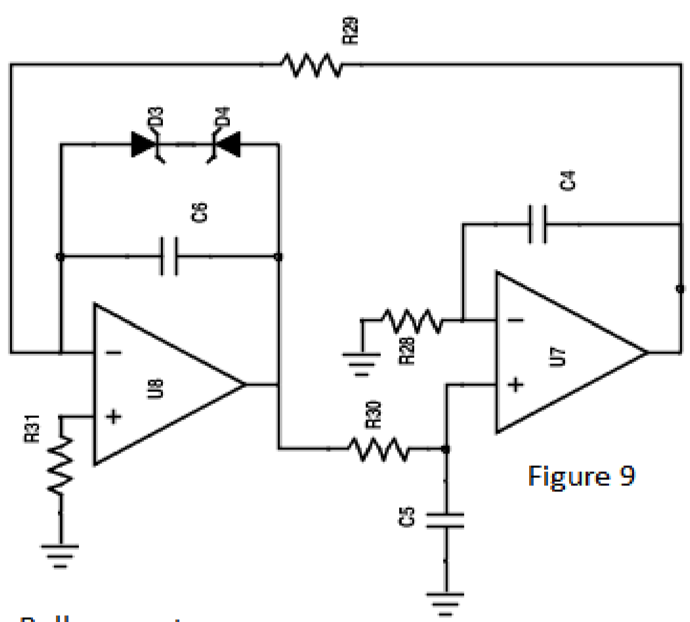
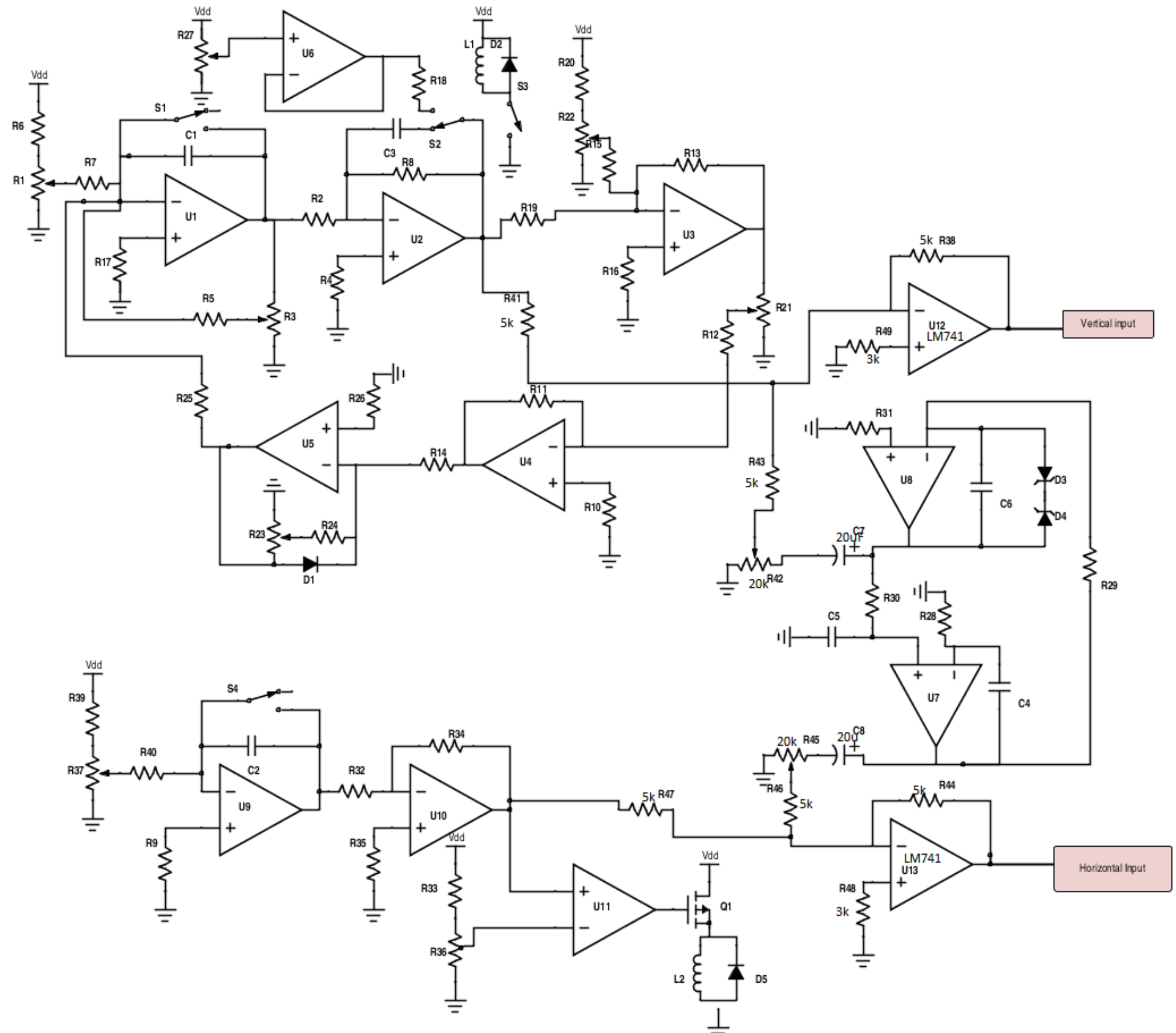


Figure 9

Ball generator



The complete circuit.

Summary

Upon deciding a final project, I had no idea how analog computers work neither have I ever seen or heard about an analog computer. Nevertheless, the existence of operational amplifiers indicated that people must have used analog circuits for computation, which sounded to be a good way to get a deeper intuition of how the analog components around us work. In the course of my research, stumble upon the design of several small analog computers, such as the Lorenz equations solver, most of which had complete op-amp based designs online. However, there was one that was physically intuitive but had little available online circuitries other than plugging wires in to a general purpose analog computer, and that was the bouncing ball simulator. Using my intuition of what the motion of a bouncing ball will look like and the limited resources available I decided that this will be a rather doable and educating project. It was educating!

Helpful Sites

http://www.analogmuseum.org/library/handson.pdf

https://courses.engr.illinois.edu/ece486/labs/lab1/analog_computer_manual.pdf

https://books.google.com/books?id=y1DpBQAAQBAJ&pg=PA136&lpg=PA136&dq=displaying+ball+on+an+oscilloscope+screen&source=bl&ots=8qC49DdQOk&sig=3E9pkq4oEw83CWN8cD12x-4nKck&hl=en&sa=X&ei=3OclVbnKKcWosAW3jYDYCg&ved=0CE0Q6AEwCg#v=onepage&q=displaying%20ball%20on%20an%20oscilloscope%20screen&f=false

http://www.analogmuseum.org/english/examples/bouncing_ball_box/

http://www.analogmuseum.org/english/examples/bouncing_ball/